

Introduzione

Il Web oggi è diverso, così come è diverso il modo in cui si programma per il Web. A fronte delle difficoltà che si incontrano nel trattare il codice imperativo (di ardua manutenzione) prodotto da jQuery, occorre trovare nuovi modi per gestire la complessità delle moderne interfacce utente. Serve una nuova libreria che aiuti a realizzare applicazioni front-end dichiarative, modulari, veloci e scalabili utilizzando JavaScript.

React.js è una libreria JavaScript sviluppata da Facebook che introduce idee interessanti su come lavorare con il DOM, organizzare il flusso di dati dell'applicazione e pensare gli elementi dell'interfaccia utente come singoli componenti. Certo, è una libreria che non fa alcuna supposizione sul resto dello stack di tecnologie, ma unita a Flux consente di ottenere una potente architettura front-end utile non solo agli sviluppatori esperti, ma anche a chi è all'inizio del proprio percorso. Gli sviluppatori front-end di qualsiasi livello di esperienza, che risolvono ogni tipo di difficoltà aziendale all'interno di team di tutte le dimensioni e con scadenze di varia urgenza, possono dare finalmente il benvenuto a un futuro migliore.

La semplicità, la prevedibilità e l'attenzione ai dettagli di React.js sono davvero stupefacenti.

Gli argomenti del libro

Il Capitolo 1, "Installazione degli strumenti", delinea l'obiettivo di questo libro e spiega quali strumenti occorre installare per creare le applicazioni React con la massima efficienza. Ogni strumento viene presentato con istruzioni sulla relativa modalità di installazione. Poi verrà creata una struttura per il progetto che realizzeremo nel corso del libro.

Nel Capitolo 2, "Creazione del primo elemento React", è spiegato come installare React. Viene poi presentato il DOM virtuale, con una spiegazione degli

elementi React comprensiva delle modalità di creazione e rendering utilizzando la sintassi JavaScript nativa. Infine, viene introdotta la sintassi JSX e verrà mostrato come creare elementi React con JSX.

Il Capitolo 3, “Creazione del primo componente React”, presenta i componenti React spiegando la differenza tra i componenti con stato e senza stato, affinché sia possibile decidere quali utilizzare. Il testo guida inoltre nella creazione di entrambi i tipi di componenti.

Il Capitolo 4, “Reattività dei componenti React”, descrive come risolvere un problema con React e accompagna il lettore nella pianificazione dell’applicazione React. Nel capitolo viene creato un componente React che incapsula l’intera applicazione React che sarà realizzata nel libro; viene inoltre spiegata la relazione tra componenti React padre e figlio.

Nel Capitolo 5, “Uso dei componenti React con un’altra libreria”, viene esaminato l’uso delle librerie JavaScript di terze parti con i componenti React. Viene inoltre presentato il ciclo di vita dei componenti React, con una spiegazione dei metodi di mounting e la creazione di nuovi componenti React per il progetto del libro.

Il Capitolo 6, “Aggiornamento dei componenti React”, descrive i metodi di aggiornamento del ciclo di vita dei componenti React. Viene spiegato come utilizzare gli stili CSS in JavaScript e come convalidare e impostare le proprietà predefinite dei componenti.

Il Capitolo 7, “Creazione di componenti React complessi”, è incentrato sulla realizzazione di componenti React più complessi. Vengono esaminati i dettagli dell’implementazione, nonché l’integrazione dei vari componenti in un’applicazione React coerente e pienamente funzionale.

Nel Capitolo 8, “Test dell’applicazione React con Jest”, è spiegata l’idea alla base degli unit test; viene inoltre descritto come si scrivono ed eseguono unit test con Jest, nonché come testare i componenti React. Il capitolo contiene inoltre una descrizione dettagliata di gruppi di test, specifiche, aspettative e funzioni matcher.

Il Capitolo 9, “Potenziamento dell’architettura di React con Flux”, spiega come migliorare l’architettura delle applicazioni React, presentando l’architettura Flux e spiegando il ruolo di dispatcher, archivi e creatori di azioni.

Il Capitolo 10, “Preparazione per la manutenzione semplificata con Flux”, descrive come separare le problematiche dell’applicazione React utilizzando Flux. L’applicazione React viene sottoposta a refactoring per garantire la facilità di manutenzione futura.

Che cosa serve per il libro

Per prima cosa è necessaria l'ultima versione di un browser web moderno, come Google Chrome o Mozilla Firefox:

- Google Chrome: <https://www.google.com/chrome/browser>
- Mozilla Firefox: <https://www.mozilla.org/en-US/firefox/new/>

In secondo luogo è necessario installare Git, Node.js e npm. Istruzioni dettagliate sull'installazione e sull'uso di questi strumenti sono disponibili nel Capitolo 1, "Installazione degli strumenti".

Per finire, avrete bisogno di un editor di codice: l'autore consiglia Sublime Text (<http://www.sublimetext.com>), ma si può utilizzare anche Atom (<https://atom.io>), Brackets (<http://brackets.io>), Visual Studio Code (<https://code.visualstudio.com>) o qualsiasi altro editor.

A chi si rivolge il libro

Questo libro è destinato agli sviluppatori front-end che desiderano costruire interfacce utente scalabili e gestibili per il Web. Tutto ciò che serve per iniziare a trarre vantaggio dalle idee rivoluzionarie che React.js porta nel mondo dello sviluppo web è una discreta conoscenza di base di JavaScript, HTML e CSS. Chi ha precedenti esperienze con jQuery o Angular.js riuscirà a cogliere più facilmente la diversità di React.js e a capire come approfittare dell'integrazione di diverse librerie con React.

Convenzioni

In questo libro sono presenti numerosi stili di testo che distinguono i diversi tipi di informazioni. Ecco alcuni esempi di questi stili e una spiegazione del loro significato.

Il codice nel testo, i nomi delle tabelle di database, i nomi delle cartelle, i nomi dei file, le estensioni dei file, i percorsi, gli URL fittizi, l'input dell'utente e gli handle di Twitter sono presentati in *monospaziato*. Per esempio: "Il punto di ingresso alla libreria React è l'oggetto `React`".

Un blocco di codice è formattato come segue:

```
var React = require('react');  
var ReactDOM = require('react-dom');
```

```
var reactElement = React.createElement(
  'h1',
  { className: 'header' }
);
ReactDOM.render(
  reactElement,
  document.getElementById('react-application')
);
```

Quando voglio richiamare l'attenzione su una particolare sezione di un blocco di codice, le righe o gli elementi in questione sono in grassetto:

```
<!doctype html>
<html lang="en">
  <head>
    <title>Snapterest</title>
  </head>
  <body>
    <div id="react-application">
      I am about to learn the essentials of React.js.
    </div>
    <script src="./snapterest.js"></script>
  </body>
</html>
```

Gli input e output da riga di comando sono presentati nel modo seguente:

```
cd ~
git clone https://github.com/snapkite/snapkite-engine.git
```

Termini nuovi, parole importanti, cartelle o directory ed elementi dell'interfaccia sono evidenziati in *corsivo*.

NOTA

I suggerimenti, gli avvertimenti e le note importanti appaiono in questo modo.

Scarica i file degli esempi

Sul sito dell'editore originale inglese, Packt Publishing, potete scaricare i file degli esempi presentati del testo. Per farlo è necessario registrarsi gratuitamente all'indirizzo <https://www.packtpub.com/register>. Quindi andate sulla scheda del libro

all'indirizzo <https://www.packtpub.com/web-development/react-essentials> e fate clic su *Code Files*.

GitHub

Per questo libro i file del codice sorgente possono essere scaricati anche dal repository GitHub indicato qui di seguito:

<https://github.com/fedosejev/react-essentials>

Sempre da GitHub è inoltre possibile segnalare problemi e dubbi sul codice degli esempi, contattare l'autore e trovare eventuali Errata. L'indirizzo è:

<https://github.com/fedosejev/react-essentials/issues>