



Czym jest JSON?

JSON – format wymiany danych

Wyobraź sobie, że jedziesz do jednej z chińskich prowincji. Nikt nie zna tam ani angielskiego, ani – tym bardziej – polskiego. Chcesz się porozumieć, ale nie zawsze dogadywanie się na migi pomaga. Potrzebujesz kogoś, kto będzie rozumiał język chiński i angielski.

Podobnie sytuacja wygląda, kiedy dochodzi do komunikacji dwóch (lub więcej) aplikacji napisanych w różnych językach programowania. Potrzebują jakiegoś formatu, który będzie przez nie rozumiany.

JSON – JavaScript Object Notation

JSON – to specjalny format tekstowy, dzięki któremu aplikacje mogą w łatwy i przyjemny sposób wymieniać się danymi. Spójrz na przykład obok.

Ten format jest czytelny nie tylko dla człowieka, lecz także dla komputera.

JSON – jak już wiesz – jest niezależny od języka programowania, tzn. możemy go wykorzystywać nie tylko w JavaScriptcie, lecz także w C, C++, Javie, C#, Perlu, Pythonie i wielu innych.

```
{  
  "id": 1,  
  "name": "T-shirt",  
  "price": 16.80,  
  "tags": ["clothes", "tshirts"]  
}
```

JSON – do czego go używamy?

JSON możemy wykorzystywać do przesyłania danych na serwer, odbierania danych z serwera.

Wiele narzędzi używa go również do konfiguracji, np. **npm** do konfiguracji paczek (wkrótce o tym powiemy). Dzięki temu w łatwy sposób możemy powiedzieć menadżerowi paczek jaką konfigurację ma nasz projekt.

Nie tylko **npm** korzysta w ten sposób z JSON-a, również **Webpack**, który poznamy na zajęciach.

Przykład pliku package.json

```
{  
  "name" : "mój projekt",  
  "version" : "1.2.3",  
  "description" : "Moja Aplikacja :D",  
  "tags": ["clothes", "tshirts"]  
}
```

JSON – struktura

JSON powstał w oparciu o dwie znane nam już struktury:

- **Obiekt** (ale w innych językach programowania może mieć inną nazwę np. rekord, struktura, słownik itp.) – czyli zbiór par **klucz i wartość**,
- **Tablica** (również w innych językach może mieć inną nazwę np. wektor, lista) – czyli uporządkowana lista wartości.

Klucz jest zawsze stringiem! Wartość może być: stringiem, liczbą, obiektem, tablicą, wartością boolean (true, false) lub null.

Obiekt

```
{  
  "name" : "Puszek",  
  "age"  : 2  
}
```

Tablica

```
[  
  {  
    "name" : "Puszek",  
    "age"  : 2  
  },  
  {  
    "name" : "Pusheen",  
    "age"  : 1  
  }  
]
```

JSON – jak go sprawdzić?

Jeśli sami tworzymy JSON, to warto sprawdzić jego poprawność, zanim zaczniemy się nim posługiwać. Możemy do tego celu wykorzystać JSON validator:

- <http://jsonlint.com>
- <http://jsonformatter.curiousconcept.com>

Możemy również skorzystać z narzędzi online do generowania losowych danych np.:

- <http://www.json-generator.com>
- <http://www.mockaroo.com>