

FileMaker® Server 13

Publicazione Web personalizzata
con PHP



© 2007-2013 FileMaker, Inc. Tutti i diritti riservati.

FileMaker, Inc.

5201 Patrick Henry Drive

Santa Clara, California 95054 Stati Uniti

FileMaker e Bento sono marchi di FileMaker, Inc. registrati negli U.S.A. e in altri Paesi. Il logo della cartella, FileMaker WebDirect e il logo Bento sono marchi di FileMaker, Inc. Tutti gli altri marchi sono di proprietà dei rispettivi proprietari.

La documentazione di FileMaker è protetta da copyright. Non è permesso fare copie o distribuire questa documentazione senza previa autorizzazione scritta di FileMaker. È possibile utilizzare la presente documentazione soltanto unitamente a una copia del software FileMaker concessa in licenza.

Tutte le persone, le società, gli indirizzi e-mail e gli URL elencati negli esempi sono fittizi e ogni riferimento a persone, società, indirizzi e-mail o URL esistenti è puramente casuale. Gli autori sono elencati nei documenti Riconoscimenti forniti insieme con questo software. I prodotti di terze parti e gli URL sono citati unicamente a scopo informativo e non costituiscono obbligo o raccomandazione. FileMaker, Inc. non si assume alcuna responsabilità nei confronti delle prestazioni di questi prodotti.

Per ulteriori informazioni, visitare il nostro sito Web all'indirizzo <http://www.filemaker.com/it/>.

Edizione: 01

Indice

<i>Prefazione</i>	6
Informazioni su questa guida	6
Capitolo 1	
<i>Introduzione alla Pubblicazione Web personalizzata</i>	7
Motore per la Pubblicazione Web	8
Modalità di elaborazione di una richiesta del Motore per la Pubblicazione Web	8
Pubblicazione Web personalizzata con PHP	9
Pubblicazione Web personalizzata con XML	9
Confronto tra PHP e XML	9
Motivi per scegliere PHP	9
Motivi per scegliere XML	9
Capitolo 2	
<i>Informazioni su Pubblicazione Web personalizzata con PHP</i>	11
Caratteristiche principali della Pubblicazione Web personalizzata con PHP	11
Requisiti della Pubblicazione Web personalizzata	12
Requisiti per la pubblicazione di un database mediante Pubblicazione Web personalizzata	12
Requisiti degli utenti Web per accedere ad una soluzione di Pubblicazione Web personalizzata	12
Connessione a Internet o una rete intranet	13
Installazione manuale di FileMaker API per PHP	13
Operazioni successive	14
Capitolo 3	
<i>Preparazione dei database per la Pubblicazione Web personalizzata</i>	15
Attivazione della pubblicazione Web personalizzata con PHP per database	15
Creazione dei formati per la Pubblicazione Web personalizzata con PHP	16
Protezione dei database pubblicati	16
Accesso ad un database protetto	17
Pubblicazione dei contenuti dei campi Contenitore sul Web	19
Campi Contenitore incorporati in un database	19
Campi Contenitore con file di riferimento	20
Campi Contenitore con dati memorizzati esternamente	21
Visualizzazione dei dati dei campi Contenitore da parte degli utenti Web	23
Script di FileMaker e Pubblicazione Web personalizzata	23
Suggerimenti e considerazioni sugli script	23
Comportamento degli script nelle soluzioni di Pubblicazione Web personalizzata	25
Trigger degli script e soluzioni di Pubblicazione Web personalizzata	25
Capitolo 4	
<i>Pubblicazione Web personalizzata con PHP</i>	26
Utilizzo del Motore per la Pubblicazione Web con soluzioni PHP	26
Operazioni generali per la Pubblicazione Web personalizzata con PHP	26

Capitolo 5

Utilizzo di FileMaker API per PHP

	28
Dove ottenere informazioni aggiuntive	28
Informazioni di riferimento a FileMaker API per PHP	28
Esercitazioni di FileMaker API for PHP	29
Esempi FileMaker API per PHP	29
Utilizzo della classe FileMaker	29
Oggetti classe FileMaker	29
Oggetti di comando FileMaker	30
Collegamento a un database FileMaker	30
Utilizzo dei record	31
Creazione di un record	31
Duplicazione di un record	31
Modifica di un record	31
Eliminazione di un record	32
Esecuzione di script di FileMaker	32
Recupero della lista di script disponibili	32
Esecuzione di uno script FileMaker	33
Esecuzione di uno script prima di eseguire un comando	33
Esecuzione di uno script prima di ordinare un gruppo di risultati	33
Esecuzione di uno script dopo la creazione di un gruppo di risultati	33
Ordine di esecuzione degli script	34
Utilizzo dei formati FileMaker	34
Utilizzo dei portali	35
Elenco dei portali definiti in un formato specifico	35
Recupero dei nomi portale per un oggetto risultato specifico	35
Recupero di informazioni sui portali per un formato specifico	35
Recupero di informazioni per un portale specifico	35
Recupero dei nomi tabella per un portale	35
Recupero dei record del portale per un record specifico	36
Creazione di un nuovo record del portale	36
Eliminazione di un record da un portale	36
Utilizzo delle liste valori	36
Recupero dei nomi di tutte le liste valori di un determinato formato	36
Recupero di una matrice di tutte le liste valori di un determinato formato	37
Recupero dei valori per una lista valori indicata	37
Esecuzione di richieste di ricerca	38
Utilizzo del comando Find All	39
Utilizzo del comando Find Any	39
Utilizzo del comando Find	39
Utilizzo di un comando Compound Find	40
Elaborazione dei record in un gruppo di risultati	42
Filtraggio delle righe del portale restituite dalle richieste di ricerca	43
Pre-convalida di comandi, record e campi	43
Pre-convalida dei record in un comando	45
Pre-convalida dei record	45
Pre-convalida dei campi	45
Elaborazione degli errori di convalida	45
Gestione degli errori	47

Capitolo 6

Preparazione, test e monitoraggio di un sito

	48
Preparazione di un sito Pubblicazione Web personalizzata	48
Test di un sito Pubblicazione Web personalizzata	49
Monitoraggio di un sito	50
Usò dei registri degli accessi e degli errori del server Web	50
Utilizzo del registro del Motore per la Pubblicazione Web	51
Utilizzo del registro errori del Modulo Server Web	53
Utilizzo dei registri Tomcat	53
Risoluzione dei problemi del sito	54

Appendice A

Codici di errore per la Pubblicazione Web personalizzata con PHP

	55
Codici di errore per database FileMaker	55
Codici di errore per i componenti PHP	62

Indice

63

Prefazione

Informazioni su questa guida

Si suppone che il lettore di questa guida abbia esperienza con PHP, sappia sviluppare siti Web e utilizzare FileMaker®Pro per creare database. È necessario comprendere le nozioni di base sulla progettazione dei database con FileMaker Pro e i concetti di campo, relazione, formato, portale e contenitore. Per informazioni sul FileMaker Pro, vedere la Guida di FileMaker Pro.

Questa guida fornisce le seguenti informazioni sulla Pubblicazione Web personalizzata con PHP su FileMaker Server:

- come sviluppare una soluzione di Pubblicazione Web personalizzata con PHP
- come pubblicare i database utilizzando PHP
- requisiti degli utenti Web per accedere ad una soluzione di Pubblicazione Web personalizzata
- requisiti per utilizzare FileMaker API per PHP per ottenere i dati dai database ospitati da FileMaker Server

Importante È possibile scaricare la documentazione di FileMaker in formato PDF all'indirizzo <http://www.filemaker.com/it/support/index.html>. Eventuali aggiornamenti a questo documento sono disponibili anche sul sito web.

La documentazione di FileMaker Server comprende le seguenti informazioni:

Per informazioni sui seguenti argomenti:	Vedere
Installazione e configurazione di FileMaker Server	<i>Guida introduttiva di FileMaker Server</i> <i>Guida di FileMaker Server</i>
Accessibilità dei formati dei database di FileMaker Pro e FileMaker Pro Advanced agli utenti dei browser Web attraverso una rete intranet o Internet	<i>Guida di FileMaker WebDirect™</i>
Pubblicazione Web personalizzata con PHP	<i>Pubblicazione Web personalizzata di FileMaker Server con PHP</i> (questo manuale)
Pubblicazione Web personalizzata con XML	<i>Pubblicazione Web personalizzata con XML di FileMaker Server</i>
Installazione e configurazione dei driver ODBC e JDBC, e utilizzo di ODBC e JDBC	<i>Guida ODBC e JDBC di FileMaker</i>
Istruzioni SQL e standard supportati da FileMaker	<i>Guida SQL di FileMaker</i>

Capitolo 1

Introduzione alla Pubblicazione Web personalizzata

Con FileMaker Server, è possibile pubblicare i database FileMaker su Internet o su una rete intranet nei seguenti modi.

FileMaker WebDirect: Con FileMaker WebDirect, è possibile pubblicare rapidamente e facilmente i formati da un database sul Web. Non è necessario installare un software aggiuntivo, chiunque dispone di un browser Web compatibile e abbia accesso a Internet o a una rete intranet può connettersi alla soluzione FileMaker WebDirect per visualizzare, modificare, ordinare o eseguire ricerche all'interno dei record, a condizione che disponga dei privilegi di accesso necessari.

Con FileMaker WebDirect, è necessario che sul computer host sia in esecuzione FileMaker Server. L'interfaccia utente è simile all'applicazione desktop FileMaker Pro. Le pagine Web e i moduli con cui l'utente interagisce dipendono dai formati e dalle visualizzazioni definite nel database FileMaker Pro. Per ulteriori informazioni, consultare la *Guida di FileMaker WebDirect*.

Pubblicazione statica: Se i dati vengono modificati di rado oppure se non si desidera che gli utenti si connettano dinamicamente al database, è possibile ricorrere alla pubblicazione statica. La pubblicazione statica consente di esportare i dati dal database FileMaker Pro per creare una pagina Web personalizzabile con HTML. La pagina Web non riflette le modifiche apportate alle informazioni nel database e gli utenti non si connettono al database (con FileMaker WebDirect, i dati vengono aggiornati nel browser Web ogni volta che vengono aggiornati sul database). Per ulteriori informazioni vedere la Guida di FileMaker Pro.

Pubblicazione Web personalizzata: Per integrare il database FileMaker con un sito Web personalizzato, utilizzare le tecnologie di Pubblicazione Web personalizzata disponibili con FileMaker Server. FileMaker Server, che ospita i database pubblicati, non richiede che FileMaker Pro sia installato o sia in esecuzione per rendere disponibile la Pubblicazione Web personalizzata.

Con la Pubblicazione Web personalizzata è possibile:

- Integrare un database in un sito Web
- Determinare il modo in cui gli utenti interagiscono con i dati
- Controllare il modo in cui i dati vengono visualizzati nel browser Web

FileMaker Server provvede due tecnologie di Pubblicazione Web personalizzata:

- Pubblicazione Web personalizzata con PHP: Utilizzare FileMaker API per PHP, che fornisce un'interfaccia PHP con il database FileMaker orientata agli oggetti, per integrare i dati FileMaker in un'applicazione Web PHP. Eseguendo direttamente la codifica delle pagine Web PHP, si ha il pieno controllo dell'interfaccia utente e di come l'utente interagisce con i dati.
- Pubblicazione Web personalizzata con XML: Utilizzare la pubblicazione di dati XML per scambiare dati FileMaker con altri siti Web e applicazioni. Utilizzando richieste HTTP URL con comandi e parametri di query specifici FileMaker, è possibile eseguire query su un database ospitato da FileMaker Server, scaricare i dati risultanti in formato XML e utilizzare i dati XML risultanti nel modo desiderato.

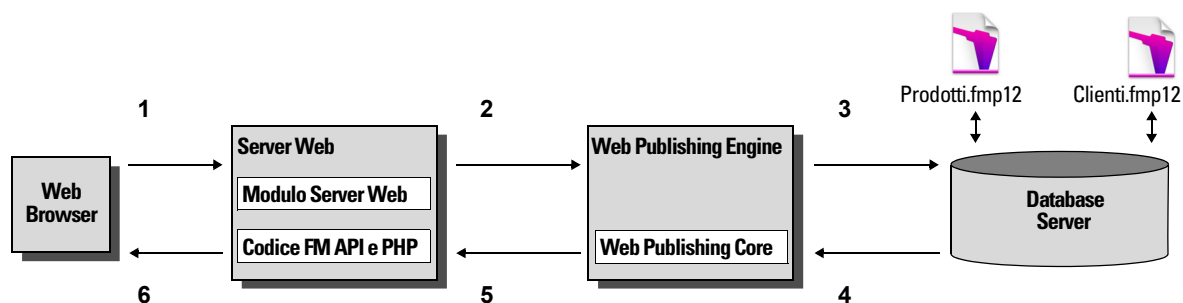
Motore per la Pubblicazione Web

Per supportare FileMaker WebDirect e Pubblicazione Web personalizzata, FileMaker Server utilizza un insieme di componenti software definiti *Motore per la Pubblicazione Web di FileMaker Server*. Il Motore per la Pubblicazione Web gestisce le interazioni tra il browser di un utente Web, il server Web e FileMaker Server.

Pubblicazione Web personalizzata con XML: Gli utenti Web accedono alla soluzione di Pubblicazione Web personalizzata facendo clic su un collegamento HREF oppure introducendo un URL (Uniform Resource Locator) che indica l'indirizzo del server Web e una richiesta di stringa query FileMaker. Il Motore per la Pubblicazione Web restituisce i dati XML specificati nella richiesta di stringa query.

Pubblicazione Web personalizzata con PHP: Quando un utente Web accede alla soluzione Pubblicazione Web personalizzata, PHP su FileMaker Server si collega con il Motore per la Pubblicazione Web e risponde attraverso FileMaker API per PHP.

Uso del Motore per la Pubblicazione Web di FileMaker Server per la Pubblicazione Web personalizzata



Modalità di elaborazione di una richiesta del Motore per la Pubblicazione Web

1. Una richiesta viene inviata da un'applicazione o da un browser Web al server Web.
2. Il server Web invia la richiesta attraverso il modulo Server Web di FileMaker al Motore per la Pubblicazione Web.
3. Il Motore per la Pubblicazione Web richiede i dati al database ospitato dal server Database.
4. FileMaker Server invia i dati FileMaker richiesti al Motore per la Pubblicazione Web.
5. Il Motore per la Pubblicazione Web converte i dati FileMaker per rispondere alla richiesta.
 - Per richieste PHP, il Motore per la Pubblicazione Web risponde alla richiesta API.
 - Per le richieste XML, il Motore per la Pubblicazione Web invia i dati XML direttamente al server Web.
6. Il server Web invia l'output al browser Web o al programma che lo richiedono.

Importante Quando si pubblicano dati sul Web si deve prestare attenzione alla sicurezza. Consultare le linee guida per la sicurezza contenute nella *Guida per l'utente di FileMaker Pro* disponibili in formato PDF sul sito <http://www.filemaker.com/it/support/index.html>.

Pubblicazione Web personalizzata con PHP

FileMaker API per PHP fornisce un'interfaccia PHP con il database FileMaker orientata agli oggetti. FileMaker API per PHP consente di accedere ai dati e alla logica memorizzati in un database FileMaker Pro e di pubblicarli sul web o di esportarli in altre applicazioni. API supporta anche comandi di ricerca complessi e composti per estrarre e filtrare i dati memorizzati nei database FileMaker Pro.

Progettato inizialmente come linguaggio di programmazione procedurale, PHP è stato migliorato come linguaggio di sviluppo web orientato agli oggetti. PHP fornisce le funzioni del linguaggio di programmazione per costruire virtualmente ogni tipo di logica in una pagina web. Ad esempio è possibile utilizzare costrutti di logica condizionale per controllare la creazione di pagine, il percorso dei dati o il flusso di lavoro. PHP fornisce anche l'amministrazione e la sicurezza del sito.

Pubblicazione Web personalizzata con XML

La Pubblicazione Web personalizzata XML di FileMaker consente di inviare richieste query a un database FileMaker Pro ospitato da FileMaker Server e visualizzare, modificare o trattare i dati risultanti. Utilizzando una richiesta HTTP con i parametri e i comandi di query appropriati, è possibile recuperare i dati FileMaker come documento XML. È possibile esportare i dati XML in altre applicazioni.

Confronto tra PHP e XML

Le seguenti sezioni forniscono alcune direttive per stabilire la soluzione più adeguata per un determinato sito.

Motivi per scegliere PHP

- PHP è un linguaggio di scripting procedurale più efficace e orientato agli oggetti e, nello stesso tempo, relativamente facile da apprendere. Offre molte risorse per l'apprendimento, lo sviluppo e il supporto.
- FileMaker API per PHP consente di accedere ai dati e alla logica memorizzati in un database FileMaker Pro e di pubblicarli sul web o di esportarli in altre applicazioni.
- PHP consente di utilizzare una logica condizionale per controllare la costruzione o il flusso delle pagine.
- PHP fornisce le funzioni del linguaggio di programmazione per costruire vari tipi di logica in una pagina web.
- PHP è uno dei linguaggi di scripting Web più conosciuti.
- PHP è un linguaggio open source, disponibile sul sito <http://php.net>.
- PHP consente di accedere ad una grande quantità di componenti di terze parti che possono essere integrati nelle proprie soluzioni.

Motivi per scegliere XML

- La sintassi dei parametri delle richieste XML FileMaker è progettata per interagire con i database, semplificando lo sviluppo della soluzione.
- XML è uno standard W3C.
- XML è un formato leggibile dall'uomo e dalla macchina che supporta Unicode, in grado di abilitare i dati per la comunicazione con ogni tipo di linguaggio scritto.

- XML è particolarmente adatto per la presentazione di record, liste e dati con struttura ad albero.
- È possibile utilizzare FMPXMLRESULT per accedere ai dati XML utilizzando la Pubblicazione Web personalizzata e per l'esportazione XML dai database FileMaker Pro.

Nota Per ulteriori informazioni sulla Pubblicazione Web personalizzata con XML, vedere *Pubblicazione Web personalizzata di FileMaker Server con XML*.

Capitolo 2

Informazioni su Pubblicazione Web personalizzata con PHP

Pubblicazione Web personalizzata con PHP consente di utilizzare il linguaggio di scripting PHP per integrare i dati dei database FileMaker con i formati delle pagine Web personalizzate. La Pubblicazione Web personalizzata con PHP fornisce FileMaker API per PHP che è una classe PHP creata da FileMaker che consente di accedere ai database ospitati da FileMaker Server. Questa classe PHP si collega al Motore per la Pubblicazione Web di FileMaker Server e rende disponibili i dati sul motore PHP del server Web.

Caratteristiche principali della Pubblicazione Web personalizzata con PHP

- Creare le applicazioni Web che utilizzano il linguaggio di scripting open source PHP. Utilizzare la versione di PHP 5 supportata da FileMaker Server o utilizzare la propria versione di PHP 5. (se si sceglie di utilizzare la propria versione di PHP, vedere “Installazione manuale di FileMaker API per PHP” a pagina 13.)
- Ospitare i database su FileMaker Server. FileMaker Pro non è necessario per la Pubblicazione Web personalizzata perché FileMaker Server ospita i database.
- Scrivere il codice PHP che consente di creare, eliminare, modificare e duplicare i record in un database FileMaker ospitato. Il codice può eseguire la convalida dei campi e dei record prima di salvare le modifiche nel database ospitato.
- Scrivere il codice PHP con cui si accede ai formati, ai portali, alle liste valori e ai campi correlati. Come in FileMaker Pro, l'accesso ai dati, ai formati e ai campi dipende dalle impostazioni degli account utente definite nei privilegi di accesso del database. Per il Motore per la Pubblicazione Web sono stati inoltre introdotti altri miglioramenti per la sicurezza. Vedere “Protezione dei database pubblicati” a pagina 16.
- Scrivere il codice PHP con cui vengono eseguiti script complessi, a più istruzioni. FileMaker supporta più di 65 istruzioni di script nella Pubblicazione Web personalizzata. Vedere “Script di FileMaker e Pubblicazione Web personalizzata” a pagina 23.
- Scrivere il codice PHP con cui vengono eseguiti script complessi.

Requisiti della Pubblicazione Web personalizzata

Questa sezione spiega quali sono i requisiti per sviluppare una soluzione di Pubblicazione Web personalizzata utilizzando PHP, ciò di cui gli utenti Web hanno bisogno per accedere alla soluzione di Pubblicazione Web personalizzata e l'impatto che può avere l'hosting di una soluzione di pubblicazione Web sul server.

Requisiti per la pubblicazione di un database mediante Pubblicazione Web personalizzata

Per pubblicare i database utilizzando la Pubblicazione Web personalizzata con PHP, è necessario:

- una distribuzione FileMaker Server, che comprende tre componenti.
 - un server Web, Microsoft IIS (Windows) o Apache (OS X). Il Modulo Server Web FileMaker è installato sul server Web.
 - il Motore per la Pubblicazione Web di FileMaker
 - Il server Database FileMaker
- PHP installato sul server Web. FileMaker Server può installare la versione supportata di PHP 5 o si può utilizzare la propria versione. La versione minima richiesta di PHP su OS X è PHP 5.3.15. La versione minima richiesta di PHP su Windows è PHP 5.3.27. Per informazioni su PHP, visitare il sito <http://php.net>. La versione PHP installata sul server Web deve supportare le funzioni cURL (client URL library). Per informazioni su cURL, visitare il sito <http://php.net/curl>.

Importante Quando si installa la versione di PHP 5 supportata da FileMaker Server, questa non viene visualizzata nello strumento Server Admin OS X: ciò è normale. Se per attivare PHP si utilizza lo strumento Server Admin OS X, viene disattivata la versione di PHP 5 supportata da FileMaker Server e attivata la propria versione di PHP.

- uno o più database FileMaker Pro ospitati da FileMaker Server.
- indirizzo IP o nome di dominio dell'host del server Web.
- un browser Web e la possibilità di accedere al server Web per sviluppare e controllare la soluzione di Pubblicazione Web personalizzata

Per ulteriori informazioni, vedere *Guida introduttiva di FileMaker Server*.

Requisiti degli utenti Web per accedere ad una soluzione di Pubblicazione Web personalizzata

Per accedere alla soluzione di Pubblicazione Web personalizzata che fa uso di PHP, gli utenti Web devono disporre di:

- un browser Web
- accesso a Internet o a una rete intranet e al server Web
- indirizzo IP o nome di dominio dell'host del server Web.

Se il database è protetto da password, gli utenti Web devono anche introdurre un nome utente e una password per l'account database.

Connessione a Internet o una rete intranet

Quando si pubblicano database su Internet o su una rete intranet, è necessario che sul computer host sia in esecuzione FileMaker Server e i database da condividere devono essere ospitati e disponibili. Inoltre:

- Pubblicare il database su un computer con una connessione continua a Internet o a un'intranet. È possibile pubblicare database anche in assenza di una connessione continua, ma in tal caso i database risulteranno disponibili agli utenti solo quando il computer che li ospita è connesso a Internet o alla rete intranet.
- Il computer host del server Web, che fa parte della distribuzione FileMaker Server, deve avere un nome di dominio o un indirizzo IP statico (permanente) dedicato. Se la connessione a Internet viene effettuata tramite un provider di servizi Internet (ISP), l'indirizzo IP potrebbe essere allocato in modo dinamico (ossia cambiare ad ogni connessione). L'utilizzo di un indirizzo IP dinamico rende più difficoltosa l'individuazione dei database da parte degli utenti. Se non si è certi del tipo di accesso di cui si dispone, contattare il provider di servizi Internet o l'amministratore della rete.

Installazione manuale di FileMaker API per PHP

Se si installa FileMaker Server, viene offerta l'opzione di installare la versione di PHP supportata da FileMaker (PHP 5). Se il motore PHP è già stato installato e configurato e si desidera aggiungere solo FileMaker API per PHP, installare manualmente la classe FileMaker API for PHP per renderla disponibile agli script PHP.

Se non viene installata la versione di PHP supportata da FileMaker, assicurarsi di eseguire le seguenti operazioni di configurazione sulla propria versione del motore PHP:

- Attivare il modulo cURL in `php.ini`.
- Selezionare la posizione di FileMaker API per PHP nella variabile `include_path` in `php.ini`.
- Se si accede a database che contengono date e ore, installare il pacchetto `pear date`. Per maggiori informazioni, vedere: <http://pear.php.net/package/date/>

Nota FileMaker Server è stato provato con la versione PHP 5.3.15 per OS X 10.8, con la versione PHP 5.4.17 per OS X 10.9, e con la versione PHP 5.3.27 per Windows. Per ottenere i migliori risultati, utilizzare la versione di PHP appropriata.

Per rendere accessibile FileMaker API per PHP agli script PHP

Se è stato installato FileMaker Server, il pacchetto FileMaker API per PHP è compreso sotto forma di file `.zip` nella seguente posizione:

- Per IIS (Windows):
`[unità]:\Programmi\FileMaker\FileMaker Server\Web Publishing\FM_API_for_PHP_Standalone.zip`
dove `[unità]` è l'unità in cui risiede il componente server Web della distribuzione FileMaker server.
- Per Apache (OS X):
`/Libreria/FileMaker Server/Web Publishing/FM_API_for_PHP_Standalone.zip`

Il file `FM_API_for_PHP_Standalone.zip` contiene un file chiamato `FileMaker.php` e una cartella chiamata `FileMaker`. Dezippare il file e copiare il file `FileMaker.php` e la cartella `FileMaker` in una di queste posizioni:

- la cartella in cui risiedono gli script PHP.
 - Per IIS (Windows) attraverso HTTP o HTTPS:
`[unità]:\Program Files\FileMaker\FileMaker Server\HTTPServer\Conf`
dove `[unità]` è l'unità in cui risiede il componente Motore per la Pubblicazione Web della distribuzione FileMaker Server.
 - Per Apache (OS X) attraverso HTTP: `/Library/FileMaker Server/HTTPServer/htdocs`
 - Per Apache (OS X) attraverso HTTPS:
`/Library/FileMaker Server/HTTPServer/htdocs/httpsRoot`
- una delle directory `include_path` dell'installazione PHP. La posizione predefinita per OS X è `/usr/lib/php`.

Operazioni successive

Di seguito sono riportati alcuni suggerimenti per iniziare a sviluppare le soluzioni di Pubblicazione Web personalizzata:

- Utilizzare Admin Console di FileMaker Server per attivare la Pubblicazione Web personalizzata. Vedere la Guida di FileMaker Server e la *Guida introduttiva di FileMaker Server*.
- In FileMaker Pro, aprire il database FileMaker che si desidera pubblicare e assicurarsi che il database abbia i privilegi estesi appropriati abilitati per la Pubblicazione Web personalizzata. Vedere “Attivazione della pubblicazione Web personalizzata con PHP per database” a pagina 15.
- Per imparare come accedere ai dati dei database FileMaker utilizzando FileMaker API per PHP, vedere capitolo 5, “Utilizzo di FileMaker API per PHP.”

Capitolo 3

Preparazione dei database per la Pubblicazione Web personalizzata

Prima di utilizzare la Pubblicazione Web personalizzata con un database, è necessario preparare il database e proteggerlo da accessi non autorizzati.

Attivazione della pubblicazione Web personalizzata con PHP per database

È necessario attivare la Pubblicazione Web personalizzata con PHP in ogni database da pubblicare. In caso contrario, gli utenti Web non possono utilizzare la Pubblicazione Web personalizzata per accedere al database anche se è ospitato da un server FileMaker configurato per supportare un Motore per la Pubblicazione Web.

Per attivare la Pubblicazione Web personalizzata per un database:

1. In FileMaker Pro, aprire il database da pubblicare utilizzando un account con un set di privilegi Accesso completo o Gestisci privilegi estesi.
2. Assegnare il privilegio esteso **fmphp** a uno o più set di privilegi per consentire la Pubblicazione Web personalizzata con PHP.
3. Assegnare il/i set di privilegi con il privilegio esteso Pubblicazione Web personalizzata agli account appropriati (ad esempio, account Admin e Ospite).

Importante Quando si definiscono i nomi account e le password per le soluzioni di Pubblicazione Web personalizzata, usare caratteri ASCII stampabili; ad esempio, **a-z**, **A-Z** e **0-9**. Per conferire maggiore sicurezza a nomi account e password, includere alcuni caratteri non alfanumerici come ad esempio un punto esclamativo (!) o un segno di percentuale (%). Non è possibile inserire i due punti (:). Per ulteriori informazioni sull'impostazione degli account, consultare la Guida di FileMaker Pro.

4. Utilizzando Admin Console di FileMaker Server, verificare che l'hosting sia configurato correttamente per il database e che questo sia accessibile a FileMaker server. Per ulteriori informazioni, vedere la Guida di FileMaker Server.

Nota Dato che la Pubblicazione Web personalizzata con PHP non utilizza sessioni di database persistenti, i riferimenti a un'origine dati ODBC esterna nel grafico delle relazioni di FileMaker Pro può limitare la funzionalità della soluzione PHP. Se il database accede ai dati da un'origine dati SQL esterna, non è possibile aggiornare i dati dei record di una tabella esterna.

Creazione dei formati per la Pubblicazione Web personalizzata con PHP

La Pubblicazione Web personalizzata con PHP non fornisce l'accesso diretto alla tabella per i dati di un database FileMaker Pro, ma utilizza i formati definiti nei database. Mentre non vi sono requisiti per creare un formato unico per la Pubblicazione Web personalizzata con PHP, la creazione di un formato specifico per una soluzione PHP può essere utile per vari motivi:

- Migliorare le prestazioni creando un formato limitato a campi, etichette, calcoli e portali che è necessario comprendere nella soluzione PHP.
- Semplificare il codice PHP riducendo l'elaborazione dei dati, dato che i record hanno meno campi.
- Separare il lavoro di progettazione dell'interfaccia dai dati in modo che possa essere ritagliata su misura per l'utente Web.

Protezione dei database pubblicati

La Pubblicazione Web personalizzata con PHP consente di limitare l'accesso ai database pubblicati. Si possono utilizzare questi metodi:

- Richiedere le password per gli account database utilizzati per la Pubblicazione Web personalizzata con PHP.
- Attivare il privilegio esteso Pubblicazione Web personalizzata con PHP solo nei set di privilegi a cui si desidera concedere l'accesso.
- Disattivare Pubblicazione Web personalizzata con PHP per un determinato database deselezionando il privilegio esteso fmpHP per tutti i set di privilegi di quel database. Vedere la Guida di FileMaker Pro
- Attivare o disattivare la Pubblicazione Web personalizzata per tutte le soluzioni di Pubblicazione Web personalizzata nel Motore per la Pubblicazione Web che utilizzano Admin Console di FileMaker Server. Vedere *Guida introduttiva di FileMaker Server* e Guida di FileMaker Server.
- Configurare il server Web per limitare gli indirizzi IP che possono accedere ai database attraverso il Motore per la Pubblicazione Web. Ad esempio, è possibile specificare che l'accesso ai database sia consentito esclusivamente agli utenti Web con l'indirizzo 192.168.100.101. Per informazioni sulla limitazione degli indirizzi IP, vedere la documentazione del server Web.

FileMaker Server supporta la crittografia per i dati scritti su disco e per i dati trasmessi ai client.

- Crittografare il database utilizzando la funzione Crittografia database di FileMaker Pro Advanced. La crittografia protegge il file di database FileMaker e i file temporanei scritti sul disco. Per ulteriori informazioni sulla crittografia dei database, vedere la *Guida per l'utente di FileMaker Pro*, la *Guida introduttiva di FileMaker Server* e la Guida di FileMaker Pro.
- Un database crittografato ospitato su FileMaker Server viene aperto utilizzando la Admin Console o l'interfaccia a riga di comando (CLI). L'amministratore di FileMaker Server apre il file utilizzando la sua password di crittografia del database, in modo da permettere ai client di FileMaker di utilizzare il database crittografato.
- Una volta che il database di FileMaker crittografato è stato aperto con la password di crittografia dall'amministratore di FileMaker Server, i client di FileMaker non hanno bisogno di una password di crittografia per accedere al database crittografato. Per maggiori informazioni su come aprire un database crittografato, vedere la Guida di FileMaker Server.
- Utilizzare la crittografia Secure Sockets Layer (SSL) per la comunicazione tra i browser degli utenti Web e il server Web. La crittografia SSL converte le informazioni scambiate tra i server e i client in dati incomprensibili utilizzando formule matematiche note come *algoritmi di crittografia*. Questi messaggi cifrati sono utilizzati per riconvertire le informazioni in dati comprensibili tramite le chiavi di cifratura. Le connessioni SSL sono realizzate mediante una connessione HTTPS. Una volta che sono configurate e funzionanti, non occorre alcun'azione dal lato del client. Per informazioni su come abilitare, configurare e mantenere le connessioni SSL, vedere la documentazione del server Web.

Per ulteriori informazioni sulla protezione del database, vedere la *Guida per l'utente di FileMaker Pro*, disponibile come file PDF all'indirizzo <http://www.filemaker.com/it/support/index.html>.

Accesso ad un database protetto

La Pubblicazione Web personalizzata con PHP consente di limitare l'accesso ai database pubblicati proteggendoli mediante password, crittografia e connessioni sicure. Quando un utente Web accede ad un database utilizzando una soluzione PHP, il codice PHP deve fornire le credenziali al database utilizzando FileMaker API per PHP. Se l'account Ospite per il database è disattivato o non dispone del privilegio di accesso esteso **fmphp**, FileMaker API per PHP restituisce un errore e il codice PHP deve fornire informazioni di accesso per l'utente.

Nelle esercitazioni di FileMaker API per PHP è riportato un esempio che illustra come utilizzare il metodo `setProperty()` per impostare username e password in un database protetto. Vedere "Esercitazioni di FileMaker API for PHP" a pagina 29.

I seguenti punti riassumono il processo che si verifica quando si utilizza la Pubblicazione Web personalizzata per accedere ad un database protetto da password:

- Se ad un account abilitato per la Pubblicazione Web personalizzata non è stata assegnata nessuna password, la soluzione PHP deve solo fornire il nome account.
- Se l'account Ospite viene disattivato, la soluzione PHP deve fornire nome account e password. La soluzione PHP può richiedere all'utente Web nome account e password o può memorizzare nome account e password nel codice PHP. L'account deve avere abilitato il privilegio esteso **fmphp**.

- Se l'account Ospite è abilitato e ha il privilegio esteso **fmphp** attivato:
 - La soluzione PHP non deve richiedere agli utenti Web nome account e password quando viene aperto un file. Tutti gli utenti Web accedono automaticamente con l'account Ospite e assumono i privilegi dell'account Ospite.
 - Il set di privilegi predefinito per gli account Ospite fornisce accesso di sola lettura. È possibile cambiare i privilegi predefiniti, compresi i privilegi estesi, per questo account. Vedere la Guida di FileMaker Pro.
- La soluzione PHP può utilizzare l'istruzione di script Nuovo accesso per permettere agli utenti di accedere utilizzando un account diverso (ad esempio, per passare da un account Ospite ad un account con più privilegi). Vedere la Guida di FileMaker Pro. Tuttavia, dato che le connessioni PHP non utilizzano sessioni di database persistenti, la soluzione PHP deve memorizzare nome account e password per utilizzarli in ogni richiesta successiva.

Nota Per impostazione predefinita, gli utenti Web non possono modificare le password da un browser Web. È possibile attivare questa funzione in un database utilizzando l'istruzione di script Modifica password che permette agli utenti Web di modificare le password dal browser. Vedere la Guida di FileMaker Pro.

Pubblicazione dei contenuti dei campi Contenitore sul Web

I contenuti di un campo Contenitore possono essere incorporati nel database, collegati come riferimento attraverso il relativo percorso o memorizzati esternamente.

Campi Contenitore incorporati in un database

Se nei campi Contenitore vengono memorizzati i file effettivi del database FileMaker, seguire questi passi per utilizzare gli oggetti campo Contenitore in una soluzione PHP:

- Utilizzare FileMaker API per PHP per definire l'oggetto del database (`$fm`) con le credenziali appropriate (nome account e password).

```
$fm = new FileMaker();
$fm->setProperty('database', $databaseName);
$fm->setProperty('username', $userName);
$fm->setProperty('password', $passWord);
```

- Utilizzare i tag HTML corretti per indicare il tipo di oggetto compatibile con il web contenuto nel campo Contenitore e creare una stringa URL che rappresenta il percorso file dell'attributo dell'origine del tag HTML.

```
<IMG src="img.php?-url=<?php echo urlencode($record->getField('Cover
Image')); ?>">
```

- Quindi utilizzare il metodo `getContainerData()` per recuperare l'oggetto del campo Contenitore.

```
echo $fm->getContainerData($_GET['-url']);
```

Nelle esercitazioni FileMaker API per PHP sono riportati altri esempi che illustrano come utilizzare i campi Contenitore. Vedere “Esercitazioni di FileMaker API for PHP” a pagina 29.

Note

- Il Motore per la Pubblicazione Web supporta il download progressivo di file audio (.mp3), file video (consigliati .mov, .mp4 e .avi) e file PDF per i contenitori interattivi. Ad esempio, un utente Web può iniziare a visualizzare un filmato anche se il relativo file non è ancora stato scaricato completamente. Per consentire il download progressivo, può essere necessario creare i file utilizzando opzioni che supportano lo streaming o che ottimizzano la visualizzazione sul Web. Ad esempio, creare file PDF utilizzando l'opzione "Ottimizza per la visualizzazione sul Web".
- Quando non è selezionata l'impostazione di FileMaker Server **Attiva connessioni sicure**, le connessioni utilizzate da FileMaker Server per trasmettere i dati non sono crittografate durante la trasmissione.
 - I client FileMaker visualizzano i dati Contenitore interattivi con un ritardo minimo.
 - FileMaker Server decrittta i dati dei campi Contenitore in una cartella di cache sul server quando un client FileMaker Pro, FileMaker Go o dal Web richiede tali dati. I dati possono rimanere decrittati nella cartella di cache sul server per due ore, limite entro il quale FileMaker Server svuota periodicamente la cache. I dati non vengono memorizzati nella cache locale del client.

- Quando è selezionata l'impostazione di FileMaker Server **Attiva connessioni sicure**, FileMaker Server trasmette i dati utilizzando delle connessioni sicure. Prima che l'utente possa interagire con i dati Contenitore, questi devono essere scaricati completamente dai clienti FileMaker. La sicurezza dei dati è la stessa che avrebbero se fossero contenuti in una soluzione di database in locale, poiché non vengono creati file di cache e durante la trasmissione i dati sono crittografati.

Il server del database deve essere arrestato e riavviato quando viene modificata l'impostazione **Attiva connessioni sicure** affinché le modifiche abbiano effetto.

Campi Contenitore con file di riferimento

Se in un campo Contenitore viene memorizzato il riferimento a un file, è possibile utilizzare il metodo `getContainerData()` per recuperare gli oggetti del campo Contenitore dal database nel codice PHP oppure è possibile utilizzare il metodo `getContainerDataURL()` per recuperare un URL completamente qualificato per l'oggetto del campo Contenitore.

È anche necessario seguire questi passi per pubblicare i file a cui si fa riferimento utilizzando il Motore per la Pubblicazione Web:

1. Memorizzare i file oggetto Contenitore nella cartella Web all'interno della cartella FileMaker Pro.
2. In FileMaker Pro, inserire gli oggetti nel campo Contenitore e selezionare l'opzione **Memorizza solo un riferimento al file**.
3. Copiare o spostare i file dell'oggetto a cui si fa riferimento nella cartella Web nella stessa posizione relativa nella cartella seguente del server Web.
 - Per IIS (Windows) attraverso HTTP o HTTPS:
`[unità]:\Programmi\FileMaker\FileMaker Server\HTTPServer\Conf` dove [unità] è l'unità in cui risiede il componente del motore della Pubblicazione Web della distribuzione FileMaker server.
 - Per Apache (OS X) attraverso HTTP: `/Library/FileMaker Server/HTTPServer/htdocs`
 - Per Apache (OS X) attraverso HTTPS: `/Library/FileMaker Server/HTTPServer/htdocs/httpsRoot`

Note

- Per gli oggetti Contenitore memorizzati come riferimenti ai file, il server Web deve essere configurato per supportare i tipi MIME (Multipurpose Internet Mail Extensions) per i file da gestire, come i filmati. Il server Web determina il supporto per i tipi MIME (Multipurpose Internet Mail Extensions) correnti registrati per Internet. Il Motore per la Pubblicazione Web non cambia il supporto del server Web per MIME. Per ulteriori informazioni, vedere la documentazione del server Web.
- Tutti i filmati QuickTime vengono memorizzati in un campo Contenitore come riferimento.

Campi Contenitore con dati memorizzati esternamente

Se in un campo Contenitore vengono memorizzati oggetti esternamente — ovvero se si era selezionato **Memorizza dati contenitore esternamente** nella finestra di dialogo Opzioni campo — il codice PHP deve utilizzare il metodo `getContainerDataURL()` per recuperare un URL completamente qualificato per l'oggetto del campo Contenitore.

Utilizzare FileMaker API per PHP per definire l'oggetto del database con le credenziali appropriate (nome account e password), quindi usare il metodo `getContainerDataURL()`.

Esempio di immagini inserite utilizzando il tag HTML `img`

```
$fm=new FileMaker($database, $hostspec, $user, $password);
$findCommand = $fm->newFindCommand($layout);
$findCommand->addFindCriterion('type', 'png');
$result = $findCommand->execute();
$records = $result->getRecords();
foreach ($records as $record) {
    echo $record->getField('container').'<br>';
    // Per le immagini, utilizzare il tag HTML img
    echo '';
    break;
}
```

Esempio di dati incorporati utilizzando il tag HTML `embed`

```
$fm=new FileMaker($database, $hostspec, $user, $password);
$findCommand = $fm->newFindCommand($layout);
$findCommand->addFindCriterion('type', 'pdf');
$result = $findCommand->execute();
$records = $result->getRecords();
foreach ($records as $record) {
    echo $record->getField('container').'<br>';
    // Per i filmati e i file PDF, utilizzare il tag HTML embed
    //echo '<embed src="'. $fm->
        getContainerDataURL($record->getField('container')) .'">';
    break;
}
```

Caricamento di dati del campo Contenitore su FileMaker Server

Se si utilizza FileMaker Pro per caricare un database, i dati del campo Contenitore memorizzati esternamente vengono memorizzati in FileMaker Server come parte del processo. Vedere la Guida di FileMaker Pro per informazioni sul trasferimento dei file di database in FileMaker Server.

Se si carica manualmente un database che utilizza un campo Contenitore con oggetti memorizzati esternamente, è necessario seguire questi passi per pubblicare gli oggetti Contenitore memorizzati esternamente utilizzando il Motore per la Pubblicazione Web.

Per caricare un database manualmente:

1. Mettere il file di database nella posizione appropriata sul server. Inserire nelle seguenti cartelle i file dei database FileMaker Pro che si desidera aprire con FileMaker Server — o tasti di scelta rapida (Windows) o alias (OS X) riferiti a quei file:
 - Windows: [unità]:\Programmi\FileMaker\FileMaker Server\Data\Databases\
dove [unità] è l'unità principale da cui viene avviato il sistema.
 - OS X: /Libreria/FileMaker Server/Data/Database/
O è possibile inserire i file in una cartella aggiuntiva per i database (opzionale).
2. Nella cartella in cui è stato inserito il database, creare una cartella chiamata Dati_RC_FMS, se non esiste già.
3. Nella cartella Dati_RC_FMS, creare una cartella con un nome che corrisponda a quello del database. Ad esempio, se il database si chiama Clienti, creare una cartella Clienti. Inserire gli oggetti memorizzati esternamente nella nuova cartella creata.

Nota Quando i database sono ospitati su FileMaker Server, non è possibile per più database condividere una cartella di oggetti Contenitore comune. Gli oggetti Contenitore di ciascun database devono essere in una cartella con lo stesso nome del database.

4. Per i file che saranno condivisi da OS X, modificarli in modo che appartengano al gruppo **fmsadmin**.

Per ulteriori informazioni sul caricamento manuale dei database, vedere la Guida di FileMaker Server.

Note

- Il Motore per la Pubblicazione Web supporta il download progressivo di file audio (.mp3), file video (consigliati .mov, .mp4 e .avi) e file PDF per i contenitori interattivi. Ad esempio, un utente Web può iniziare a visualizzare un filmato anche se il relativo file non è ancora stato scaricato completamente. Per consentire il download progressivo, può essere necessario creare i file utilizzando opzioni che supportano lo streaming o che ottimizzano la visualizzazione sul Web. Ad esempio, creare file PDF utilizzando l'opzione "Ottimizza per la visualizzazione sul Web".
- Quando non è selezionata l'impostazione di FileMaker Server **Attiva connessioni sicure**, le connessioni utilizzate da FileMaker Server per trasmettere i dati non sono crittografate durante la trasmissione.
 - I client FileMaker visualizzano i dati Contenitore interattivi con un ritardo minimo.
 - FileMaker Server decrittifica i dati dei campi Contenitore in una cartella di cache sul server quando un client FileMaker Pro, FileMaker Go o dal Web richiede tali dati. I dati possono rimanere decrittati nella cartella di cache sul server per due ore, limite entro il quale FileMaker Server svuota periodicamente la cache. I dati non vengono memorizzati nella cache locale del client.

- Quando è selezionata l'impostazione di FileMaker Server **Attiva connessioni sicure**, FileMaker Server trasmette i dati utilizzando delle connessioni sicure. Prima che l'utente possa interagire con i dati Contenitore, questi devono essere scaricati completamente dai clienti FileMaker. La sicurezza dei dati è la stessa che avrebbero se fossero contenuti in una soluzione di database in locale, poiché non vengono creati file di cache e durante la trasmissione i dati sono crittografati.

Il server del database deve essere arrestato e riavviato quando viene modificata l'impostazione **Attiva connessioni sicure** affinché le modifiche abbiano effetto.

Visualizzazione dei dati dei campi Contenitore da parte degli utenti Web

Se un database viene pubblicato utilizzando il Motore per la Pubblicazione Web, agli oggetti del campo Contenitore vengono applicate le seguenti limitazioni:

- Gli utenti Web non possono modificare o fare aggiunte al contenuto di un campo Contenitore. Gli utenti Web non possono utilizzare campi Contenitore per caricare oggetti nel database.
- Per i database che utilizzano un campo Contenitore con le miniature abilitate, il Motore per la Pubblicazione Web scarica l'intero file e non una miniatura.

Script di FileMaker e Pubblicazione Web personalizzata

La funzione Gestisci script in FileMaker Pro consente di combinare varie attività e di automatizzare quelle eseguite più di frequente. Quando sono utilizzati con Pubblicazione Web personalizzata, gli script di FileMaker permettono agli utenti Web di eseguire una serie di operazioni. Gli script FileMaker permettono anche di eseguire operazioni che non sono supportate in altri modi, come utilizzare l'istruzione di script Modifica password per permettere agli utenti Web di modificare le password da un browser.

FileMaker supporta più di 65 istruzioni di script nella Pubblicazione Web personalizzata. Per vedere le istruzioni di script non supportate, selezionare **Pubblicazione Web personalizzata** dall'elenco **Mostra compatibilità** nella finestra Modifica script in FileMaker Pro. Le istruzioni di script di colore grigio non sono supportate per la Pubblicazione Web personalizzata. Per informazioni sulla creazione degli script, vedere la Guida di FileMaker Pro.

Suggerimenti e considerazioni sugli script

Anche se molte istruzioni di script funzionano allo stesso modo sul Web, alcune hanno un funzionamento diverso. Vedere "Comportamento degli script nelle soluzioni di Pubblicazione Web personalizzata" a pagina 25. Prima di condividere il database, testare tutti gli script che saranno eseguiti da un browser Web. Effettuare l'accesso con account diversi per assicurarsi che questi funzionino nel modo previsto per tutti i client.

Tenere presente quanto segue:

- Usare account e privilegi per limitare il set di script eseguibili da un utente Web. Verificare che gli script contengano soltanto istruzioni di script compatibili con il Web e offrire accesso soltanto agli script che possono essere usati da un browser Web.
- Considerare inoltre gli effetti collaterali degli script che eseguono una combinazione di istruzioni controllate dai privilegi di accesso. Ad esempio, se uno script contiene un'istruzione per eliminare record e l'utente Web non accede con un account che autorizza tale operazione, essa non verrà eseguita. Tuttavia, l'esecuzione dello script potrebbe continuare, provocando risultati imprevisti.

- Nella finestra Modifica script, selezionare **Esegui script con privilegi di accesso completo** per permettere agli script di eseguire operazioni non eseguibili da parte dei singoli utenti. Ad esempio, è possibile impedire agli utenti di eliminare record restringendo account e privilegi, ma nello stesso tempo è possibile consentire loro di eseguire uno script che elimini determinati tipi di record alle condizioni predefinite all'interno di uno script.
- Se gli script contengono istruzioni non supportate, come ad esempio istruzioni non compatibili con il Web, usare l'istruzione di script **Consenti interruzione utente** per determinare il modo in cui gestire le istruzioni successive:
 - Se l'opzione **Consenti interruzione utente** è attivata, le istruzioni di script non supportate interrompono l'esecuzione dello script.
 - Se l'opzione **Consenti interruzione utente** è disattivata, le istruzioni di script non supportate vengono saltate e l'esecuzione dello script continua.
 - Se questa istruzione di script non è inclusa, gli script vengono eseguiti come se la funzione fosse attivata, facendo in modo che le istruzioni di script non supportate interrompano gli script.
- Alcuni script che usano un'istruzione di un client FileMaker Pro potrebbero richiedere un'ulteriore istruzione di script Salva record/richieste per salvare i dati sull'host. Poiché gli utenti Web non hanno una connessione diretta con l'host, non ricevono notifica delle variazioni dei dati. Ad esempio, le funzioni come le liste valori condizionali non sono così immediate per gli utenti del Web perché i dati devono essere salvati sull'host prima che i loro effetti vengano visualizzati nel campo lista valori.
- Gli script che modificano i dati devono comprendere l'istruzione Salva Record/Richieste, perché le modifiche apportate ai dati non sono visibili nel browser finché i dati non vengono salvati o “inviati” al server. Alcuni esempi sono le istruzioni di script Taglia, Copia, Incolla, ecc. Molte azioni ad istruzione singola devono essere convertite in script per includere l'istruzione Salva Record/Richieste. Per creare script da eseguire da un browser Web includere l'istruzione Salva Record/Richieste alla fine dello script per assicurarsi che tutte le modifiche vengano salvate.
- Per creare script condizionali basati sul tipo di client, usare la funzione Get (VersioneApplicazione). Se il valore restituito comprende “Motore per la Pubblicazione Web” significa che l'utente sta accedendo al database con la Pubblicazione Web personalizzata. Per ulteriori informazioni sulle funzioni, consultare la Guida di FileMaker Pro.
- Dopo aver convertito i file, aprire tutti gli script che gli utenti Web potrebbero usare e selezionare **Pubblicazione Web** dall'elenco Indica compatibilità nella finestra Modifica script per assicurarsi che lo script venga eseguito correttamente con Pubblicazione Web Immediata.

Comportamento degli script nelle soluzioni di Pubblicazione Web personalizzata

Le seguenti istruzioni di script funzionano in modo diverso sul Web rispetto a FileMaker Pro. Per informazioni su tutte le istruzioni di script, vedere la Guida di FileMaker Pro.

Istruzione di script	Comportamento nelle soluzioni di Pubblicazione Web personalizzata
Esegui script	Gli script non possono essere eseguiti in altri file, a meno che i file siano ospitati su FileMaker Server e che la Pubblicazione Web personalizzata sia attivata negli altri file.
Esci dall'applicazione	Disconnette gli utenti Web, chiude tutte le finestre ma non chiude l'applicazione del browser Web.
Consenti interruzione utente	Determina il modo in cui vengono gestite le istruzioni di script non supportate. Viene attivata per bloccare l'esecuzione degli script, viene disattivata per saltare le istruzioni non supportate. Vedere "Suggerimenti e considerazioni sugli script" a pagina 23 per ulteriori dettagli. Nota Gli utenti Web non possono interrompere gli script della Pubblicazione Web personalizzata, ma questa opzione fa in modo che le istruzioni di script non supportate interrompano l'esecuzione dello script.
Imposta cattura errori	Questa istruzione è sempre attivata nella Pubblicazione Web personalizzata. Gli utenti Web non possono interrompere gli script della Pubblicazione Web personalizzata.
Pausa/continua lo script	Anche se queste istruzioni di script sono supportate nella Pubblicazione Web personalizzata, si consiglia di non utilizzarle. Quando si esegue un'istruzione di pausa, l'esecuzione dello script si arresta. Solo gli script che contengono l'istruzione per riprendere l'esecuzione possono ripartire. Se lo script rimane nello stato di pausa fino alla fine della sessione, esso non viene completato.
Ordina i record	Salvare un criterio di ordinamento con l'istruzione di script Ordina record da eseguire nella Pubblicazione Web personalizzata.
Apri URL	Questa istruzione di script non produce effetti in una soluzione di Pubblicazione Web personalizzata.
Vai al campo	Non è possibile usare Vai al campo per attivare un particolare campo nel browser Web, ma è possibile usare questa istruzione di script insieme con altre per eseguire delle operazioni. Ad esempio è possibile passare in un campo, copiarne il contenuto, passare ad un altro campo e incollare il valore. Per vederne gli effetti sul browser è necessario salvare il record con l'istruzione di script Salva record.
Salva Record/Richieste	Invia il record al database.

Trigger degli script e soluzioni di Pubblicazione Web personalizzata

In FileMaker Pro, i trigger degli script possono essere attivati sia dagli script che dalle operazioni dell'utente (come ad esempio facendo clic su un campo). Nella Pubblicazione Web personalizzata, invece, solo gli script possono attivare i trigger degli script. Ad esempio, se un utente della Pubblicazione Web personalizzata fa clic su un campo che ha un trigger di script SuEntrataOggetto, il trigger non è attivato. Tuttavia, se uno script fa in modo che il fuoco si sposti sul campo, il trigger di script SuEntrataOggetto viene attivato. Per ulteriori informazioni sui trigger degli script, vedere la Guida di FileMaker Pro.

Nota Se si desidera che uno script venga eseguito all'apertura di un file, è necessario utilizzare il trigger di script SuAperturaPrimaFinestra. Analogamente, per specificare che si desidera che uno script venga eseguito alla chiusura di un file, è necessario utilizzare il trigger di script SuChiusuraUltimaFinestra.

Capitolo 4

Pubblicazione Web personalizzata con PHP

FileMaker API per PHP è un supporto per l'integrazione dei dati dei database FileMaker Pro in soluzioni PHP. Questo capitolo descrive il modo in cui PHP lavora con il Motore per la Pubblicazione Web personalizzata di FileMaker Server. Per ulteriori informazioni su FileMaker API per PHP, vedere capitolo 5, "Utilizzo di FileMaker API per PHP."

Utilizzo del Motore per la Pubblicazione Web con soluzioni PHP

FileMaker Server è costituito da tre componenti: un server Web, il Motore per la Pubblicazione Web e il server Database. (Questi componenti possono essere distribuiti su una o due macchine. Per ulteriori informazioni vedere la *Guida introduttiva di FileMaker Server*). FileMaker Server ospita la soluzione PHP quando i file PHP vengono collocati sul server Web in cui è installato il motore PHP.

- Quando un utente Web apre una soluzione PHP, il server Web invia la richiesta al motore PHP che elabora il codice PHP.
- Se il codice PHP contiene dei richiami a FileMaker API per PHP, questi vengono interpretati e inviati come richieste al Motore per la Pubblicazione Web.
- Il Motore per la Pubblicazione Web richiede i dati ai database ospitati sul server Database.
- Il server Database invia i dati richiesti al Motore per la Pubblicazione Web.
- Il Motore per la Pubblicazione Web invia i dati al motore PHP sul server Web in risposta alla chiamata API.
- La soluzione PHP elabora i dati e li visualizza per l'utente Web.

Operazioni generali per la Pubblicazione Web personalizzata con PHP

Di seguito vi è un riepilogo dei passi da eseguire per utilizzare la Pubblicazione Web personalizzata con PHP:

1. Nella Admin Console, assicurarsi che **Attiva Pubblicazione PHP** sia selezionato. Vedere *Guida introduttiva di FileMaker Server*.
2. Nella Admin Console, selezionare il riquadro **Database** e assicurarsi che ogni database FileMaker che si sta pubblicando abbia il privilegio esteso **fmphp** attivato per la Pubblicazione Web personalizzata con PHP.

Se necessario, utilizzare FileMaker Pro per attivare la Pubblicazione Web personalizzata per un database. Vedere capitolo 3, "Preparazione dei database per la Pubblicazione Web personalizzata."

Nota Assicurarsi di utilizzare set di privilegi del database FileMaker equivalenti quando si sviluppano soluzioni PHP che verranno consegnate all'utente finale. In caso contrario, si potrebbe avere l'accesso a formati e funzioni nel database FileMaker che non sarà disponibile all'utente finale, provocando un comportamento incoerente.

3. Utilizzare gli strumenti di authoring PHP per creare la propria soluzione PHP, incorporando le funzioni FileMaker API nel codice PHP per accedere ai dati FileMaker. Vedere capitolo 5, "Utilizzo di FileMaker API per PHP."

4. Copiare o spostare la struttura della directory e i file del sito nella seguente cartella sul server Web.
 - Per IIS (Windows) attraverso HTTP o HTTPS:
`[unità]:\Programmi\FileMaker\FileMaker Server\Examples\PHP\API Examples` dove [unità] è l'unità in cui risiede il componente server Web della distribuzione FileMaker server.
 - Per Apache (OS X) attraverso HTTP: `/Library/FileMaker Server/HTTPServer/htdocs`
 - Per Apache (OS X) attraverso HTTPS:
`/Library/FileMaker Server/HTTPServer/htdocs/httpsRoot`
5. Se un campo Contenitore del database contiene un riferimento al file anziché il file reale, l'oggetto Contenitore di riferimento deve essere memorizzato nella cartella Web di FileMaker Pro quando il record viene creato o modificato. Copiare o spostare l'oggetto in una cartella con la stessa posizione relativa nella cartella principale del software del server Web.
Vedere “Pubblicazione dei contenuti dei campi Contenitore sul Web” a pagina 19.
6. Assicurarsi che i meccanismi della sicurezza per il sito o per il programma siano a posto.
7. Controllare il sito utilizzando gli stessi account e gli stessi privilegi definiti per gli utenti Web.
8. Rendere il sito disponibile e noto agli utenti. L'URL che l'utente Web inserisce ha il seguente formato:

```
http://<server>/<percorso_sito>
```

- <server> è la macchina in cui FileMaker Server risiede
- <percorso_sito> è il percorso relativo alla homepage del sito, determinato dalla struttura di directory utilizzata nel passo indicato sopra passo 4.

Ad esempio se il server Web è 192.168.123.101 e la homepage del sito sul server Web è `c:\inetpub\wwwroot\customers\index.php`, l'utente Web deve inserire l'URL:

```
http://192.168.123.101/customers/index.php
```

Nota PHP 5 utilizza la codifica Latin-1 (ISO-8859-1). FileMaker Server restituisce dati Unicode (UTF-8). Utilizzare Admin Console di FileMaker Server per indicare la codifica caratteri predefinita per il sito. Per i siti PHP, è possibile indicare UTF-8 o ISO-8859-1; si raccomanda di impostare UTF-8. Indicare la stessa impostazione per l'attributo `charset` nella sezione `<HEAD>` dei file PHP del sito.

Per informazioni sulla distribuzione e sull'utilizzo di una soluzione PHP, vedere capitolo 6, “Preparazione, test e monitoraggio di un sito.”

Capitolo 5

Utilizzo di FileMaker API per PHP

FileMaker API per PHP implementa una classe PHP—la classe FileMaker—che fornisce un'interfaccia orientata agli oggetti per i database FileMaker. FileMaker API per PHP consente di accedere ai dati e alla logica memorizzati nei database FileMaker Pro e di pubblicarli sul Web o di esportarli in altre applicazioni.

FileMaker API per PHP permette al codice PHP di eseguire lo stesso tipo di funzioni già disponibili nei database FileMaker Pro:

- creare, eliminare, modificare e duplicare i record
- eseguire richieste di ricerca
- convalidare campi e record
- utilizzare formati
- eseguire script FileMaker
- visualizzare portali e record correlati
- utilizzare liste valori

Questo capitolo descrive come utilizzare gli oggetti della classe FileMaker e i metodi per aggiungere queste funzioni comuni a una soluzione PHP. Questo capitolo non tratta tutti gli argomenti relativi a FileMaker API per PHP, ma introduce le nozioni ed i metodi più importanti.

Dove ottenere informazioni aggiuntive

Per ulteriori informazioni su FileMaker API per PHP, consultare le seguenti risorse.

Se il motore PHP è già stato installato e configurato e si desidera aggiungere solo FileMaker API per PHP, vedere “Installazione manuale di FileMaker API per PHP” a pagina 13.

Informazioni di riferimento a FileMaker API per PHP

Installando FileMaker API per PHP, si possono trovare informazioni di riferimento sul componente server Web della distribuzione FileMaker Server.

- Per IIS (Windows):
[unità]:\Programmi\FileMaker\FileMaker Server\Documentation\
PHP API Documentation\index.html
dove [unità] è l'unità in cui risiede il componente server Web della distribuzione FileMaker server.
- Per Apache (OS X): /Libreria/FileMaker Server/Documentation/PHP API
Documentation/index.html

Esercitazioni di FileMaker API for PHP

Installando FileMaker API per PHP, si possono trovare esercitazioni sul componente server Web della distribuzione FileMaker Server.

- Per IIS (Windows): [unità]:\Programmi\FileMaker\FileMaker Server\Examples\PHP\Tutorial dove [unità] è l'unità in cui risiede il componente server Web della distribuzione FileMaker server.
- Per Apache (OS X): /Libreria/FileMaker Server/Examples/PHP/Tutorial

Per ospitare i file delle esercitazioni PHP, copiarli nella cartella principale del server Web.

Esempi FileMaker API per PHP

Installando FileMaker API per PHP, si possono trovare altri esempi sul componente server Web della distribuzione FileMaker Server.

- Per IIS (Windows): [unità]:\Programmi\FileMaker\FileMaker Server\Examples\PHP\API Examples dove [unità] è l'unità in cui risiede il componente server Web della distribuzione FileMaker server.
- Per Apache (OS X): /Libreria/FileMaker Server/Examples/PHP/API Examples

Per ospitare i file degli esempi API, copiarli nella cartella principale del server Web.

Utilizzo della classe FileMaker

Per utilizzare la classe FileMaker nella soluzione PHP, aggiungere al codice PHP la seguente istruzione:

```
require_once ('FileMaker.php');
```

Oggetti classe FileMaker

La classe FileMaker definisce gli oggetti della classe utilizzata per recuperare i dati dai database FileMaker Pro.

Oggetto classe	Utilizzare l'oggetto per
Database FileMaker	Definire le proprietà del database Collegarsi a un database FileMaker Pro Ottenere informazioni su FileMaker API per PHP
Comando	Creare comandi per aggiungere, eliminare, duplicare e modificare i record, eseguire richieste di ricerca ed eseguire script.
Formato	Lavorare con i formati di database
Numero di record	Lavorare con i dati dei record
Campo	Lavorare con i dati dei campi
Set correlato	Lavorare con i record del portale
Risultato	Elaborare i record restituiti da una richiesta di ricerca
Errore	Controllare se si è verificato un errore Elaborare eventuali errori

Oggetti di comando FileMaker

La classe FileMaker definisce un oggetto di comando base utilizzato per creare l'istanza di un comando specifico e indicare i parametri del comando. Per eseguire il comando, richiamare il metodo `execute()`.

La classe FileMaker definisce i seguenti comandi specifici:

- Comando Add
- Comando Compound Find
- Comando Delete
- Comando Duplicate
- Comando Edit
- Comando Find, comando Find All, comando Find Any
- Comando Find Request, che ottiene delle aggiunte al comando di ricerche composte
- Comando Esegui script

Questi comandi sono descritti nel dettaglio nelle seguenti sezioni:

- “Utilizzo dei record” a pagina 31
- “Esecuzione di script di FileMaker” a pagina 32
- “Esecuzione di richieste di ricerca” a pagina 38

Collegamento a un database FileMaker

La classe FileMaker definisce un oggetto del database sul quale viene creata un'istanza per collegarsi a un server o a un database. Definire le proprietà dell'oggetto con il costruttore della classe o richiamando il metodo `setProperty()`.

Esempio: Collegamento ad un server per ottenere una lista di database

```
$fm = new FileMaker();  
$databases = $fm->listDatabases();
```

Esempio: Collegamento ad un database specifico su un server

Le proprietà di username e password determinano il set di privilegi per la connessione.

```
$fm = new FileMaker();  
$fm->setProperty('database', 'questionnaire');  
$fm->setProperty('hostspec', 'http://192.168.100.110');  
$fm->setProperty('username', 'web');  
$fm->setProperty('password', 'web');
```

Nota La proprietà `hostspec` è preimpostata sul valore `http://localhost`. Se il motore PHP è in attivo sulla stessa macchina del componente server Web della distribuzione FileMaker Server, non è necessario indicare la proprietà `hostspec`. Se il motore PHP è attivo su una macchina diversa, utilizzare la proprietà `hostspec` per indicare la posizione del componente server Web della distribuzione FileMaker Server.

Utilizzo dei record

La classe FileMaker definisce un oggetto record sul quale viene creata un'istanza per utilizzare i record. L'istanza di un oggetto record rappresenta un record di un database FileMaker Pro. Utilizzare un oggetto record con i comandi Add, Delete, Duplicate e Edit per modificare i dati del record. I comandi di ricerca —Find, Find All, Find Any e Compound Find—restituiscono una matrice di oggetti record.

Creazione di un record

Esistono due modi per creare un record:

- Utilizzare il metodo `createRecord()`, specificando un nome formato e, opzionalmente, una matrice dei valori di campo. È inoltre possibile impostare i valori singolarmente nel nuovo oggetto record.

Il metodo `createRecord()` non consente di salvare il nuovo record nel database. Per salvare il record nel database, utilizzare il metodo `commit()`.

Ad esempio:

```
$rec = $fm->createRecord('Form View', $values);  
$result = $rec->commit();
```

- Utilizzare il comando Add. Utilizzare il metodo `newAddCommand()` per creare un oggetto FileMaker_Command_Add, indicando il nome del formato e una matrice con i dati del record. Per salvare il record nel database, utilizzare il metodo `execute()`.

Ad esempio:

```
$newAdd = $fm->newAddCommand('Respondent', $respondent_data);  
$result = $newAdd->execute();
```

Duplicazione di un record

Duplicare un record esistente utilizzando il comando Duplicate. Utilizzare il metodo `newDuplicateCommand()` per creare un oggetto FileMaker_Command_Duplicate, indicando il nome del formato e l'ID del record da duplicare. Quindi duplicare il record utilizzando il metodo `execute()`.

Esempio

```
$newDuplicate = $fm->newDuplicateCommand('Respondent', $rec_ID);  
$result = $newDuplicate->execute();
```

Modifica di un record

Esistono due modi per modificare un record:

- Utilizzando il comando Edit. Utilizzare il metodo `newEditCommand()` per creare un oggetto FileMaker_Command_Edit, indicando il nome del formato, l'ID del record da modificare e una matrice di valori da aggiornare. Quindi modificare il record utilizzando il metodo `execute()`.

Ad esempio:

```
$newEdit = $fm->newEditCommand('Respondent', $rec_ID, $respondent_data);  
$result = $newEdit->execute();
```

- Utilizzo di un oggetto record. Recuperare un record dal database, cambiare i valori campo e modificare il record utilizzando il metodo `commit()`.

Ad esempio:

```
$rec = $fm->getRecordById('Form View', $rec_ID);  
$rec->setField('Name', $nameEntered);  
$result = $rec->commit();
```

Eliminazione di un record

Esistono due modi per eliminare un record:

- Recuperare un record dal database e utilizzare il metodo `delete()`.

Ad esempio:

```
$rec = $fm->getRecordById('Form View', $rec_ID);  
$rec->delete();
```

- Eliminare un record esistente utilizzando il comando Delete. Utilizzare il metodo `newDeleteCommand()` per creare un oggetto `FileMaker_Command_Delete`, indicando il nome del formato e l'ID del record da eliminare. Quindi eliminare il record utilizzando il metodo `execute()`.

Ad esempio:

```
$newDelete = $fm->newDeleteCommand('Respondent', $rec_ID);  
$result = $newDelete->execute();
```

Esecuzione di script di FileMaker

Uno script FileMaker è un set denominato di istruzioni di script. La classe FileMaker definisce vari metodi che consentono di utilizzare gli script FileMaker definiti in un database FileMaker Pro. Per informazioni sulle istruzioni di script compatibili con il Web (istruzioni di script che possono essere eseguite in una soluzione Web), vedere “Script di FileMaker e Pubblicazione Web personalizzata” a pagina 23.

Recupero della lista di script disponibili

Utilizzare il metodo `listScripts()` per ottenere una lista degli script disponibili dal database al momento collegato. Il metodo `listScripts()` restituisce una matrice di script che possono essere eseguiti con lo username e la password indicati quando è stato definito il collegamento con il database. (vedere “Collegamento a un database FileMaker” a pagina 30.).

Esempio

```
$scripts = $fm->listScripts();
```


Esecuzione di uno script FileMaker

Utilizzare il metodo `newPerformScriptCommand()` per creare un oggetto `FileMaker_Command_PerformScript`, specificando il formato, il nome dello script ed eventuali parametri di script. Quindi, eseguire lo script utilizzando il metodo `execute()`.

Esempio

```
$newPerformScript = $fm->newPerformScriptCommand('Order Summary',  
'ComputeTotal');  
$result = $newPerformScript->execute();
```

Esecuzione di uno script prima di eseguire un comando

Utilizzare il metodo `setPreCommandScript()` per indicare uno script che viene eseguito prima che venga eseguito un comando. Nell'esempio seguente viene utilizzato un comando `Find`, ma può essere utilizzato il metodo `setPreCommandScript()` con un comando qualsiasi.

Esempio

```
$findCommand = $fm->newFindCommand('Students');  
$findCommand->addFindCriterion('GPA', $searchValue);  
$findCommand->setPreCommandScript('UpdateGPA');  
$result = $findCommand->execute();
```

Esecuzione di uno script prima di ordinare un gruppo di risultati

Utilizzare il metodo `setPreSortScript()` per specificare uno script che viene eseguito dopo che è stato generato un gruppo di risultati `Find`, ma prima che tale gruppo venga ordinato. Per ulteriori informazioni vedere “Utilizzo del comando `Find`” a pagina 39.

Esempio

```
$findCommand = $fm->newFindCommand('Students');  
$findCommand->setPreSortScript('RemoveExpelled');
```

Esecuzione di uno script dopo la creazione di un gruppo di risultati

Utilizzare il metodo `setScript()` per specificare uno script eseguito dopo la creazione di un gruppo di risultati `Find`. Per ulteriori informazioni vedere “Utilizzo del comando `Find`” a pagina 39.

Esempio

```
$findCommand = $fm->newFindCommand('Students');  
$findCommand->setScript('myScript', 'param1|param2|param3');
```

Ordine di esecuzione degli script

Per un singolo comando si possono indicare i metodi `setPreCommandScript()`, `setPreSortScript()` e `setScript()` insieme ai metodi `setResultLayout()` e `addSortRule()`. Di seguito è indicato l'ordine in cui FileMaker Server e il Motore per la Pubblicazione Web elaborano questi metodi:

1. Eseguire lo script indicato con il metodo `setPreCommandScript()`, se specificato.
2. Elaborare il comando come un comando Find o Delete Record.
3. Eseguire lo script indicato con il metodo `setPreSortScript()`, se specificato.
4. Ordinare il gruppo di risultati Find, se è stato indicato il metodo `addSortRule()`.
5. Elaborare il metodo `setResultLayout()` per passare ad un formato diverso, se specificato.
6. Eseguire lo script indicato con il metodo `setScript()`, se specificato.
7. Restituire il gruppo di risultati Find finale.

Se in uno dei passi indicati in precedenza viene generato un codice di errore, l'esecuzione del comando viene arrestata; eventuali passi successivi non vengono eseguiti. Vengono tuttavia eseguite tutte le fasi precedenti della richiesta.

Ad esempio, considerare un comando che elimina il record corrente, ordina i record e poi esegue uno script. Se il metodo `addSortRule()` specifica un campo inesistente, la richiesta elimina il record corrente e restituisce il codice di errore 102 (“Manca un campo”), ma non esegue lo script.

Utilizzo dei formati FileMaker

Un formato è un insieme di campi, oggetti, immagini e sezioni che rappresenta il modo in cui le informazioni vengono organizzate e presentate durante l'utilizzo, la visualizzazione in anteprima e la stampa dei record. La classe FileMaker definisce vari metodi che consentono di utilizzare i formati definiti in un database FileMaker Pro. Molti oggetti della classe FileMaker consentono di ottenere informazioni sui formati.

Con questo oggetto classe	Utilizzare questi metodi
Database	<ul style="list-style-type: none"> ▪ <code>listLayouts()</code> consente di ottenere una lista di nomi formato disponibili. ▪ <code>getLayout()</code> consente di ottenere un oggetto del formato indicando un nome formato.
Formato	<ul style="list-style-type: none"> ▪ <code>getName()</code> recupera il nome formato di un determinato oggetto del formato. ▪ <code>listFields()</code> recupera una matrice di tutti i nomi campo utilizzati in un formato. ▪ <code>getFields()</code> recupera una matrice associativa con i nomi di tutti i campi come chiave e gli oggetti FileMaker_Field associati come valori di matrice. ▪ <code>listValueLists()</code> recupera una matrice di nomi lista valori. ▪ <code>listRelatedSets()</code> recupera una matrice dei nomi dei set correlati. ▪ <code>getDatabase()</code> restituisce il nome del database.
Numero di record	<ul style="list-style-type: none"> ▪ <code>getLayout()</code> restituisce l'oggetto del formato associato ad un record specifico.
Campo	<ul style="list-style-type: none"> ▪ <code>getLayout()</code> restituisce l'oggetto del formato contenente un campo specifico.
Comando	<ul style="list-style-type: none"> ▪ <code>setResultLayout()</code> restituisce i risultati del comando in un formato diverso da quello corrente.

Utilizzo dei portali

Un portale è una tabella che visualizza righe di dati di uno o più record correlati. La classe FileMaker definisce un oggetto set correlato e diversi metodi che consentono di utilizzare i portali definiti in un database FileMaker Pro.

Un oggetto set correlato è una matrice di oggetti record del portale a cui si fa riferimento; ogni oggetto record rappresenta una riga di dati del portale.

Elenco dei portali definiti in un formato specifico

Per un determinato oggetto del formato, utilizzare il metodo `listRelatedSets()` per recuperare una lista dei nomi tabella per tutti i portali definiti in questo formato.

Esempio

```
$tableNames = $currentLayout->listRelatedSets();
```

Recupero dei nomi portale per un oggetto risultato specifico

Per un determinato oggetto FileMaker_Result, usare il metodo `getRelatedSets()` per recuperare i nomi di tutti i portali di questo record.

Esempio

```
$relatedSetsNames = $result->getRelatedSets();
```

Recupero di informazioni sui portali per un formato specifico

Per un determinato oggetto del formato, utilizzare il metodo `getRelatedSets()` per recuperare una matrice di oggetti FileMaker_RelatedSet che descrivono i portali del formato. La matrice restituita è una matrice associativa con i nomi tabella come chiavi di matrice e gli oggetti associati FileMaker_RelatedSet come valori di matrice.

Esempio

```
$relatedSetsArray = $currentLayout->getRelatedSets();
```

Recupero di informazioni per un portale specifico

Per un determinato oggetto del formato, utilizzare il metodo `getRelatedSet()` per recuperare l'oggetto FileMaker_RelatedSet che descrive uno specifico portale.

Esempio

```
$relatedSet = $currentLayout->getRelatedSet('customers');
```

Recupero dei nomi tabella per un portale

Per un oggetto set correlato, utilizzare il metodo `getName()` per ottenere il nome tabella del portale.

Esempio

```
$tableName = $relatedSet->getName();
```

Recupero dei record del portale per un record specifico

Per un determinato oggetto record, utilizzare il metodo `getRelatedSet()` per recuperare una matrice di record correlati per un portale specifico su tale record.

Esempio

```
$relatedRecordsArray = $currentRecord->getRelatedSet('customers');
```

Creazione di un nuovo record del portale

Utilizzare il metodo `newRelatedRecord()` per creare un nuovo record nel set correlato indicato e salvare le modifiche nel database utilizzando il metodo `commit()`.

Esempio

```
//creare una nuova riga portale nel portale 'customer'
$new_row = $currentRecord->newRelatedRecord('customer');

//impostare i valori campo nella nuova riga del portale
$new_row->setField('customer::name', $newName);
$new_row->setField('customer::company', $newCompany);

$result = $new_row->commit();
```

Eliminazione di un record da un portale

Utilizzare il metodo `delete()` per eliminare un record di un portale.

Esempio

```
$relatedSet = $currentRecord->getRelatedSet('customers');
/* Runs through each of the portal rows */
foreach ($relatedSet as $nextRow) {

    $nameField = $nextRow->getField('customer::name')
    if ($nameField == $badName ) {
        $result = $nextRow->delete();
    }
}
```

Utilizzo delle liste valori

Una lista valori è un set di scelte predefinite. La classe FileMaker definisce vari metodi che consentono di utilizzare le liste valori definite in un database FileMaker Pro.

Recupero dei nomi di tutte le liste valori di un determinato formato

Per un determinato oggetto del formato, utilizzare il metodo `listValueLists()` per recuperare una matrice che contiene i nomi delle liste valori.

Esempio

```
$valueListNames = $currentLayout->listValueLists();
```

Recupero di una matrice di tutte le liste valori di un determinato formato

Per un determinato oggetto del formato, utilizzare il metodo `getValueListsTwoFields()` per recuperare una matrice che contiene i valori di tutte le liste valori. La matrice restituita è una matrice associativa. Le chiavi della matrice sono nomi della lista valori e i valori della matrice sono matrici associative che elencano i nomi da visualizzare e le scelte corrispondenti di ciascuna lista valori.

Esempio

```
$valueListsArray = $currentLayout->getValueListsTwoFields();
```

Nota Sebbene il metodo `getValueLists()` sia ancora supportato in FileMaker API per PHP, è obsoleto. Utilizzare invece il metodo `getValueListsTwoFields()`.

Recupero dei valori per una lista valori indicata

Per un determinato oggetto del formato, utilizzare il metodo `getValueListTwoFields()` per ottenere una matrice di scelte definite per la lista di valori indicata. La matrice restituita è una matrice associativa che contiene i valori visualizzati del secondo campo della lista valori come chiavi, e i valori memorizzati associati del primo campo, come valori della matrice.

A seconda delle opzioni selezionate nella finestra di dialogo Specifica i campi per la lista valori nel database FileMaker, il metodo `getValueListTwoFields()` restituisce, come valori memorizzati e visualizzati, solo il valore del primo campo, solo del secondo campo o i valori di entrambi i campi di una lista valori.

- Se non è selezionato **Visualizza anche i valori secondo campo**, il metodo `getValueListTwoFields()` restituisce il valore del primo campo della lista valori sia come valore memorizzato che come valore visualizzato.
- Se **Visualizza anche i valori secondo campo** e **Mostra valori solo dal secondo campo** sono entrambi selezionati, il metodo `getValueListTwoFields()` restituisce il valore del primo campo come valore memorizzato e il valore del secondo campo come valore visualizzato.
- Se è selezionato **Visualizza anche i valori secondo campo**, ma non **Mostra valori solo dal secondo campo**, il metodo `getValueListTwoFields()` restituisce il valore del primo campo come valore memorizzato ed entrambi i valori sia del primo che del secondo campo come valore visualizzato.

Per trovare il valore visualizzato e il valore memorizzato, utilizzare un iteratore con il metodo `getValueListTwoFields()`.

Esempio

```
$layout = $fm->getLayout('customers');
$valuearray = $layout->getValueListTwoFields("region", 4);
foreach ($valuearray as $displayValue => $value) {
    ....
}
```

Note

- Sebbene il metodo `getValueList()` sia ancora supportato in FileMaker API per PHP, è obsoleto. Utilizzare invece il metodo `getValueListTwoFields()`.
- Quando si utilizza il metodo `getValueListTwoFields()`, assicurarsi di utilizzare un ciclo `foreach` per scorrere attraverso la matrice associativa. Non utilizzare un ciclo `for` dato che può restituire risultati non previsti.

Esecuzione di richieste di ricerca

La classe FileMaker definisce quattro tipi di oggetti di comando Find:

- Comando Find All. Vedere “Utilizzo del comando Find All” a pagina 39.
- Comando Find Any. Vedere “Utilizzo del comando Find Any” a pagina 39.
- Comando Find. Vedere “Utilizzo del comando Find” a pagina 39.
- Comando Compound Find. Vedere “Utilizzo di un comando Compound Find” a pagina 40.

La classe FileMaker definisce anche vari metodi che possono essere utilizzati per tutti e quattro i tipi di comando Find:

- Utilizzare il metodo `addSortRule()` per aggiungere una regola che definisce il modo in cui ordinare il gruppo di risultati. Utilizzare il metodo `clearSortRules()` per cancellare tutti i criteri di ordinamento definiti.
- Utilizzare il metodo `setLogicalOperator()` per scegliere ricerche con operatore logico AND oppure OR.
- Utilizzare il metodo `setRange()` per richiedere solo una parte del gruppo di risultati. Utilizzare il metodo `getRange()` per recuperare la definizione dell'intervallo corrente.

Usando il metodo `setRange()` si possono migliorare le prestazioni della soluzione riducendo il numero di record restituiti dalla richiesta di ricerca Find. Ad esempio, se una richiesta Find restituisce 100 record, è possibile suddividere il gruppo di risultati in cinque gruppi da 20 record piuttosto che elaborare tutti i 100 record insieme.

- È possibile eseguire script FileMaker insieme a comandi Find.
 - Per eseguire uno script prima di eseguire il comando Find, utilizzare il metodo `setPreCommandScript()`.
 - Per eseguire uno script prima di ordinare il gruppo di risultati, utilizzare il metodo `setPreSortScript()`.
 - Per eseguire uno script dopo che un gruppo di risultati è stato generato, ma non ancora ordinato, utilizzare il metodo `setScript()`.

Utilizzo del comando Find All

Utilizzare il comando Find All per recuperare tutti i record di un determinato formato. Utilizzare il metodo `newFindAllCommand()`, indicando un formato specifico, per creare un oggetto `FileMaker_Command_FindAll`. Quindi, eseguire la richiesta di ricerca utilizzando il metodo `execute()`.

Esempio

```
$findCommand = $fm->newFindAllCommand('Form View');  
$result = $findCommand->execute;
```

Nota Quando si utilizza il comando Find All, evitare problemi di sovraccarico della memoria del computer specificando un numero massimo predefinito di record da restituire per pagina.

Utilizzo del comando Find Any

Utilizzare il comando Find Any per recuperare un record a caso da un determinato formato. Utilizzare il metodo `newFindAnyCommand()`, indicando un formato specifico, per creare un oggetto `FileMaker_Command_FindAny`. Quindi, eseguire la richiesta di ricerca utilizzando il metodo `execute()`.

Esempio

```
$findCommand = $fm->newFindAnyCommand('Form View');  
$result = $findCommand->execute;
```

Utilizzo del comando Find

Utilizzare il metodo `newFindCommand()`, indicando un formato specifico, per creare un oggetto `FileMaker_Command_Find`. Quindi, eseguire la richiesta di ricerca utilizzando il metodo `execute()`.

Utilizzare il metodo `addFindCriterion()` per aggiungere dei criteri alla richiesta di ricerca. Utilizzare il metodo `clearFindCriteria()` per cancellare tutti i criteri di ricerca definiti.

Esempio - Ricerca di un criterio in base al nome del campo

```
$findCommand = $fm->newFindCommand('Form View');  
$findCommand->addFindCriterion('Questionnaire ID', $active_questionnaire_id);  
$result = $findCommand->execute();
```

Esempio - Aggiunta di un criterio di ordinamento

```
$findCommand = $fm->newFindCommand('Customer List');  
$findCommand->addSortRule('Title', 1, FILEMAKER_SORT_ASCEND);  
$result = $findCommand->execute();
```

Utilizzo di un comando Compound Find

Il comando Compound Find consente di combinare più oggetti Find Request in un singolo comando.

Per creare un comando Compound Find:

- Creare un oggetto `FileMaker_Command_CompoundFind` utilizzando il metodo `newCompoundFindCommand()`.
- Creare uno o più oggetti `FileMaker_Command_FindRequest` utilizzando il metodo `newFindRequest()`.
- Utilizzare il metodo `setOmit()` per indicare i record del gruppo di risultati di una richiesta di ricerca specifica che devono essere omessi dal gruppo di risultati finale.
- Utilizzare il metodo `add()` per aggiungere oggetti Find Request a un oggetto di comando Compound Find.
- Eseguire il comando Compound Find utilizzando il metodo `execute()`.

Esempio - Comando Compound Find

```
// Creare l'oggetto di comando Compound Find
$compoundFind = $fm->newCompoundFindCommand('Form View');

// Creare la prima richiesta di ricerca
$findreq1 = $fm->newFindRequest('Form View');

// Creare la seconda richiesta di ricerca
$findreq2 = $fm->newFindRequest('Form View');

// Creare la terza richiesta di ricerca
$findreq3 = $fm->newFindRequest('Form View');

// Indicare un criterio di ricerca per la prima richiesta di ricerca
$findreq1->addFindCriterion('Quantity in Stock', '<100');

// Indicare un criterio di ricerca per la seconda richiesta di ricerca
$findreq2->addFindCriterion('Quantity in Stock', '0');
$findreq2->setOmit(true);

// Indicare un criterio di ricerca per la terza richiesta di ricerca
$findreq3->addFindCriterion('Cover Photo Credit', 'The London Morning News');
$findreq3->setOmit(true);

// Aggiungere le richieste di ricerca ad un comando compound find
$compoundFind->add(1,$findreq1);
$compoundFind->add(2,$findreq2);
$compoundFind->add(3,$findreq3);

// Impostare un ordinamento
$compoundFind->addSortRule('Title', 1, FILEMAKER_SORT_DESCEND);

// Eseguire il comando find command
$result = $compoundFind->execute();

// Ottenere i record dal gruppo trovato
$records = $result->getRecords();

// Stampare il numero dei record trovati
echo 'Found '. Count($records). " results.<br><br>";
```

Elaborazione dei record in un gruppo di risultati

- Recuperare una matrice contenente tutti i record del gruppo di risultati utilizzando il metodo `getRecords()`. Ogni elemento della matrice è un oggetto `FileMaker_Record` o un'istanza del nome classe impostato nell'API per creare un'istanza di record. La matrice può essere vuota se il gruppo di risultati non contiene nessun record.
- Ottenere una lista dei nomi campo per tutti i campi del gruppo di risultati utilizzando il metodo `getFields()`. Il metodo restituisce solo i nomi campo. Per ottenere ulteriori informazioni sui campi, utilizzare l'oggetto del formato associato.
- Ottenere il numero di record di tutto il gruppo trovato utilizzando il metodo `getFoundSetCount()`.
- Ottenere il numero di record del gruppo trovato filtrato utilizzando il metodo `getFetchCount()`. Se nel comando Find non sono stati indicati dei parametri di intervallo, questo valore è uguale al risultato ottenuto con il metodo `getFoundSetCount()`. È sempre uguale al valore di `count($response->getRecords())`.
- Per un determinato record, utilizzare il metodo `getField()` per restituire i contenuti di un campo sotto forma di stringa.
- Per un record specifico utilizzare il metodo `getFieldAsTimestamp()` per restituire i contenuti di un campo sotto forma di indicatore data e ora Unix (rappresentazione interna PHP di una data).
 - Se il campo è un campo data, l'indicatore data e ora per la data del campo è impostato su mezzanotte.
 - Se il campo è un campo ora, l'indicatore data e ora per l'ora è impostato su 1 gennaio 1970.
 - Se il campo è un campo indicatore data e ora, il valore dell'indicatore data e ora FileMaker applica direttamente i valori dell'indicatore data e ora Unix.
 - Se il campo indicato non è un campo data o ora o se l'indicatore data e ora generato non è compreso nell'intervallo, il metodo `getFieldAsTimestamp()` restituisce un oggetto `FileMaker_Error`.
- Per un record specifico, utilizzare il metodo `getContainerData()` per restituire un oggetto del campo Contenitore come dati binari:

```
<IMG src="img.php?url=<?php echo urlencode($record->getField('Cover
Image')); ?>">
echo $fm->getContainerData($_GET['-url']);
```

- Per un record specifico, utilizzare il metodo `getContainerDataURL()` per restituire un URL completamente qualificato per l'oggetto del campo Contenitore:

```
// Per le immagini, utilizzare il tag HTML img
echo '';
// Per i filmati e i file PDF, utilizzare il tag HTML embed
//echo '<embed src="'. $fm->
getContainerDataURL($record->getField('container')) ."'>';
```

Filtraggio delle righe del portale restituite dalle richieste di ricerca

In una soluzione con molti record correlati, l'interrogazione e l'ordinamento dei record del portale potrebbe richiedere molto tempo. Per limitare il numero di record da visualizzare nel set correlato, utilizzare il metodo `setRelatedSetsFilters()` con le richieste di ricerca. Il metodo `setRelatedSetsFilters()` comprende due argomenti:

- valore del filtro dei set correlati: `layout` oppure `none`.
 - Indicando il valore `none` il Motore per la Pubblicazione Web restituisce tutte le righe del portale e i record del portale non vengono preordinati.
 - Indicando il valore `layout`, vengono applicate le impostazioni effettuate nella finestra di dialogo Impostazione portale di FileMaker Pro. I record sono ordinati in base all'ordinamento definito nella finestra di dialogo Impostazione portale, con il set di record filtrato in modo da partire con la riga iniziale specificata.
- il numero massimo di record del portale ha restituito un valore intero o `tutti`.
 - Questo valore è utilizzato solo se nella finestra di dialogo Impostazione portale è attivata l'impostazione Mostra la barra di scorrimento verticale. Se viene indicato un valore intero, viene restituito un numero di righe pari a quel numero a partire dalla riga iniziale. Indicando `tutti`, il Motore per la Pubblicazione Web restituisce tutti i record correlati.
 - Se l'impostazione Mostra la barra di scorrimento verticale è disabilitata, l'impostazione Numero di righe nella finestra di dialogo Impostazione portale stabilisce il numero massimo di record correlati restituiti.

Pre-convalida di comandi, record e campi

La classe FileMaker consente di *pre-convalidare* i dati di un campo di una soluzione PHP sul server Web prima di salvare i dati nel database.

Se si decide di usare una pre-convalida, considerare il numero dei valori dei dati che l'utente Web sta inserendo. Se l'utente sta aggiornando un numero ridotto di campi, le prestazioni possono essere migliorate evitando di utilizzare una pre-convalida. Ma se l'utente sta inserendo dati in molti campi, la pre-convalida fa sì che non accada che un record venga respinto dal database a causa di errori di convalida.

Con la classe FileMaker, il motore PHP esegue la pre-convalida delle seguenti limitazioni di campo:

- non vuoto
I dati validi sono una stringa di caratteri non vuota. I dati devono contenere almeno un carattere.
- solo numerico
I dati validi contengono solo caratteri numerici.
- numero massimo di caratteri
I dati validi contengono per lo più il numero massimo di caratteri specificato.

- anno a quattro cifre

I dati validi sono una stringa di caratteri che indica una data con un anno a quattro cifre nel formato g/m/aaaa, dove m è un numero tra 1 e 12 compreso, g è un numero tra 1 e 31 compreso e aaaa è un numero a quattro cifre tra 0001 e 4000 compreso. Per esempio 30/1/3030 è un valore anno a quattro cifre valido. Tuttavia, 31/4/2013 è un valore a quattro cifre non valido dato che aprile non ha 31 giorni. La convalida della data supporta come delimitatori i caratteri barra (/), barra rovesciata (\) e trattino (-). Tuttavia la stringa non può contenere una combinazione di delimitatori. Ad esempio, la stringa 30\1-2012 non è valida.

- ora del giorno

I dati validi sono una stringa di caratteri che indica un valore ora da 12 ore in uno di questi formati:

- H
- H:M
- H:M:S
- H:M:S AM/PM
- H:M AM/PM

dove H è un numero tra 1 e 12 compreso; M e S sono numeri tra 1 e 60 compreso.

La pre-convalida del motore PHP supporta il controllo automatico dei dati dei campi in base al tipo di campo:

- data

Un campo definito come campo Data è convalidato in base al criterio "anno a quattro cifre", fatta eccezione per il valore anno che può contenere da 0-4 cifre (il valore dell'anno può essere vuoto). Ad esempio 1/30 è una data valida anche se non viene specificato nessun anno.

- ora

Un campo definito come campo Ora è convalidato in base al criterio "ora del giorno", fatta eccezione per il componente ora (H) che può essere un numero tra 1 e 24 compreso, per supportare i valori dei formati da 24 ore.

- indicatore data e ora

Un campo definito come Indicatore data e ora è convalidato in base al criterio "ora" per il componente ora e in base al criterio "data" per il componente data.

La classe FileMaker non può eseguire una pre-convalida di tutte le opzioni di convalida campo disponibili in FileMaker Pro. Le seguenti opzioni di convalida non possono essere pre-convalidate preche dipendono dallo stato di tutti i dati del database nel momento in cui i dati sono stati salvati:

- valore unico
- valore esistente
- da-a
- elemento della lista valori
- verifica mediante calcolo

Pre-convalida dei record in un comando

Per un oggetto di comando, utilizzare il metodo `validate()` per convalidare un campo o tutto il comando in base ai criteri di convalida applicati dal motore PHP. Se viene utilizzato l'argomento del nome campo opzionale, viene convalidato solo quel campo.

Se la pre-convalida riesce, il metodo `validate()` restituisce `TRUE`. Se la pre-convalida fallisce, il metodo `validate()` restituisce l'oggetto `FileMaker_Error_Validation` che contiene le informazioni su ciò che non ha reso possibile la convalida.

Pre-convalida dei record

Per un oggetto record, utilizzare il metodo `validate()` per convalidare un campo o tutti i campi del record in base ai criteri di convalida applicati dal motore PHP. Se viene utilizzato l'argomento del nome campo opzionale, viene convalidato solo quel campo.

Se la pre-convalida riesce, il metodo `validate()` restituisce `TRUE`. Se la pre-convalida fallisce, il metodo `validate()` restituisce l'oggetto `FileMaker_Error_Validation` che contiene le informazioni su ciò che non ha reso possibile la convalida.

Pre-convalida dei campi

Per un oggetto del campo, utilizzare il metodo `validate()` per stabilire se un determinato valore è valido per un campo oppure no.

Se la pre-convalida riesce, il metodo `validate()` restituisce `TRUE`. Se la pre-convalida fallisce, il metodo `validate()` restituisce l'oggetto `FileMaker_Error_Validation` che contiene le informazioni su ciò che non ha reso possibile la convalida.

Elaborazione degli errori di convalida

Se la pre-convalida fallisce, l'oggetto `FileMaker_Error_Validation` restituito contiene una matrice a tre elementi per ogni errore di convalida:

1. L'oggetto del campo che non ha superato la pre-convalida
2. Un valore di convalida costante che indica il criterio di convalida che ha provocato l'errore:
 - 1 - `FILEMAKER_RULE_NOTEMPTY`
 - 2 - `FILEMAKER_RULE_NUMERICONLY`
 - 3 - `FILEMAKER_RULE_MAXCHARACTERS`
 - 4 - `FILEMAKER_RULE_FOURDIGITYEAR`
 - 5 - `FILEMAKER_RULE_TIMEOFDAY`
 - 6 - `FILEMAKER_RULE_TIMESTAMP_FIELD`
 - 7 - `FILEMAKER_RULE_DATE_FIELD`
 - 8 - `FILEMAKER_RULE_TIME_FIELD`
3. Il valore effettivo inserito nel campo che non ha superato la pre-convalida

È inoltre possibile utilizzare i seguenti metodi con un oggetto `FileMaker_Error_Validation`:

- Utilizzare il metodo `isValidationErrorMessage()` per verificare se l'errore è un errore di convalida.
- Utilizzare il metodo `numErrors()` per ottenere il numero di criteri di convalida che hanno dato errore.

Esempio

```
//Creare una richiesta Add
$addrequest = $fm->newAddCommand('test', array('join' => 'added', 'maxchars' =>
'abcx', 'field' => 'something' , 'numericonly' => 'abc'));

//Convalidare tutti i campi
$result = $addrequest->validate();

//Se il metodo validate() restituisce degli errori, stampare il nome del campo,
il numero dell'errore e il valore che ha dato errore.
if(FileMaker::isError($result)){
    echo 'Validation failed:'. "\n";
    $validationErrors= $result->getErrors();
    foreach ($validationErrors as $error) {
        $field = $error[0];
        echo 'Field Name: ' . $field->getName(). "\n";
        echo 'Error Code: ' . $error[1]. "\n";
        echo 'Value: ' . $error[2]. "\n";
    }
}
```

Output

```
Errore di convalida:
Nome campo: numericonly
Codice di errore: 2
Valore: abcx
Nome campo: maxchars
Codice di errore: 3
Valore: abc
```

Gestione degli errori

La classe FileMaker definisce l'oggetto FileMaker_Error per supportare la gestione degli errori che si verificano in una soluzione PHP.

L'errore può verificarsi durante l'esecuzione di un comando. Se si verifica un errore, il comando restituisce l'oggetto FileMaker_Error. Si consiglia di controllare l'errore che viene restituito quando viene eseguito un comando.

Utilizzare i seguenti metodi per avere ulteriori informazioni sull'errore indicato nell'oggetto FileMaker_Error.

- Controllare se una variabile è un oggetto FileMaker Error utilizzando il metodo `isError()` .
- Ricavare il numero di errori che si sono verificati utilizzando il metodo `numErrors()` .
- Recuperare una matrice di matrici che descrivono gli errori che si sono verificati utilizzando il metodo `getErrors()` .
- Visualizzare un messaggio di errore utilizzando il metodo `getMessage()` .

Esempio

```
$result = $findCommand->execute();
if (FileMaker::isError($result)) {
    echo "<p>Error: " . $result->getMessage() . "</p>";
    exit;
}
```

Per informazioni sui codici di errore restituiti con l'oggetto FileMaker Error, vedere appendice A, “Codici di errore per la Pubblicazione Web personalizzata con PHP.”

Capitolo 6

Preparazione, test e monitoraggio di un sito

In questo capitolo sono contenute le istruzioni per preparare e testare un sito Pubblicazione Web personalizzata prima della sua distribuzione in un ambiente di produzione. Sono anche presenti le istruzioni per utilizzare i file di registro per controllare il sito durante il test o dopo la sua distribuzione.

Preparazione di un sito Pubblicazione Web personalizzata

Prima di poter testare correttamente il sito, è necessario copiare o spostare i file richiesti nelle posizioni corrette sul/sui server di preparazione.

Per organizzare il sito e prepararlo per il test:

1. Completare tutti i passi in capitolo 3, "Preparazione dei database per la Pubblicazione Web personalizzata."
2. Controllare che la Pubblicazione Web personalizzata con PHP sia stata attivata e configurata correttamente nell'Admin Console di FileMaker Server.

Nota Per istruzioni, vedere la Guida di FileMaker Server.

3. Verificare che il server Web e il Motore per la Pubblicazione Web siano in esecuzione.
4. Copiare o spostare i file del sito nel componente server Web della distribuzione FileMaker Server.

Copiare o spostare i file del sito nella seguente directory sulla macchina del server Web:

- Per IIS (Windows) attraverso HTTP o HTTPS:
[unità]:\Programmi\FileMaker\FileMaker Server\HTTPServer\Conf dove [unità] è l'unità in cui risiede il componente del motore della Pubblicazione Web della distribuzione FileMaker server.
- Per Apache (OS X) attraverso HTTP: /Library/FileMaker Server/HTTPServer/htdocs
- Per Apache (OS X) attraverso HTTP:
/Library/FileMaker Server/HTTPServer/htdocs/httpsRoot

5. Se non è ancora stato fatto, copiare o spostare eventuali oggetti Contenitore a cui si fa riferimento nella directory corretta sulla macchina del server Web.
 - Se il file di database è ospitato nella posizione appropriata e accessibile dal componente server database di FileMaker Server e se nei campi Contenitore vengono memorizzati i file effettivi del database FileMaker, i contenuti del campo Contenitore non richiedono un riposizionamento.
 - Se un campo Contenitore del database contiene un riferimento al file anziché il file reale, l'oggetto Contenitore di riferimento deve essere memorizzato nella cartella Web di FileMaker Pro quando il record viene creato o modificato. Per preparare il sito, copiare o spostare i contenitori a cui si fa riferimento ad una cartella con la stessa posizione relativa nella cartella principale del software del server Web.
 - Se si utilizza FileMaker Pro per caricare un database con campi Contenitore che memorizzano oggetti esternamente, i dati del campo Contenitore memorizzati esternamente vengono memorizzati in FileMaker Server come parte del processo. Vedere la Guida di FileMaker Pro per informazioni sul trasferimento dei file di database in FileMaker Server.
 - Se si carica manualmente un database che utilizza un campo Contenitore con oggetti memorizzati esternamente, è necessario copiare o spostare gli oggetti a cui fa riferimento in una sottocartella della cartella RC_Data_FMS, come descritto in “Campi Contenitore con dati memorizzati esternamente” a pagina 21.

Vedere “Pubblicazione dei contenuti dei campi Contenitore sul Web” a pagina 19.

6. Iniziare a testare il sito.

Test di un sito Pubblicazione Web personalizzata

Prima di comunicare agli utenti che il sito Pubblicazione Web Immediata è disponibile, è importante verificare che abbia il layout desiderato e che funzioni nel modo previsto.

- Testare le funzioni come ad esempio la ricerca, l'aggiunta, l'eliminazione e l'ordinamento dei record con account e set di privilegi diversi.
- Verificare che i set di privilegi funzionino come previsto in base al tipo di account utilizzato. Assicurarsi che gli utenti non autorizzati non possano accedere o modificare i dati.
- Controllare tutti gli script per verificare che producano il risultato previsto. Per informazioni su come progettare script compatibili con il Web consultare la sezione “Script di FileMaker e Pubblicazione Web personalizzata” a pagina 23.

- Testare il sito con sistemi operativi e browser Web diversi.
- Quando si creano soluzioni utilizzando FileMaker API per PHP, si raccomanda di costruirle con il supporto dei cookie attivato. Con i cookie attivati, infatti, FileMaker API per PHP ha tempi di risposta migliori. I cookie non sono necessari per utilizzare le funzioni della Pubblicazione Web personalizzata, ma permettono al Motore per la Pubblicazione Web di memorizzare nella cache le informazioni della sessione.

Nota Se il server Web, il Motore per la Pubblicazione Web e il server Database sono stati installati in una distribuzione con una singola macchina, è possibile visualizzare e testare il sito senza utilizzare una connessione di rete. Spostare i file del sito nella directory appropriata su tale macchina e inserire nel browser il seguente URL:

```
http://127.0.0.1/<percorso_sito>
```

dove <percorso_sito> è il percorso relativo alla home page del sito.

Monitoraggio di un sito

Utilizzare i seguenti tipi di file di registro per controllare un sito Pubblicazione Web personalizzata e raccogliere informazioni sugli utenti web che visitano il sito:

- Registri degli accessi e degli errori del server Web
- Registro Motore per la Pubblicazione Web
- Registro errori Modulo Server Web
- Registri Tomcat

Uso dei registri degli accessi e degli errori del server Web

- IIS (Windows): Il server Web Microsoft IIS genera file di registro degli accessi e mostra gli errori nel Visualizzatore eventi di Windows anziché registrarli in un file di registro. Il file di registro degli accessi, incluso per default nell'Extended Log File Format W3C, è un registro di tutte le richieste HTTP in ingresso sul server Web. È anche possibile usare il Common Logfile Format per il registro degli accessi. Per ulteriori informazioni consultare la documentazione del server Web Microsoft IIS.
- Apache (solo OS X): Il server Web Apache genera un file di registro degli accessi e un file di registro degli errori. Il file di registro degli accessi Apache, incluso per default nel Common Logfile Format W3C, è un registro di tutte le richieste HTTP in ingresso sul server Web. Il registro errori Apache è un registro dei problemi che riguardano richieste HTTP di elaborazione. Per ulteriori informazioni su questi file di registro, vedere la documentazione del server Web Apache.

Nota Per informazioni sul W3C Common Logfile Format e sul W3C Extended Log File Format, consultare il sito Web del World Wide Web Consortium all'indirizzo <http://www.w3.org>.

Utilizzo del registro del Motore per la Pubblicazione Web

Per impostazione predefinita, il Motore per la Pubblicazione Web genera un file di registro chiamato `wpe.log` che è un registro degli errori del Motore per la Pubblicazione Web, compresi errori dell'applicazione, errori di utilizzo e errori di sistema. È anche possibile far sì che il Motore per la Pubblicazione Web comprenda le informazioni relative alla Pubblicazione Web personalizzata, come le richieste dell'utente finale XML per generare dati di uscita di pubblicazione Web o modifiche alle impostazioni di Pubblicazione Web personalizzata.

Dato che FileMaker API per PHP utilizza HTTP POST per accedere al Motore per la Pubblicazione Web, il file `wpe.log` non registra informazioni relative alle richieste PHP. Il file `wpe.log` può essere utilizzato per verificare quando gli utenti hanno fatto le richieste PHP, consultando le richieste XML registrate.

Il file `wpe.log` si trova nel componente Motore per la Pubblicazione Web della distribuzione FileMaker Server:

- **IIS (Windows):** `[unità]:\Program Files\FileMaker\FileMaker Server\HTTPServer\logs\wpe.log`
dove `[unità]` è l'unità principale da cui viene avviato il sistema.
- **Apache (OS X):** `/Library/FileMaker Server/HTTPServer/Logs/wpe.log`

Impostazioni del registro Motore per la Pubblicazione Web

Il file `wpe.log` viene generato se l'opzione **Consenti registrazione per Pubblicazione Web** è abilitata nell'Admin Console.

Opzione registro abilitata	Informazioni registrate nel <code>wpe.log</code>
Messaggi livello errore	Gli errori Motore per la Pubblicazione Web, compresi gli errori di applicazione, errori di utilizzo ed errori di sistema.
Messaggi livello errore e informazioni	Gli errori sopra descritti e le informazioni sull'accesso al Motore per la Pubblicazione Web. Contiene un record con tutte le richieste degli utenti finali XML per generare dati di uscita della Pubblicazione Web personalizzata.

L'impostazione **Messaggi livello errore** è attivata in modo predefinito. Per informazioni sull'impostazione di queste opzioni offerte da Admin Console, vedere la Guida di FileMaker Server.

Importante Col tempo il file `wpe.log` può diventare molto grande. Utilizzare l'Admin Console per impostare la dimensione massima per il file `wpe.log`. Quando il file `wpe.log` raggiunge queste dimensioni massime, il Motore per la Pubblicazione Web copia il file `wpe.log` in un singolo file backup, `wpe.log.1`, e crea un nuovo file `wpe.log`. Se si desidera più di una copia di backup, è possibile salvare regolarmente un archivio del file `wpe.log.1`.

Formato registro Motore per la Pubblicazione Web

Il file `wpe.log` utilizza la seguente formattazione per ogni voce:

```
[TIMESTAMP_GMT] [WPC_HOSTNAME] [CLIENT_IP:PORT] [ACCOUNT_NAME] [MODULE_TYPE]
[SEVERITY] [FM_ERRORCODE] [RETURN_BYTES] [MESSAGE]
```

dove:

- `[TIMESTAMP_GMT]` è la data e l'ora dell'inserimento, ora di Greenwich (GMT).
- `[WPC_HOSTNAME]` è il nome della macchina su cui è installato il Motore per la Pubblicazione Web.
- `[CLIENT_IP:PORT]` è l'indirizzo IP e la porta del client dove è stata originata la richiesta XML.
- `[ACCOUNT_NAME]` è il nome dell'account utilizzato per l'accesso al database FileMaker ospitato.
- `[MODULE_TYPE]` è: XML, per la Pubblicazione Web personalizzata con richieste XML, oppure PHP, per la Pubblicazione Web personalizzata con le richieste PHP.
- `[SEVERITY]` è INFO, che indica un messaggio informativo, oppure ERROR, che indica un messaggio di errore.
- `[FM_ERROR_CODE]` è il numero dell'errore riportato per un messaggio di errore. Il numero dell'errore può essere un codice di errore per i database FileMaker (vedere "Codici di errore per database FileMaker" a pagina 55).

Inoltre, il numero dell'errore può essere un numero di errore HTTP, preceduto da una stringa "HTTP:".

- `[RETURN_BYTES]` è il numero dei byte restituiti dalla richiesta.
- `[MESSAGE]` fornisce informazioni aggiuntive sulla voce di registro.

Esempi di messaggi di registro del Motore per la Pubblicazione Web

I seguenti esempi mostrano i tipi di messaggi che possono essere inclusi nel file `wpe.log`:

- Quando il Motore per la Pubblicazione Web viene avviato e arrestato

```
02/06/2013 15:15:31 -0700 - - - - INFO - - Motore per la
Pubblicazione Web FileMaker Server avviato.
02/06/2013 15:46:52 -0700 - - - - INFO - - Motore per la
Pubblicazione Web FileMaker Server arrestato.
```

- Richieste query XML riuscite o fallite

```
02/06/2013 15:21:08 -0700 WPC_SERVER 192.168.100.101:0 jdoe XML
INFO 0 3964 "/fmi/xml/fmresultset.xml?-db=Contacts&-
lay=Contact_Details&-findall"
02/06/2013 15:26:31 -0700 WPC_SERVER 192.168.100.101:0 jdoe XML
ERRORE 5 596 "/fmi/xml/fmresultset.xml?-db=Contacts&-
layout=Contact_Details&-findall"
```

- Errori di script

```
02/06/2013 17:33:12 -0700 WPC_SERVER 192.168.100.101:0 jdoe - ERRORE
4 - Errore di Script Web: 4, File: "10b_MeetingsUpload", script:
"OnOpen", istruzione di script: "Mostra finestra personalizz."
```

- Modifiche alle impostazioni di Pubblicazione Web Personalizzato

```
09/06/2013 10:59:49 -0700 WPC_SERVER 192.168.100.101:0 jdoe - INFO
- - Il Motore per la Pubblicazione Web XML è abilitato.
```

- Errori di sistema

```
02/06/2013 15:30:42 -0700 WPC_SERVER 192.168.100.101:0 jdoe XML
ERROR - - Errore nella comunicazione
```

Utilizzo del registro errori del Modulo Server Web

Se il server Web non è in grado di connettersi al Motore per la Pubblicazione Web, il Modulo Server Web genera un file di registro che è un registro di tutti gli errori che si sono verificati in questa operazione. Questo file di registro è chiamato `web_server_module_log.txt` e si trova nel componente server Web della distribuzione FileMaker Server:

- IIS (Windows): `[unità]:\Programmi\FileMaker\FileMaker Server\Logs\web_server_module_log.txt`
dove `[unità]` è l'unità principale da cui viene avviato il sistema.
- Apache (OS X): `/Libreria/FileMaker Server/Logs/web_server_module_log.txt`

Utilizzo dei registri Tomcat

Quando FileMaker Server ha un problema provocato da un errore nel server Web interno, potrebbe essere utile visualizzare i registri Tomcat. I registri Tomcat si trovano sul componente del server Web della distribuzione FileMaker Server:

- IIS (Windows): `[unità]:\Program Files\FileMaker\FileMaker Server\Admin\admin-master-tomcat\logs\`
dove `[unità]` è l'unità principale da cui viene avviato il sistema.
- Apache (OS X): `/Library/FileMaker Server/Admin/admin-master-tomcat/logs/`

Risoluzione dei problemi del sito

Se si presentano dei problemi nella visualizzazione o nell'utilizzo del sito, verificare che:

- I privilegi estesi nel database siano impostati per la Pubblicazione Web personalizzata con PHP e assegnati a un account utente. Vedere “Attivazione della pubblicazione Web personalizzata con PHP per database” a pagina 15.
 - Il database sia ospitato e aperto da FileMaker Server. Vedere la Guida di FileMaker Server.
 - Gli eventuali nomi utente e password utilizzati per il database siano corretti.
 - Il server Web e il Motore per la Pubblicazione Web siano in esecuzione.
 - La Pubblicazione PHP sia attivata nel Motore per la Pubblicazione Web.
 - Aprire la pagina relativa alle prove tecniche di FileMaker Server:
`http://<server>:16000/test`
- In cui <server> è la macchina in cui risiede FileMaker Server.
- Fare clic sul collegamento **Test Pubblicazione Web personalizzata PHP** per aprire una pagina PHP con accesso al database di test FMServer_Sample.

Per ulteriori informazioni, vedere *Guida introduttiva di FileMaker Server* e la Guida di FileMaker Server.

Appendice A

Codici di errore per la Pubblicazione Web personalizzata con PHP

Il Motore per la Pubblicazione Web supporta due tipi di codice di errore che possono verificarsi nella Pubblicazione Web personalizzata:

- Errori database e di richiesta dati. Il Motore per la Pubblicazione Web genera un codice di errore ogniqualvolta vengono richiesti dei dati da un database pubblicato. FileMaker API per PHP restituisce questo codice di errore come oggetto FileMaker_Error. Vedere la sezione successiva, “Codici di errore per database FileMaker.” .
- errori PHP. Questi errori sono generati e restituiti dai componenti PHP, compreso il modulo cURL. Vedere “Codici di errore per i componenti PHP” a pagina 62.

Codici di errore per database FileMaker

Spetta allo sviluppatore della soluzione Pubblicazione Web personalizzata controllare il valore del codice di errore e gestirlo in modo opportuno. Il Motore per la Pubblicazione Web non gestisce gli errori di database.

Codice errore	Descrizione
-1	Errore sconosciuto
0	Nessun errore
1	Azione annullata dall'utente
2	Errore di memoria
3	Comando non disponibile (ad esempio, sistema operativo non appropriato, modo errato e così via)
4	Comando sconosciuto
5	Comando non valido (ad esempio, un'istruzione di script Imposta campo priva di calcolo)
6	File di sola lettura
7	Memoria esaurita
8	Risultato vuoto
9	Privilegi insufficienti
10	Dati richiesti non disponibili
11	Nome non valido
12	Il nome esiste già
13	File o oggetto in uso
14	Fuori intervallo
15	Impossibile dividere per zero
16	Operazione non riuscita, nuovo tentativo richiesta (ad esempio una query dell'utente)
17	Tentativo di convertire il set di caratteri stranieri in UTF-16 non riuscito
18	Il client deve fornire le informazioni sull'account per procedere

Codice errore	Descrizione
19	La stringa contiene caratteri diversi da A-Z, a-z, 0-9 (ASCII)
20	Comando o operazione annullata dallo script attivato
21	Richiesta non supportata (ad esempio, quando si crea un collegamento fisico su un file system che non supporta collegamenti fisici)
100	Manca un file
101	Manca un record
102	Manca un campo
103	Manca una relazione
104	Manca uno script
105	Manca un formato
106	Manca una tabella
107	Manca un indice
108	Manca una lista valori
109	Manca un set di privilegi
110	Mancano tabelle correlate
111	Ripetizione campo non valida
112	Manca una finestra
113	Manca una funzione
114	Manca un riferimento al file
115	Set di menu mancante
116	Oggetto del formato mancante
117	Origine dati mancante
118	Manca un tema
130	File danneggiati o non presenti; reinstallarli
131	Impossibile trovare i file del supporto per la lingua (come le soluzioni pronte per l'uso)
200	Accesso al record negato
201	Impossibile modificare il campo
202	Accesso al campo negato
203	Nel file non c'è nessun record da stampare o la password non consente l'accesso alla stampa
204	Nessun accesso ai campi nei criteri di ordinamento
205	L'utente non dispone dei privilegi di accesso per creare nuovi record; l'importazione sovrascriverà i dati esistenti
206	L'utente non dispone del privilegio per cambiare la password o il file non è modificabile
207	L'utente non ha privilegi sufficienti per cambiare lo schema del database, oppure il file non è modificabile
208	La password non contiene abbastanza caratteri
209	La nuova password deve essere diversa da quella esistente
210	L'account utente è inattivo
211	La password è scaduta

Codice errore	Descrizione
212	Account utente e/o password non validi. Riprovare
213	Il nome utente e/o la password non esistono
214	Troppi tentativi di accesso
215	I privilegi di amministratore non possono essere duplicati
216	L'account Ospite non può essere duplicato
217	L'utente non dispone di privilegi sufficienti per modificare l'account Admin
218	Password e password di conferma non corrispondono
300	File bloccato o in uso
301	Record usato da un altro utente
302	Tabella usata da un altro utente
303	Schema database usato da un altro utente
304	Formato usato da un altro utente
306	ID modifica del record non corrispondente
307	Non è stato possibile bloccare la transazione a causa di un errore di comunicazione con l'host
308	Tema bloccato e usato da un altro utente
400	Criteri di ricerca vuoti
401	Nessun record soddisfa la richiesta
402	Il campo selezionato non è un campo di confronto per un riferimento
403	Limite massimo di record per la versione di prova di FileMaker Pro superato
404	Criterio di ordinamento non valido
405	Il numero di record specificato supera il numero di record che possono essere omessi
406	Criteri di sostituzione/riserializzazione non validi
407	Manca uno o entrambi i campi di confronto (relazione non valida)
408	Tipo di dati associato al campo specificato non valido per questa operazione
409	Ordine di importazione non valido
410	Ordine di esportazione non valido
412	Per recuperare il file è stata usata una versione errata di FileMaker Pro
413	Tipo di campo non valido
414	Il formato non può visualizzare il risultato
415	Uno o più record correlati richiesti non sono disponibili
416	È necessaria una chiave primaria dalla tabella delle origini dati
417	Il database non è un'origine dati supportata
418	Errore interno all'IMMISSIONE in un campo
500	Il valore della data non soddisfa le opzioni di verifica
501	Il valore dell'ora non soddisfa le opzioni di verifica
502	Il valore del numero non soddisfa le opzioni di verifica
503	Il valore nel campo non è compreso nell'intervallo specificato nelle opzioni di verifica
504	Il valore del campo non è univoco come richiesto dalle opzioni di verifica

Codice errore	Descrizione
505	Il valore del campo non esiste nel file di database come richiesto dalle opzioni di verifica
506	Il valore nel campo non è elencato nella lista di valori specificata nelle opzioni di verifica
507	Il valore nel campo non ha superato il test del calcolo dell'opzione di verifica
508	Valore non valido immesso in modo Trova
509	Il campo richiede un valore valido
510	Valore correlato vuoto o non disponibile
511	Il valore nel campo supera la massima dimensione del campo
512	Il record è già stato modificato da un altro utente
513	Non è stata specificata alcuna verifica ma i dati non sono adatti al campo
600	Errore di stampa
601	La combinazione di intestazione e piè di pagina supera una pagina
602	Il corpo non rientra in una pagina per l'impostazione della colonna corrente
603	Connessione di stampa interrotta
700	Tipo di file errato per l'importazione
706	File EPSF privo di immagine di anteprima
707	Impossibile trovare traduttore per immagini
708	Impossibile importare il file. È necessario un computer a colori
709	Non è riuscita l'importazione del filmato QuickTime
710	Impossibile aggiornare il riferimento al file QuickTime. Il file di database è di sola lettura.
711	Impossibile trovare il traduttore per l'importazione
714	Operazione non consentita dai privilegi della password
715	È stato specificato un foglio di lavoro di Excel o un intervallo con nome mancante
716	Una query SQL che impiega istruzioni DELETE, INSERT o UPDATE non è consentita per l'importazione ODBC
717	Informazioni XML/XSL insufficienti per procedere con l'importazione o l'esportazione
718	Errore di analisi del file XML (da Xerces)
719	Errore di conversione XML usando XSL (da Xalan)
720	Errore durante l'esportazione; il formato desiderato non supporta i campi multipli
721	Errore sconosciuto nel parser o nel convertitore
722	Impossibile importare dati in un file che non ha campi
723	Non si dispone dell'autorizzazione per aggiungere o modificare record nella tabella di destinazione
724	Non si dispone dell'autorizzazione per aggiungere record alla tabella di destinazione
725	Non si dispone dell'autorizzazione per modificare record nella tabella di destinazione
726	Vi sono più record nel file di importazione che nella tabella di destinazione. Non tutti i record sono stati importati
727	Vi sono più record nella tabella di destinazione che nel file di importazione. Non tutti i record sono stati aggiornati
729	Errori durante l'importazione. Impossibile importare i record

Codice errore	Descrizione
730	Versione Excel non supportata (convertire il file al formato Excel 2007/2008 o a una versione successiva supportata e riprovare)
731	Il file da cui state importando non contiene dati
732	Questo file non può essere inserito perché contiene altri file
733	Una tabella non può essere importata in se stessa
734	I file di questo tipo non possono essere visualizzati come immagine
735	I file di questo tipo non possono essere visualizzati come immagine. Verranno inseriti e visualizzati come file
736	Vi sono troppi dati da esportare in questo formato. Sarà troncato.
737	La tabella Bento che si sta importando è assente
738	Il tema importato esiste già
800	Impossibile creare il file su disco
801	Impossibile creare il file temporaneo sul disco di sistema
802	Impossibile aprire il file. Questo errore può essere provocato da una o dalle seguenti cause: <ul style="list-style-type: none"> ■ Nome database non valido ■ Il file è chiuso in FileMaker Server ■ Autorizzazione non valida
803	Il file è per un singolo utente oppure non è stato possibile trovare l'host
804	Impossibile aprire il file.
805	Usare il comando Recupera
806	Impossibile aprire il file con questa versione di FileMaker Pro
807	Il file non è un file FileMaker Pro oppure è gravemente danneggiato
808	Impossibile aprire il file. I privilegi di accesso sono danneggiati
809	Il disco o il volume è pieno
810	Il disco o il volume è protetto
811	Impossibile aprire il file temporaneo come file di FileMaker Pro
812	Capacità dell'host superata
813	Errore di sincronizzazione del record in rete
814	Impossibile aprire i file. È già aperto il numero massimo
815	Impossibile aprire il file di riferimento
816	Impossibile convertire il file
817	Impossibile aprire il file poiché non fa parte di questa soluzione
819	Impossibile salvare una copia locale di un file remoto
820	File in fase di chiusura
821	L'host ha forzato una disconnessione
822	File FMI non trovati; reinstallare i file non presenti
823	Impossibile impostare il file su utente singolo; alcuni ospiti sono connessi
824	Il file è danneggiato o non è un file FileMaker
825	Il file non è autorizzato a fare riferimento al file protetto

Codice errore	Descrizione
826	Il percorso al file specificato non è valido.
850	Il percorso non è valido per il sistema operativo
851	Impossibile eliminare un file esterno dal disco
852	Impossibile salvare il file sulla memoria esterna
853	Impossibile trasferire uno o più Contenitori
900	Errore generico del modulo di gestione del controllo ortografico
901	Dizionario principale non installato
902	Impossibile avviare la Guida
903	Impossibile usare il comando in un file condiviso
905	Non è selezionato nessun campo attivo; il comando può essere usato solo se un campo è attivo
906	File corrente non condiviso; il comando può essere utilizzato solo se il file è condiviso
920	Impossibile inizializzare il modulo di gestione del controllo ortografico
921	Impossibile caricare il dizionario utente per la modifica
922	Impossibile trovare il dizionario utente
923	Il dizionario utente è di sola lettura
951	Errore imprevisto
954	Grammatica XML non supportata
955	Nessun nome per il database
956	È stato superato il numero massimo di sessioni del database
957	Conflitto tra i comandi
958	Parametro mancante nella query
959	La tecnologia Pubblicazione Web personalizzata è disabilitata
960	Il parametro non è valido
1200	Errore di calcolo generico
1201	Troppi pochi parametri nella funzione
1202	Troppi parametri nella funzione
1203	Fine calcolo non previsto
1204	Sono previsti un numero, una costante di testo, un nome di campo o una "("
1205	Il commento non termina con "*" /
1206	La costante di testo deve terminare con un punto interrogativo
1207	Parentesi mancante
1208	Operatore mancante, funzione non trovata o "(" non prevista
1209	Nome (come nome campo o nome formato) mancante
1210	La funzione del plug-in è già stata registrata
1211	Utilizzo della lista valori non consentito in questa funzione
1212	Qui è previsto un operatore (ad esempio, +, -, *)
1213	Questa variabile è già stata definita nella funzione Dichiarata

Codice errore	Descrizione
1214	MEDIO, CONTEGGIO, ESPANSO, RICAVARIPETIZIONI, MAX, MIN, VPN, DEVST, SOMMA e RICAVARIASSUNTO: espressione trovata dove è necessario un campo solo
1215	Questo parametro è un parametro non valido per la funzione Get
1216	Solo i campi Riassunto sono consentiti come primo argomento in RICAVARIASSUNTO
1217	Il campo di separazione non è valido
1218	Impossibile valutare il numero
1219	Non è possibile usare un campo nella propria formula
1220	Il campo deve essere di tipo normale o Calcolo
1221	I dati devono essere di tipo Numero, Data, Ora o Indicatore data e ora
1222	Impossibile memorizzare il calcolo
1223	La funzione a cui si fa riferimento non è ancora implementata
1224	La funzione a cui si fa riferimento non esiste
1225	La funzione a cui si fa riferimento non è supportata in questo contesto
1300	Il nome specificato non può essere utilizzato
1301	Uno dei parametri della funzione importata o incollata ha lo stesso nome della funzione già presente nel file
1400	Errore nell'inizializzazione del driver ODBC; assicurarsi che i driver del client ODBC siano installati correttamente.
1401	Allocazione ambiente fallita (ODBC)
1402	Impossibile liberare l'ambiente (ODBC)
1403	Impossibile disconnettersi (ODBC)
1404	Impossibile allocare la connessione (ODBC)
1405	Impossibile liberare la connessione (ODBC)
1406	Controllo SQL API (ODBC) fallito
1407	Impossibile allocare l'istruzione (ODBC)
1408	Errore esteso (ODBC)
1409	Errore esteso (ODBC)
1410	Errore esteso (ODBC)
1411	Errore esteso (ODBC)
1412	Errore esteso (ODBC)
1413	Errore esteso (ODBC)
1414	L'istruzione SQL è troppo lunga
1450	L'azione richiede un'estensione dei privilegi PHP
1451	L'azione richiede che il file corrente sia remoto
1501	Errore autenticazione SMTP
1502	Connessione rifiutata da server SMTP
1503	Errore SSL
1504	Il Server SMTP richiede che la connessione sia crittografata
1505	Autenticazione specificata non supportata dal server SMTP

Codice errore	Descrizione
1506	Impossibile inviare messaggio/i e-mail
1507	Impossibile accedere al server SMTP
1550	Impossibile caricare il plug-in o plug-in non valido
1551	Impossibile installare il plug-in. Impossibile eliminare un plug-in esistente o scrivere nella cartella o sul disco
1626	Il protocollo non è supportato
1627	Autenticazione non riuscita
1628	Errore con SSL
1629	Connessione scaduta; il valore del timeout è 60 secondi
1630	Il formato URL non è corretto
1631	Impossibile effettuare la connessione

Codici di errore per i componenti PHP

FileMaker API per PHP utilizza diversi componenti PHP. Questi componenti PHP possono restituire altri codici di errore rispetto a quelli sopra esposti.

Ad esempio se i servizi Web Publishing Core o FileMaker Server non sono attivi, può essere inviato l'errore del modulo cURL CURLE_GOT_NOTHING (52).

Per informazioni sui codici di errore PHP correlati, vedere il sito Web PHP <http://php.net>.

Indice

A

- account e privilegi
 - account Ospite 18
 - attivazione per Pubblicazione Web personalizzata 15
 - script 23
- account Ospite
 - attivazione 18
 - disattivazione 18
- add() metodo 40
- Admin Console 14, 15
- Admin di FileMaker Server
 - vedere Admin Console
- attivazione di Pubblicazione Web personalizzata in un database 15

C

- campi Contenitore
 - accesso ai dati da parte degli utenti Web 23
 - con dati memorizzati esternamente 21
 - con file di riferimento 20
 - pubblicazione dei contenuti di 19
- campo
 - anno a quattro cifre 44
 - data 44
 - indicatore data e ora 44
 - non vuoto 43
 - numero massimo di caratteri 43
 - ora 44
 - ora del giorno 44
 - solo numerico 43
- campo anno a quattro cifre 44
- campo data 44
- campo Indicatore data e ora 42, 44
- campo non vuoto 43
- campo numero massimo di caratteri 43
- campo ora 44
- campo ora del giorno 44
- campo solo numerico 43
- cartella Web, copia degli oggetti campo Contenitore 20
- Classe FileMaker 29
- client URL library 12
- codici di errore database 55
- codifica Latin-1 27
- codifica UTF-8 27
- Comando Add 31
- Comando Delete 32
- Comando Duplicate 31
- Comando Edit 31
- Comando Find 39
- Comando Find All 39
- Comando Find Any 39

- Compound Find
 - comando 40
 - esempio 41
- connessione
 - a FileMaker Server 30
 - a un database FileMaker 30
- convalida
 - anno a quattro cifre 44
 - campi 45
 - comandi 43
 - data 44
 - indicatore data e ora 44
 - non vuoto 43
 - numero massimo di caratteri 43
 - ora 44
 - ora del giorno 44
 - record 45
 - solo numerico 43
- convalida Da-a 44
- convalida valore esistente 44
- convalida valore univoco 44
- creazione di un record 31
- crittografia Secure Sockets Layer (SSL) 17
- cURL 12

D

- database, protezione dei database pubblicati 16
- documentazione 6
- documentazione di installazione 6
- documentazione elettronica 6
- Documentazione JDBC 6
- Documentazione ODBC 6
- documentazione online 6
- download progressivo 19, 22
- duplicazione di un record 31

E

- elaborazione di un gruppo di risultati 42
- elaborazione di una richiesta del Motore per la Pubblicazione Web 8
- elemento convalida lista valori 44
- eliminazione di un record 32
- errori
 - codici di errore database 55
 - file registro per server Web 50
 - gestione 47
 - PHP. 62
- errori modulo cURL 62
- esecuzione di richieste di ricerca 38
- Esempi di FileMaker API per PHP 29
- Esercitazioni
 - di FileMaker API per PHP 29
- esercitazioni su FileMaker API per PHP 29

F

file di registro wpe.log 52
 file registro
 accesso server Web 50
 descrizione 50, 52
 Tomcat 53
 web_server_module_log.txt 53
 file registro accessi per server Web, descrizione 50
 FileMaker API per PHP
 definizione 9
 esempi 29
 installazione manuale 13
 FileMaker Server
 documentazione 6
 installazione 6
 FileMakerWebDirect
 definizione 7
 documentazione 6
 filmati QuickTime, pubblicazione sul Web 20
 Finestra di dialogo Impostazione portale 43
 formati 34

G

gestione errori 47
 getContainerData(), metodo 19, 42
 getContainerDataURL(), metodo 21, 42
 getDatabase(), metodo 34
 getErrors(), metodo 47
 getFetchCount(), metodo 42
 getField(), metodo 42
 getFieldAsTimestamp(), metodo 42
 getFields(), metodo 34, 42
 getFoundSetCount(), metodo 42
 getLayout(), metodo 34
 getMessage(), metodo 47
 getName(), metodo 34, 35
 getRange(), metodo 38
 getRecords(), metodo 42
 getRelatedSet(), metodo 35
 getRelatedSets(), metodo 35
 getValueListsTwoFields(), metodo 37
 getValueListTwoFields(), metodo 37
 gruppo di risultati 42

I

Indicatore data e ora Unix 42
 indirizzo IP dinamico 13
 indirizzo IP statico 13
 Informazioni di riferimento
 a FileMaker API per PHP 28
 informazioni di riferimento 28
 installazione di FileMaker API per PHP 13
 installazione manuale del FileMaker API per PHP 13

L

limitazioni ODBC 15
 liste valori 36
 listFields(), metodo 34
 listLayouts(), metodo 34
 listRelatedSets(), metodo 34, 35
 listScripts(), metodo 32
 listValueLists(), metodo 34, 36

M

metod
 numErrors() 45, 47
 metodi
 add() 40
 addSortRule() 38
 clearSortRules() 38
 commit() 31
 createRecord() 31
 delete() 32, 36
 getContainerData() 19, 42
 getContainerDataURL() 21, 42
 getDatabase() 34
 getErrors() 47
 getFetchCount() 42
 getField() 42
 getFieldAsTimestamp() 42
 getFields() 34, 42
 getFoundSetCount() 42
 getLayout() 34
 getMessage() 47
 getName() 34, 35
 getRange() 38
 getRecords() 42
 getRelatedSet() 35
 getRelatedSets() 35
 getValueListsTwoFields() 37
 getValueListTwoFields() 37
 isError() 47
 isValidationError() 45
 listFields() 34
 listLayouts() 34
 listRelatedSets() 34, 35
 listScripts() 32
 listValueLists() 34, 36
 newAddCommand() 31
 newCompoundFindCommand() 40
 newDeleteCommand() 32
 newDuplicateCommand() 31
 newEditCommand() 31
 newFindAllCommand() 39
 newFindAnyCommand() 39
 newFindCommand() 39
 newFindRequest() 40
 newPerformScriptCommand() 33
 newRelatedRecord() 36
 setLogicalOperator() 38
 setOmit() 40
 setPreCommandScript() 33, 38
 setPreSortScript() 33, 38

- setProperty() 30
- setRange() 38
- setRelatedSetsFilters() 43
- setResultsLayout() 34
- setScript() 33, 38
- validate() 45
- metodo addSortRule() 38
- metodo clearSortRules() 38
- metodo commit() 31
- metodo createRecord() 31
- metodo delete() 32, 36
- metodo isError() 47
- metodo isValidatationError() 45
- metodo newAddCommand() 31
- metodo numErrors() 45, 47
- metodo setLogicalOperator() 38
- metodo setOmit() 40
- metodo setPreCommandScript() 33, 38
- metodo setPreSortScript() 33, 38
- metodo setProperty() 30
- metodo setRange() 38
- metodo setRelatedSetsFilters() 43
- metodo setResultsLayout() 34
- metodo setScript() 33, 38
- metodo validate() 45
- modifica di un record 31
- monitoraggio di siti Web 50
- Motore per la Pubblicazione Web
 - codici di errore generati 55
 - descrizione 8
 - elaborazione di una richiesta 8
 - registro dell'applicazione 51

N

- newCompoundFindCommand(), metodo 40
- newDeleteCommand(), metodo 32
- newDuplicateCommand(), metodo 31
- newEditCommand(), metodo 31
- newFindAllCommand(), metodo 39
- newFindAnyCommand(), metodo 39
- newFindCommand(), metodo 39
- newFindRequest(), metodo 40
- newPerformScriptCommand(), metodo 33
- newRelatedRecord(), metodo 36
- nomi utente
 - definizione per la Pubblicazione Web personalizzata 15

O

- oggetti classe FileMaker
 - database 30
 - definizione 30
 - record 31
- oggetti di classe FileMaker
 - set correlato 35

- Oggetti di comando FileMaker
 - Add 31
 - Delete 32
 - Duplicate 31
 - Edit 31
- oggetti di comando FileMaker
 - comando Compound Find 40
 - comando Find 38, 39
 - comando Find All 39
 - comando Find Any 39
- Oggetti di comando Find 38
- oggetto del database 30
- oggetto record 31
- oggetto set correlato 35
- origine dati SQL esterna 15

P

- password
 - definizione per la Pubblicazione Web personalizzata 15
 - password non necessaria per l'accesso 18
 - script Modifica password 18
- PHP
 - attivazione nel database 15
 - riepilogo delle operazioni per la pubblicazione 26
 - risoluzione dei problemi 54
 - test del sito Web 49
 - vantaggi 9
 - versione supportata 13
- PHP 5 12
- PHP API per la Pubblicazione Web personalizzata 9
- portali 35
- pre-convalida
 - anno a quattro cifre 44
 - campi 45
 - comandi 43
 - data 44
 - indicatore data e ora 44
 - non vuoto 43
 - numero massimo di caratteri 43
 - ora 44
 - ora del giorno 44
 - record 45
 - solo numerico 43
- preparazione di siti Web 48
- privilegi di accesso 18
- privilegio esteso per la Pubblicazione Web personalizzata 15
- protezione dei database pubblicati 16
- prove tecniche 54
- pubblicazione PHP 26
- pubblicazione statica, definizione 7
- pubblicazione sul Web
 - codici di errore database 55
 - con PHP 26
 - filmati QuickTime 20
 - oggetti campo Contenitore 19
 - protezione dei database 16

Pubblicazione Web personalizzata
 attivazione nel database 15
 attivazione nel Motore per la Pubblicazione Web 16
 con PHP 9
 con XML 9
 definizione 7
 limitazione accesso degli indirizzi IP nel server Web 16
 privilegio esteso per 15
 script 25
 utilizzo degli script 23
 Pubblicazione Web personalizzata XSLT 9

R

rappresentazione della data 42
 record 31
 Registri Tomcat 53
 registro dell'applicazione 51
 requisiti server 12
 risoluzione dei problemi siti Web
 siti Web Pubblicazione Web personalizzata 49
 risoluzione di problemi dei siti Web
 verifica impostazione 54

S

SAT
 vedere Admin Console
 script
 account e privilegi 23
 Modifica password 18
 nella Pubblicazione Web personalizzata 23
 Nuovo accesso 18
 suggerimenti e considerazioni 23
 trigger 25
 Script Modifica password 18
 Script Nuovo accesso 18
 Server Admin OS X 12
 server Web
 file registro 50
 sessioni di database persistenti 15, 18
 sessioni di database, persistenza 15, 18
 set di privilegi, assegnazione per la Pubblicazione Web personalizzata 15
 sicurezza
 account e password 16
 documentazione 8
 limitazione dell'accesso dagli indirizzi IP 16
 linee guida per la protezione dei database pubblicati 16
 siti Web
 monitoraggio 50
 pagine di supporto per FileMaker 6
 preparazione 48
 risoluzione di problemi 54
 test 49
 streaming. *Vedere* download progressivo
 Strumento Server Admin
Vedere OS X Server Admin

T

test dei siti Web 49
 test della pubblicazione PHP 54
 Trigger 25

U

Unicode 27
 utenti Web
 utilizzo dei dati dei campi Contenitore 23
 utilizzo
 formati 34
 liste valori 36
 portali 35
 record 31
 script 32

V

Vantaggi di XML 9
 verifica mediante calcolo 44

W

web_server_module_log.txt, file registro 53