

Laboratorium 1

Zapoznanie się z systemem, serwer Node.js, framework Express.js

Część pierwsza – środowisko pracy i serwer Node.js

Zadanie 1 – zapoznanie się z systemem

Będziemy używać kilku narzędzi, proszę sprawdzić czy są zainstalowane, jakie mają wersje:

- **node** – przetwarzanie języka JavaScript po stronie serwera, pozwoli na tworzenie fajnych aplikacji internetowych,
- **npm** – menadżer pakietów dla Node.js, ważne, żeby był w miarę aktualny, wszystkie pakiety są dostępne na <https://www.npmjs.com>,
- **nodemon** – automatyczne przeładowywanie serwera po zmianie pliku źródłowego, wygodne
- **express** – express-generator, pozwala na wygenerowanie podstawowego szablonu aplikacji internetowej, będziemy używać, zob. <https://expressjs.com/en/starter/generator.html>
- **WebStorm** – świetne narzędzie do tworzenia aplikacji internetowych, można sobie zainstalować, klucz licencyjny u prowadzącego, <https://www.jetbrains.com/webstorm/>

Zadanie 2 – Prosty skrypt w Node.js

Proszę utworzyć skrypt `hello.js`, który wyświetli w terminalu (polecenie `console.log(...)`):

- sumę dwóch liczb `a` i `b` (proszę tutaj utworzyć funkcję `suma()` i jej użyć),
- długość przeciwprostokątnej trójkąta prostokątnego o przyprostokątnych równych `a` i `b`,
- tabliczkę mnożenia do 100.

Wartości `a` i `b` proszę przekazać jako parametry uruchamiania skryptu, np.:

```
node hello.js 12 8.5
```

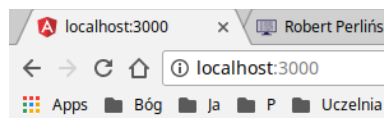
Przyda się funkcja `parseInt()` albo `parseFloat()`. Dostęp do argumentów uruchamiania skryptu `node` poprzez `process.argv[index_parametru]`.

<https://stackoverflow.com/questions/4351521/how-do-i-pass-command-line-arguments-to-a-node-js-program>

Zadanie 3 – serwer w Node.js

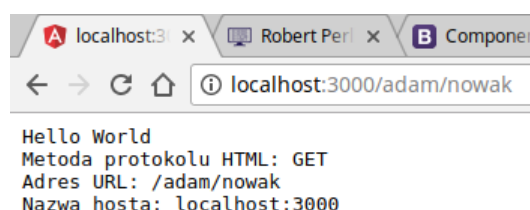
<https://nodejs.org/en/docs/guides/getting-started-guide/>

Proszę napisać prosty serwer wyświetlający napis „Nareszcie nowy rok akademicki!”, uruchamiamy go w Node.js. Można pokusić się o wyświetlenie ładnego nagłówka w HTML, np. H3. Przykładowe działanie:



Nareszcie nowy rok akademicki!

Korzystając z parametru żądania `req` (`request.headers.host`, `request.method`, `request.url`) do powyższej aplikacji dodać wyświetlanie po stronie klienta metody HTTP jaką wykonano żądanie, nazwy hosta i adresu URL żądanego zasobu, np.:



Część druga – framework Express.js

Zadanie 4 – Hello World w Express.js

Tworzymy nowy projekt hello. Wykorzystujemy program npm:

- tworzymy katalog hello
- wewnątrz inicjalizujemy projekt (npm init), tworzy się plik package.json,
- doinstalowujemy szablon aplikacji Express (npm install --save express)

Aplikacja ma wyświetlać „Pierwsza aplikacja w Express.js” dla strony głównej i aktualną godzinę dla adresu '/time'. Obie informacje mają działać tylko dla metody GET. Można przekopiować np. z: <https://expressjs.com/de/starter/hello-world.html> :)

Zadanie 5 – obsługa kilku metod w Express.js

Do aplikacji z zadania 4 dodajemy obsługę trzech innych metod protokołu HTTP, np. POST, PUT i DELETE. Dla każdej metody funkcja ma wyświetlać użytkownikowi inny napis, np. „Żądanie dla metody PUT”.

Do przetestowania proszę wykorzystać dodatek do obsługi REST, np. Postman w Chrome czy Chromium (powinny być zainstalowane). Testujemy działanie utworzonych funkcji korzystając z zainstalowanego dodatku.