Alma Mater Studiorum
Università di Bologna

Scuola di Ingegneria

Tecnologie Web T

Esercitazione 08
Node.js

Home Page del corso: http://lia.disi.unibo.it/Courses/twt2223-info/
Versione elettronica: L.08.NodeJS.pdf
Versione elettronica: L.08.NodeJS-2p.pdf

Agenda

- Breve Riepilogo
- Node.js: Set Up dell’ambiente
- Primi Passi con Node
- Il primo Server: Hello World!!
- Esercizio 1: Conta Parole
- Esercizio 1.5: Conta Righe, Parole, e Parole su Righe
- Esercizio 2: Multi File
- Esercizio 3: Elimina Parola Parametrico

Node in a nutshell

- Utilizzo dello stesso JavaScript engine sia lato browser che servitore
  - ➔Ovviamente senza bisogno di DOM lato server
- Gestione eventi su una coda degli eventi
  - ➔Ogni operazione esegue come una chiamata dall'event loop
- Uso di interfaccia ad eventi per ogni operazione di SO
  - ➔Wrapping di tutte le chiamate bloccanti di SO (I/O su file e socket/network)
- Uso di sistema di moduli (import/export)
  - ➔Moduli specializzati per il supporto a data management efficiente

Node: Set Up dell'Ambiente

In laboratorio tutto è già stato impostato da noi, ma a casa?

Iniziamo a scaricare Node.js da qui (multipiattaforma):

https://nodejs.org/it/download/

Qui potrete anche trovare:

- Documentazione
- Esempi
- Guide

Primi Passi con Node

Vi è stato fornito nello starting kit un primo esempio molto semplice di server Node

- *Crea un semplice Server Web che genera una pagina Hello World*
- *Carica il modulo **http***
- *Crea un end point su hostname e sulla porta port*
- *Si mette in ascolto di richieste http, rispondendo con una pagina Web*

Il Primo Server: Hello World (server.js)

Prendetevi un po' di tempo per analizzare il codice ed eseguirlo passo passo

Aprite il terminale, andate nella cartella che contiene il vostro server e digitate: ***node server.js***

Ora aprite il browser e andate su:

localhost:3000

```
const http = require('http');

const hostname = '127.0.0.1';
const port = 3000;

const server = http.createServer((req, res) => {
  res.statusCode = 200;
  res.setHeader('Content-Type', 'text/plain');
  res.end('Hello World');
});

server.listen(port, hostname, () => {
  console.log(`Server running at http://${hostname}:${port}/`);
});
```

Esercizio 1: Conta Parole

Ora a voi…

Partite dal codice JavaScript che vi è stato dato e tramite il **modulo fs** leggete un piccolo file di testo dato. Per adesso:

➔leggere l'intero file caricandolo tutto in memoria

➔contare numero di parole

- *Si faccia utilizzo del modulo fs*
- *Le parole sono delimitate da uno o più blank space*
- *Fare attenzione ai new line* ☺
- *Restituire una pagina Web con un input text readonly e una label per la presentazione del risultato*

  ➔*Suggerimento: attenzione al Content-Type*

Esercizio 1.5: Conta Righe, Parole, e Parole su Righe

Partire dall'esercizio precedente e leggere sempre lo stesso file fornito nel kit, stavolta come **stream**, e contare il **numero totale di righe, il numero totale di parole, le parole per ogni riga, e determinare la riga con il maggior numero di parole**

- *Si faccia utilizzo del modulo **readline**. Questo modulo, permette di creare un'interfaccia ed espone un evento «on('line', function)»*
- *L'interfaccia ha questa signature (per maggiori dettagli vedere la documentazione del modulo)*

```
var rl = readline.createInterface({
  input: fs.createReadStream(file),
  output: process.stdout,
  terminal: false
});
```

- *Restituire una pagina Web che presenti il risultato*

Esercizio 2: multi file

Partire dall'esercizio precedente e configurarlo per la lettura di un file a scelta (***secondo quanto specificato nell'URL***)

- *Si faccia utilizzo del **modulo url***
- *Selezionare il file di testo attraverso il nome*
- *Restituire una pagina che presenti il risultato di tutti i conteggi e il nome del file scelto*

Esercizio 3: Elimina parola parametrico

Partire dall'esercizio precedente e configurarlo perché accetti **una parola come parametro di GET**

- *Cercare nel file selezionato la parola passata per parametro*
- *Contarne le occorrenze. Se il numero di ripetizioni supera 5, riscrivere il file eliminando tutte le occorrenze di tale parola*
- *Restituire una pagina Web che presenti il risultato del conteggio e il numero di parole eliminate*