

Introduzione

Questo è un libro di programmazione di livello principiante e intermedio. Non è l'obiettivo di questo libro coprire le tecniche avanzate nelle attuali versioni del linguaggio di programmazione PHP. È prevista una certa conoscenza dei concetti generali di programmazione, ma non si presume alcuna formazione o esperienza specifica. Tutti gli esempi di codice in questo libro sono compatibili con PHP 8. La maggior parte degli esempi è compatibile con PHP 7. I metodi (funzioni) più recenti disponibili in PHP (alla data di pubblicazione) sono stati utilizzati per fornire al lettore le tecniche di codifica più avanzate. Gli esempi utilizzano metodi di base forniti nel linguaggio PHP. Questo linguaggio include molti metodi aggiuntivi per eseguire attività simili. Il lettore può, e dovrebbe, ricercare ulteriori modi per migliorare la sicurezza, le prestazioni e altre tecniche. L'obiettivo di questo libro è incoraggiare gli utenti a considerare sempre l'utilizzo dei metodi più sicuri ed efficienti di sviluppo del programma. Il codice contenuto in questo libro fornisce alcuni esempi di utilizzo di queste tecniche. L'utente deve ricordare che nessun programma è sicuro al 100%. Il programmatore può solo sforzarsi di rendere un'applicazione il più sicura possibile. Ci vuole la collaborazione di un team di sviluppatori, personale di rete, amministratori della sicurezza, personale del data center e altri per fornire l'ambiente più sicuro.

Un approccio differente

Oggi sul mercato ci sono numerosi libri su PHP. Che cosa rende questo diverso dagli altri?

- Questo libro utilizza il concetto di “imparare facendo”, che mostra al lettore come sviluppare applicazioni con istruzioni condizionali, cicli, array e metodi. Oltre 70 metodi PHP (funzioni) vengono introdotti e dimostrati in esempi di programmazione.
- All'inizio del libro, il lettore viene introdotto alle tecniche di programmazione a oggetti (OO). Molti altri libri trattano solo brevemente (o per nulla) la programmazione OO nei capitoli finali.
- I metodi di insiemi orientati agli oggetti vengono utilizzati per verificare e filtrare l'input dell'utente. Molti altri libri mostrano semplicemente un metodo prestabilito che accetta i dati e li memorizza.

- Uno degli obiettivi principali del libro è portare il lettore a creare tutti i programmi nel modo più sicuro ed efficiente possibile. Vengono mostrate le più recenti tecniche di crittografia delle password (`password_hash`).
- I metodi `try` e `catch` vengono introdotti per acquisire eccezioni e alcuni errori. Le versioni più recenti di PHP sono state create per gestire eccezioni ed errori utilizzando questo approccio.
- La progettazione del programma multilivello viene introdotta nei primi capitoli. Ciò consente al lettore di scoprire quale logica e codifica dovrebbero aver luogo in ogni livello. Molti libri su PHP non trattano nemmeno questo argomento.
- La maggior parte degli esempi nel libro viene utilizzata per sviluppare un'applicazione principale (ABC Canine Shelter Reservation System). Man mano che si procede nella lettura, l'applicazione viene creata dall'inizio, per fasi, mostrando al lettore come lo sviluppo dovrebbe essere suddiviso. Solo dopo che ogni fase è stata completata e testata può iniziare la fase successiva. Questo approccio funziona di pari passo con la progettazione multilivello. Ulteriori esercizi di programmazione e un progetto finale sono forniti per migliorare la comprensione dello sviluppo.
- Viene introdotta la creazione di registri utente, modifiche ed errori. Ciò consente al lettore di comprendere come fornire capacità di backup e ripristino per mantenere un'applicazione funzionante correttamente quando si verificano violazioni della sicurezza o eccezioni.
- L'introduzione degli oggetti per i dati e del livello dei dati dimostra al lettore l'importanza di creare un'applicazione che offra la possibilità di modificare le tecniche di archiviazione dei dati e la posizione di archiviazione dei dati senza richiedere una riscrittura importante dell'applicazione. Vengono forniti esempi XML, JSON e MySQL.
- Una relazione naturale fra PHP, HTML5, CSS3 e JavaScript è illustrata in tutto il libro. Questa relazione è uno dei principali punti di forza di PHP.
- In tutto il libro, vengono forniti riferimenti per indirizzare il lettore ad approfondire gli argomenti trattati.

Nota speciale per gli insegnanti

La struttura di questo libro consente flessibilità negli stili e negli approcci di insegnamento. Ogni università si avvicina alla formazione iniziale dei concetti di programmazione in modi diversi. Questo libro fornisce tre diversi tipi di esercizi di programmazione, che consentono agli insegnanti di scegliere il migliore per il loro ambiente. In ogni capitolo sono forniti esercizi per consentire allo studente di acquisire esperienza pratica con le tecniche mostrate modificando gli esempi esistenti per produrre i risultati desiderati. Questi esercizi permettono agli studenti di raggiungere un certo livello di competenza prima di provare a programmare gli esercizi alla fine dei capitoli. Inoltre, viene fornito un progetto finale che consiste nel costruire un'applicazione che utilizza gli stessi tipi di algoritmi e tecniche di programmazione mostrati nel libro.

File degli esempi

È stato compiuto ogni sforzo per rilevare eventuali errori nel codice (e nella grammatica). Fateci sapere se/quando riscontrate problemi in questo libro. Si prega di inviare tutte le correzioni a Steve Prettyman (steve_prettyman@hotmail.com).

I file degli esempi sono disponibili all'indirizzo <https://github.com/Apress/learn-php-8>. La copia del codice dal libro può causare errori a causa dei requisiti di formato per la pubblicazione.

Panoramica dei capitoli

Capitolo 1 – Introduzione a PHP 8

Dopo aver completato questo capitolo, sarete in grado di:

- conoscere le differenze esistenti fra LAMP, WAMP e MAMP;
- installare correttamente una versione di LAMP, WAMP o MAMP;
- scoprire in Internet come risolvere i problemi;
- spiegare la differenza esistente fra un linguaggio di programmazione e un linguaggio per script;
- creare un semplice programma PHP senza errori.

Capitolo 2 – Interfacce, piattaforme e programmazione a tre livelli

Dopo aver completato questo capitolo, sarete in grado di:

- fornire esempi di piattaforme o container che possono ospitare programmi PHP;
- creare un'applicazione web semplice e dinamica, usando PHP;
- spiegare in che cosa consiste la progettazione a tre livelli e determinare che cosa contiene ogni livello;
- progettare un'applicazione a tre livelli;
- spiegare ogni fase del ciclo di vita di sviluppo del programma;
- definire e spiegare che cosa significano i termini MVC e dependency injection.

Capitolo 3 – Le basi: la sintassi di PHP 8

Dopo aver completato questo capitolo, sarete in grado di:

- creare un programma PHP semplice e senza errori;
- comprendere l'uso e il valore delle istruzioni condizionali;
- comprendere l'uso e il valore dei cicli `for`, `while` e `foreach`;
- comprendere l'uso e il valore delle funzioni;
- comprendere l'uso e il valore degli array.

Capitolo 4 – Programmazione modulare

Dopo aver completato questo capitolo, sarete in grado di:

- creare un semplice programma PHP modulare a oggetti privo di errori;
- creare una classe PHP e poi un'istanza della classe (un oggetto);
- creare un programma PHP incapsulato a oggetti, comprendente i metodi *getter* e *setter*;
- creare metodi (funzioni) PHP che accettano parametri e restituiscono valori;
- creare proprietà (variabili) PHP pubbliche e private;
- importare in un programma codice PHP da un altro file o da una libreria;
- convalidare le informazioni ricevute utilizzando operatori ternari (condizionali).

Capitolo 5 – Interfacce utente sicure

Dopo aver completato questo capitolo, sarete in grado di:

- spiegare perché l'input dell'utente deve essere convalidato nei livelli dell'interfaccia e delle regole operative;
- spiegare perché l'input dell'utente deve essere filtrato nel livello delle regole operative;
- usare codice HTML5 e JavaScript per convalidare l'input dell'utente;
- usare istruzioni *if* (istruzioni condizionali) in PHP per convalidare e filtrare l'input;
- usare cicli *foreach* per creare dinamicamente una casella di selezione HTML da un file XML;
- usare semplici array per il filtraggio e la convalida;
- passare semplici array ai metodi (funzioni);
- impiegare la *dependency injection* per controllare le modifiche alla versione del codice.

Capitolo 6 – Gestione e registrazione delle eccezioni

Dopo aver completato questo capitolo, sarete in grado di:

- spiegare la differenza fra errori ed eccezioni;
- creare un programma PHP in grado di gestire le eccezioni generiche;
- creare un programma PHP in grado di creare, lanciare e gestire eccezioni;
- spiegare e utilizzare un'istruzione *switch* e/o un'istruzione *if/else* annidata;
- creare un programma PHP che utilizzi un 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 le eccezioni e invii e-mail al personale di supporto.

Capitolo 7 – Oggetti per i dati

Dopo aver completato questo capitolo, sarete in grado di:

- creare una classe per dati che inserisca, aggiorni ed elimini dati XML o JSON;
- spiegare come si crea una classe per dati che aggiorni i dati MySQL utilizzando uno script SQL;
- creare un programma PHP che crei un registro di backup delle modifiche;
- creare un programma PHP in grado di recuperare i dati da un backup precedente;
- applicare le modifiche per creare informazioni valide e aggiornate;
- utilizzare la dependency injection per collegare una classe per dati a un'altra classe nel livello delle regole operative;
- creare un'applicazione PHP a tre livelli.

Capitolo 8 – Autenticazione

Dopo aver completato questo capitolo, sarete in grado di:

- definire il significato delle sessioni e spiegare come utilizzarle per l'autenticazione;
- creare un programma PHP che autentichi gli accessi degli utenti;
- creare un programma PHP che registri gli utenti;
- creare un programma PHP che consenta agli utenti di modificare la propria password;
- creare un programma PHP che registri i tentativi di accesso non validi;
- creare un programma PHP che utilizzi le attuali tecniche di hashing delle password.

Capitolo 9 – Interfacce multifunzionali

Dopo aver completato questo capitolo, sarete in grado di:

- creare un'applicazione PHP completa in grado di eliminare, aggiornare e inserire dati;
- dare un aspetto professionale a un'applicazione utilizzando gli stili CSS;
- usare JavaScript per accettare dati da un altro programma;
- proteggere tutti i programmi di un'applicazione, richiedendo ID utente e password;
- popolare degli oggetti HTML con valori tratti da un oggetto JSON;
- creare un programma PHP che utilizzi le più recenti tecniche di crittografia delle password.

Il revisore tecnico

Satej Kumar Sahu è Senior Enterprise Architect presso Honeywell. È appassionato di tecnologie, persone e natura. Crede che attraverso la tecnologia e un processo decisionale coscienzioso ognuno di noi abbia il potere di rendere questo mondo un luogo migliore. Nel suo tempo libero legge, gioca a basket e si diverte con amici e familiari.

Ringraziamenti

Grazie a tutti coloro che hanno contribuito a realizzare questo libro. Un ringraziamento speciale va a tutti i miei studenti che hanno cominciato a studiare PHP: loro sono stati i veri tester e debugger di questo viaggio.

Un riconoscimento speciale va a tutti gli sviluppatori open source e a tutti coloro che forniscono tutorial e formazione a titolo gratuito a ogni utente che voglia saperne di più sulla programmazione.