

Introduzione

Questo libro riguarda il framework ASP.NET Core: spiega che cos'è e come utilizzarlo per realizzare applicazioni web. Anche se alcuni di questi contenuti sono già disponibili online, il tutto si trova disperso in Internet all'interno di svariati documenti e blog. Questo libro vi guiderà nella realizzazione delle vostre prime applicazioni, introducendo progressivamente sempre nuovi argomenti, a mano a mano che i precedenti si saranno sedimentati.

Presento ogni argomento impiegando esempi relativamente compatti, invece di costruire un'unica grossa applicazione nel corso del libro. Entrambi gli approcci presentano vantaggi, ma ho preferito assicurarmi che l'argomento restasse sempre circoscritto, evitando il carico mentale di dover elaborare un progetto sempre più ampio.

Entro la fine del libro, avrete una solida conoscenza di come si realizzano app con ASP.NET Core, dei punti di forza e dei punti deboli del framework e di come sfruttare le sue funzionalità per realizzare app sicure. Anche se non ho dedicato molto spazio agli aspetti di architettura di un'applicazione, mi sono premurato di esporre le migliori pratiche, specialmente dove ho sorvolato sull'architettura per non dilungarmi troppo.

A chi è rivolto questo libro

Questo libro è dedicato agli sviluppatori C# interessati ad apprendere un framework web multiplatforma. Non è richiesta alcuna esperienza di programmazione di applicazioni web (per esempio potete essere sviluppatori di applicazioni mobile o desktop) ma un po' di esperienza con ASP.NET o un altro framework per il Web sarà indubbiamente utile. A parte una certa conoscenza operativa di C# e .NET, mi aspetto sappiate impiegare la programmazione a oggetti e che abbiate una conoscenza di base dei database relazionali, almeno in generale. Immagino che abbiate una certa familiarità anche con HTML e CSS, e del ruolo di JavaScript come linguaggio per script client-side. Ma non avete bisogno di conoscere un framework JavaScript o CSS per seguire questo libro, sebbene ASP.NET Core funzioni benissimo con entrambi.

I framework web comprendono, naturalmente, un'ampia varietà di argomenti, dai database alle reti, alla programmazione visuale, agli script client-side. Cercherò di fornire il maggior contesto possibile e includerò indirizzi web e libri utili per approfondire gli argomenti.

Come è organizzato questo libro

Questo libro è suddiviso in tre parti, ventitré capitoli e due appendici. Idealmente, dovrete leggere il libro da cima a fondo e poi utilizzarlo come testo di riferimento, ma mi rendo conto che non tutti possono farlo. Anche se uso piccole app di esempio per illustrare un argomento, alcuni capitoli si fondano sui precedenti e, pertanto, è opportuno che i contenuti vengano letti in modo sequenziale.

Vi consiglio, soprattutto, di leggere in sequenza i capitoli della Parte I, in quanto ciascun capitolo estende gli argomenti introdotti nei capitoli precedenti. Anche la Parte II andrebbe letta in modo sequenziale, sebbene la maggior parte dei capitoli abbia una natura più indipendente e quindi possiate muovervi con maggiore libertà, senza timore di perdervi. I capitoli della Parte III possono essere letti nell'ordine preferito, anche se consiglio di affrontarli solo dopo aver letto i capitoli delle Parti I e II.

La Parte I fornisce un'introduzione generale ad ASP.NET Core e all'architettura di una tipica applicazione web. Dopo aver trattato le basi, approfondiremo il framework Razor Pages, che costituisce il nucleo centrale della maggior parte delle applicazioni web realizzate con ASP.NET Core su server e la relativa architettura MVC (*Model-View-Controller*).

- Il Capitolo 1 introduce ASP.NET Core e lo colloca nel panorama dello sviluppo web. Spiega quando usare e non utilizzare ASP.NET Core, le basi delle richieste web in ASP.NET Core e le opzioni disponibili per un ambiente di sviluppo.
- Il Capitolo 2 esamina tutti i componenti di una semplice applicazione ASP.NET Core, descrivendo il loro ruolo e come essi si combinano per generare una risposta a una richiesta web.
- Il Capitolo 3 descrive la pipeline del middleware, la struttura principale di un'applicazione in ASP.NET Core. Essa definisce il modo in cui vengono elaborate le richieste in arrivo e il modo in cui deve essere generata una risposta.
- Il Capitolo 4 mostra come usare Razor Pages per realizzare siti web basati su pagine. Razor Pages è il modo migliore per realizzare applicazioni supportate dal server in ASP.NET Core e le sue pagine sono concepite per le applicazioni basate su pagine.
- Il Capitolo 5 descrive il sistema di routing di Razor Pages. Il routing è il processo di mappatura degli URL della richiesta in arrivo a una determinata classe e a un suo metodo, il quale poi viene eseguito per generare una risposta.
- Il Capitolo 6 parla del binding del modello: il processo di mappatura fra i dati dei moduli e i parametri URL passati in una richiesta da un lato e, dall'altro, oggetti C# concreti.
- Il Capitolo 7 mostra come generare pagine web HTML utilizzando il linguaggio per template Razor.
- Il Capitolo 8 si sviluppa a partire dal Capitolo 7 introducendo i Tag Helper, che possono ridurre enormemente la quantità di codice necessario per realizzare moduli e pagine web.
- Il Capitolo 9 spiega come usare i controller MVC per realizzare API che possono essere richiamate da app client-side.

La Parte II tratta alcuni argomenti importanti per la realizzazione di intere applicazioni web, dopo aver introdotto tutte le basi.

- Il Capitolo 10 descrive come usare il container DI di ASP.NET Core per configurare i servizi dell'applicazione.
- Il Capitolo 11 spiega come leggere le impostazioni e i segreti in ASP.NET Core e come mapparli su oggetti a tipizzazione forte.
- Il Capitolo 12 introduce Entity Framework Core, per il salvataggio dei dati su un database relazionale.
- Il Capitolo 13 sviluppa gli argomenti della Parte I introducendo MVC e la pipeline di filtraggio di Razor Pages.
- Il Capitolo 14 spiega come aggiungere all'applicazione i profili utente e l'autenticazione, impiegando ASP.NET Core Identity.
- Il Capitolo 15 sviluppa il capitolo precedente introducendo l'autorizzazione per gli utenti, per poter limitare le pagine cui gli utenti registrati hanno accesso.
- Il Capitolo 16 mostra come pubblicare l'app, come configurare l'app per un ambiente di produzione e come ottimizzare gli asset client-side.

I capitoli che compongono la Parte III trattano alcuni aspetti importanti e trasversali dello sviluppo in ASP.NET Core.

- Il Capitolo 17 mostra come configurare il logging nell'applicazione e come scrivere messaggi log in più posizioni.
- Il Capitolo 18 esplora alcune considerazioni di sicurezza da considerare quando si sviluppa un'applicazione, fra cui la configurazione dell'applicazione per HTTPS.
- Il Capitolo 19 spiega come realizzare e impiegare vari componenti custom, come il middleware, e come gestire requisiti di configurazione complessi.
- Il Capitolo 20 sviluppa il Capitolo 19 e mostra come realizzare componenti Razor Page custom, come Tag Helper e attributi di convalida.
- Il Capitolo 21 parla del nuovo `IHttpClientFactory` e spiega come usarlo per creare istanze di `HttpClient` per la chiamata di API remote.
- Il Capitolo 22 esplora l'astrazione generica `IHost`, che potete utilizzare per creare servizi Windows e daemon Linux. Imparerete anche a eseguire task in background per le vostre applicazioni.
- Il Capitolo 23 mostra come sottoporre a test un'applicazione ASP.NET Core con il framework di testing xUnit. Il capitolo tratta sia gli unit test sia gli integration test con Test Host.

Le due appendici forniscono alcune informazioni supplementari.

- L'Appendice A spiega come configurare l'ambiente di sviluppo, in Windows, Linux o macOS.
- L'Appendice B fornisce alcune basi di .NET 5.0, .NET Core e .NET Standard, spiega come si collocano nel panorama .NET e descrive il significato che hanno per le app.

Il codice

Questo libro contiene molti esempi di codice sorgente in listati numerati o in linea nel testo. In entrambi i casi, il codice sorgente è formattato con un font a larghezza fissa per distinguerlo dal testo circostante. Alcune parti dei listati sono anche **in grassetto** per evidenziare il codice che è stato modificato rispetto ai passi precedenti del capitolo, come nel caso dell'aggiunta di una nuova funzionalità a una riga di codice esistente.

In molti casi, il codice sorgente originale è stato riformattato; ho aggiunto righe vuote e ho rielaborato l'indentazione adattandolo allo spazio disponibile nella pagina del libro. Inoltre, ho rimosso i commenti originariamente presenti nel codice sorgente, quando il codice si trova già descritto nel testo. Molti listati sono seguiti da annotazioni numerate che sottolineano i concetti più importanti.

Il codice sorgente è presente in tutti i capitoli tranne il Capitolo 1. Potete scaricare il codice sorgente di ciascun capitolo dal mio repository GitHub in <https://github.com/andrewlock/asp-dot-net-core-in-action-2e>. Troverete un file compresso contenente tutto il codice sorgente anche sul sito web di Apogeo all'indirizzo <https://bit.ly/apo-anc>.

Tutti gli esempi di codice di questo libro impiegano .NET 5.0 e sono stati compilati in Visual Studio e Visual Studio Code. Per compilare ed eseguire gli esempi, dovrete installare .NET SDK, come descritto nell'Appendice A.

L'autore

Andrew Lock si è laureato in Ingegneria presso la Cambridge University, con specializzazione in Software Engineering e successiva laurea in Digital Image Processing. È stato sviluppatore professionista in .NET per oltre dieci anni, impiegando un'ampia varietà di tecnologie, fra cui WinForms, ASP.NET WebForms, ASP.NET MVC e ASP.NET Webpages. Ha realizzato e curato applicazioni con ASP.NET Core fin dal rilascio della versione 1.0, nel 2016. Andrew gestisce un blog molto attivo in <https://andrewlock.net> dedicato ad ASP.NET Core. È spesso coinvolto dalla community del team Microsoft ASP.NET, sul blog .NET e negli incontri settimanali della community. Andrew è *Microsoft Valued Professional* (MVP) fin dal 2017.