

Introduzione

Questo manuale è stato concepito come un testo di programmazione per principianti. Non rientra pertanto fra gli scopi di questo libro la trattazione delle tecniche di programmazione avanzate offerte dalle versioni più recenti del linguaggio PHP. La lettura richiede solo conoscenze minime dei concetti generali che ruotano attorno alla programmazione, anche se per comprendere i concetti esposti non è necessaria alcuna precedente esperienza nel campo della programmazione o competenza professionale.

Tutti gli esempi di codice presenti nel libro sono compatibili con PHP 7. La maggior parte di essi è però compatibile anche con PHP 5.6. Sono stati impiegati i metodi (funzioni) più recenti fra quelli disponibili in PHP, almeno al momento della pubblicazione, in modo da offrire al lettore tecniche di programmazione il più possibile aggiornate. Gli esempi utilizzano anche molti metodi di base forniti direttamente dal linguaggio PHP. Il linguaggio consente però di svolgere compiti analoghi in vari modi. Il lettore è pertanto esortato a individuare altri modi in grado di migliorare la sicurezza, le prestazioni e altre funzionalità dei programmi trattati. Il principale scopo di questo libro è quello di stimolare il lettore nella ricerca dei metodi più sicuri ed efficienti per lo sviluppo web. Il codice presentato nel libro fornisce alcuni esempi d'uso di queste tecniche. Occorre però considerare il fatto che *nessun programma può considerarsi assolutamente sicuro*. Il programmatore può solo cercare di realizzare un'applicazione che sia *la più sicura possibile*. Tuttavia è necessario un impegno congiunto del team di sviluppatori, del personale di gestione della rete, degli amministratori della sicurezza, del personale del centro di elaborazione dati e di tutti coloro che hanno a che fare con l'ambiente in cui il software si troverà a operare, ognuno dei quali dovrà contribuire a garantire la sicurezza.

Un approccio differente

Sul mercato vi sono vari manuali dedicati alla programmazione con PHP. Che cosa distingue questo libro?

- Questo libro si basa sul concetto che per imparare sia necessario fare pratica. Pertanto in queste pagine vedremo “nella pratica” come sviluppare applicazioni contenenti istruzioni condizionali, cicli, array e metodi. Negli esempi di codice sono stati introdotti e illustrati oltre settanta metodi (funzioni) PHP.

- Fin dai primissimi esempi, il lettore viene introdotto alle tecniche della programmazione a oggetti. Molti altri libri, invece, descrivono solo brevemente la programmazione a oggetti, e magari solo nei capitoli finali. Fin troppi libri, poi, non affrontano affatto l'argomento.
- Per verificare e filtrare i dati introdotti dall'utente (l'input) vengono utilizzati metodi set a oggetti. In molti altri libri il metodo set viene impiegato unicamente per accettare e memorizzare i dati.
- Un importante obiettivo del libro è quello di indurre il lettore a sviluppare i propri programmi nel modo più sicuro ed efficiente possibile. Pertanto vengono introdotte le tecniche più aggiornate di crittografia della password (`password_hash`).
- Vengono introdotti i metodi try e catch, in grado di lanciare e poi raccogliere tutte le eccezioni e anche alcuni errori. Le versioni più recenti di PHP sono predisposte per gestire le eccezioni e gli errori utilizzando questo approccio. In molti altri libri ci si limita a utilizzare die o altre tecniche per chiudere, semplicemente, il programma in esecuzione.
- La programmazione multi-livello viene introdotta fin dai primi capitoli. Questo consente al lettore di comprendere fin da subito quali tipi di elementi logici e di codice dovranno trovarsi in ciascun livello. Molti libri dedicati a PHP non affrontano neppure questo argomento.
- La maggior parte degli esempi presentati nel libro viene impiegata per sviluppare un'unica grande applicazione (ABC Canine Shelter Reservation System). Nel corso dei capitoli, l'applicazione viene costruita da zero, in più fasi distinte, per esemplificare al lettore il modo in cui può essere suddiviso in più fasi lo sviluppo di una qualsiasi applicazione. Solo nel momento in cui una fase può dirsi completata e ben collaudata lo sviluppo può procedere con la fase successiva. Questo approccio è perfettamente in linea con la progettazione multi-livello. Vengono inoltre proposti utili esercizi di programmazione e un progetto a lungo termine, con lo scopo di migliorare la comprensione delle varie fasi dello sviluppo.
- Vengono introdotti argomenti come la creazione di nuovi utenti, la modifica dei dati degli utenti e la creazione di file log (registri) degli errori. Questo consente al lettore di acquisire competenze nel campo delle funzionalità di backup e ripristino, in modo da mantenere sempre il più possibile operativa l'applicazione, superando brillantemente problemi riguardanti la sicurezza o eventi eccezionali.
- L'introduzione degli oggetti per dati e delle strutture per memorizzare i dati illustra al lettore quale importanza abbia la creazione di un'applicazione in grado di fornire la possibilità di cambiare la tecnica impiegata per la gestione dei dati e la posizione stessa dei dati, senza che ciò richieda importanti riscritture del codice. Vengono forniti esempi XML, JSON e MySQL.
- Nel corso del libro vengono descritte le relazioni naturali esistenti fra il codice PHP, HTML5, CSS3 e JavaScript. Queste relazioni rappresentano uno dei grandi punti di forza di PHP.
- Nel libro vengono forniti numerosi riferimenti in grado di indirizzare l'utente verso ulteriori risorse che gli consentiranno di comprendere meglio il materiale o di approfondire ogni argomento.

Nota speciale rivolta ai docenti

Il modo in cui è strutturato questo libro offre la massima flessibilità in termini di stile e approccio all'insegnamento. Questo perché ogni ente può adottare approcci differenti all'insegnamento della programmazione. Questo libro fornisce tre diversi tipi di esercizi, che consentiranno ai docenti di scegliere l'approccio più adatto. Gli esercizi presentati nel corso dei vari paragrafi consentono agli studenti di mettere in pratica le tecniche acquisite, modificando gli esempi per produrre i risultati desiderati. Questi esercizi aiutano lo studente a sedimentare i concetti appena trattati, prima di affrontare gli esercizi proposti alla fine dei capitoli. Quindi, un progetto a lungo termine, svolto e ampliato di capitolo in capitolo, consente di realizzare un'applicazione parallela che impiega gli stessi tipi di algoritmi e le stesse tecniche di programmazione illustrate nel testo.

Esempi di codice, immagini e link

È stato fatto ogni tentativo di individuare gli errori nel codice e nel testo. Eventuali problemi possono essere segnalati direttamente all'autore steve_prettyman@hotmail.com. Tutti gli esempi di codice, le immagini e i link sono disponibili per il download ai seguenti indirizzi:

- <http://www.apress.com/9781484217290> (a cura dell'editore originale inglese Apress)
- <http://www.littleoceanwaves.com/securephp/> (a cura dell'autore Steve Prettyman).

Panoramica sui capitoli

Capitolo 1 – Introduzione a PHP 7

- Comprendere le differenze fra LAMP, WAMP e MAMP.
- Installare con successo una versione di LAMP, WAMP o MAMP.
- Ricercare in Internet la soluzione di eventuali problemi.
- Spiegare la differenza fra un linguaggio di programmazione e un linguaggio per script.
- Creare un semplice programma PHP esente da errori.

Capitolo 2 – Interfacce, piattaforme, container e programmazione a tre livelli

- Fornire esempi di piattaforme o container che possono ospitare programmi PHP.
- Creare una semplice applicazione web dinamica utilizzando PHP.
- Descrivere lo schema progettuale a tre livelli e determinare il contenuto di ciascun livello.
- Progettare un'applicazione a tre livelli.

- Spiegare ogni fase dello sviluppo di un programma.
- Definire e spiegare i concetti di MVC e di Dependency Injection.

Capitolo 3 – Programmazione modulare

- Creare un semplice programma modulare PHP orientato agli oggetti ed esente da errori.
- Creare una classe PHP e poi creare un'istanza di tale classe (un oggetto).
- Creare un programma PHP incapsulato orientato agli oggetti, comprendente metodi GET e SET.
- Creare metodi PHP (funzioni) che accettino parametri e restituiscano informazioni.
- Creare proprietà pubbliche e private PHP (variabili).
- Importare codice PHP da un altro file o da una libreria all'interno di un programma.
- Convalidare le informazioni ricevute utilizzando operatori ternari (condizionali).

Capitolo 4 – La sicurezza dell'interfaccia utente

- Spiegare perché l'input dell'utente deve essere convalidato nei livelli dell'interfaccia e delle regole operative.
- Spiegare perché l'input dell'utente debba essere filtrato nel livello delle regole operative.
- Utilizzare codice HTML5 per convalidare l'input dell'utente.
- Utilizzare codice JavaScript per convalidare l'input dell'utente.
- Utilizzare istruzioni PHP `if` (ovvero istruzioni condizionali) per convalidare e filtrare l'input.
- Utilizzare cicli `foreach` per creare dinamicamente una casella di selezione HTML a partire da un file XML.
- Utilizzare semplici array per il filtraggio e la convalida.
- Passare semplici array nei metodi (funzioni).
- Comprendere come utilizzare la Dependency Injection per controllare le modifiche alle versioni del codice.

Capitolo 5 – Gestione e logging delle eccezioni

- Spiegare la differenza esistente fra errori ed eccezioni.
- Creare un programma PHP in grado di gestire eccezioni generali.
- Creare un programma PHP in grado di creare, lanciare e gestire eccezioni personalizzate.
- Spiegare e utilizzare un'istruzione `switch` e/o un costrutto `if/else embedded`.
- Creare un programma PHP che utilizzi il ciclo `while` e/o `for`.

- Creare un programma che legga/aggiorni un file di testo utilizzando un array bi-dimensionale.
- Creare un programma PHP che registri su file log le eccezioni e poi le invii per posta elettronica al personale incaricato del supporto tecnico.

Capitolo 6 – Oggetti per la gestione dei dati

- Creare una classe per i dati, che inserisca, aggiorni e cancelli dati in file XML o JSON.
- Creare una classe per dati che aggiorni dati in database MySQL utilizzando uno script SQL.
- Creare un programma PHP che crei un file log degli interventi.
- Creare un programma PHP che ripristini i dati da un backup.
- Applicare delle modifiche per creare informazioni valide e aggiornate.
- Usare la Dependency Injection per collegare una classe per dati a un'altra classe del livello delle regole operative.
- Creare un'applicazione PHP a tre livelli.

Capitolo 7 – Autenticazione

- Definire le sessioni e spiegare come vengono utilizzate per l'autenticazione.
- Creare un programma PHP in grado di autenticare gli utenti al login.
- Creare un programma PHP che registri gli utenti.
- Creare un programma PHP che consenta agli utenti di cambiare la propria password.
- Creare un programma PHP che registri i tentativi di login non validi.
- Creare un programma PHP che utilizzi tecniche di crittografia della password.

Capitolo 8 – Interfacce multifunzionali

- Creare un'intera applicazione PHP che cancelli, aggiorni e inserisca dei dati.
- Creare un'applicazione dall'aspetto professionale utilizzando stili CSS.
- Utilizzare JavaScript per accettare e manipolare i dati di un altro programma.
- Mettere al sicuro i programmi di un'applicazione richiedendo nome-utente e password.
- Compilare gli oggetti HTML con valori contenuti in un oggetto JSON.
- Creare un programma PHP che utilizzi tecniche aggiornate di crittografia della password.