

JSON

Nikola Milikić

Email: nikola.milikic@fon.bg.ac.rs

URL: <http://nikola.milikic.info>

JSON – JavaScript Object Notation

- Laki (lightweight) format za razmenu podataka
- Jednostavan
 - Za ljude koji ga pišu
 - Za mašine koje ga procesiraju
- JSON je tekstualni format
- Nezavisan od programskog jezika

Tipovi podataka

- Number
- String – sekvenca karaktera
- Boolean – *true* ili *false*
- Objekat – neuređen asocijativni niz (ključ/vrednost)
- Niz – uređena sekvenca nula ili više vrednosti
- null – “prazna vrednost”

JSON objekat

- Predstavlja neuređenu kolekciju parova naziv/vrednost
- JSON objekat počinje sa otvorenom zagradom ({), a završava sa zatvorenom zagradom (})
- Naziv i vrednost su razdvojeni dvotačkom (:), a parovi naziv/vrednost su razdvojeni zapetom (,)

Primer JSON objekta

{

“title” : “The Matrix”,

“producer” : “Joel Silver”,

“release_year” : 1999

}

JSON niz

- JSON niz predstavlja uređenu sekvencu JSON objekata
- Počinje simbolom [, a završava simbolom]
- Objekti su razdvojeni zapetom

Primer JSON niza

```
[  
  {  
    "title" : "The Matrix",  
    "producer" : "Joel Silver",  
    "release_year" : 1999  
  },  
  {  
    "title" : "Equilibrium",  
    "producers" : [  
      {  
        "name" : "Joel Silver"  
      },  
      {  
        "name" : "Lucas Foster"  
      }  
    ],  
    "release_year" : 1999  
  }  
]
```

Generisanje JSON-a i parsiranje

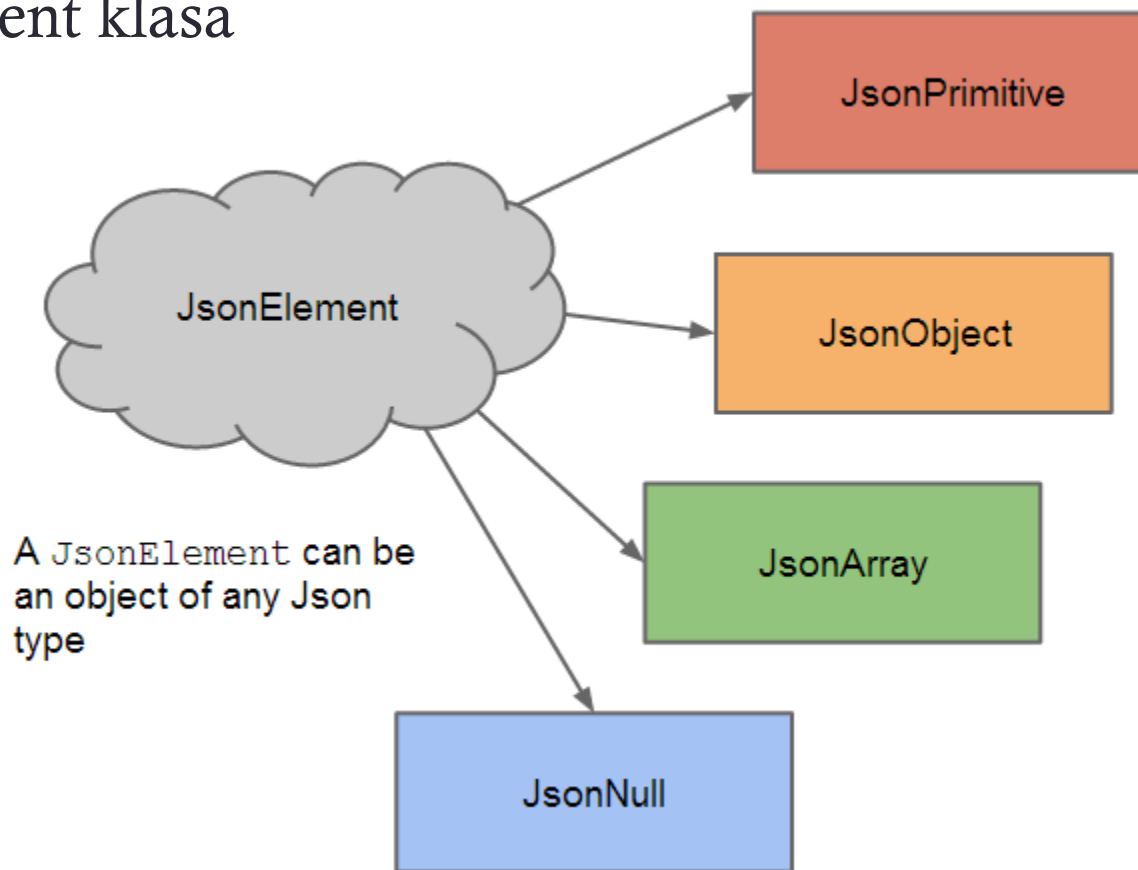
- Biblioteke u Javi za manipulisanje JSON-om:
 - [org.json](#)
 - [google-gson](#)

GSON

- Java biblioteka kreirana od strane Google-a
- Služi za konverziju iz Java objekata u JSON i obratno
- Link: <https://code.google.com/p/google-gson/>

GSON

JsonElement klasa



Ručno kreiranje JSON objekta

```
JsonObject employeeJson = new JsonObject();  
employeeJson.addProperty("name", "John");  
employeeJson.addProperty("last", "Smith");
```

Serijalizacija u JSON

```
Gson gson = new GsonBuilder().create();
```

```
FileWriter writer = new FileWriter("employee.json");
```

```
writer.write(gson.toJson(employeeJson));
```

```
writer.close();
```

employee.json

```
{  
  name : "John",  
  surname: "Smith"  
}
```

Imamo POJO klasu

```
public class Employee {  
    String name;  
    String surname;  
  
    (getter)  
    (setter)  
    ....  
}
```

Serijalizacija u JSON

```
Gson gson = new GsonBuilder().create();
```

```
Employee emp = ...
```

```
FileWriter writer = new FileWriter("employee.json");
```

```
writer.write(gson.toJson(emp));
```

```
writer.close();
```

Ucitavanje iz JSON fajla

```
FileReader reader = new FileReader("employee.json");
```

```
Gson gson = new GsonBuilder().create();
```

```
Employee emp = gson.fromJson(reader, Employee.class);
```


POJO klasa sa imenima polja koja ne odgovaraju atributima JSON fajla

```
public class Employee {  
    @SerializedName("name")  
    String firstName;  
  
    @SerializedName("surname")  
    String lastName;  
  
    (getter)  
    (setter)  
    ....  
}
```

Učitavanje niza JSON objekata

```
Gson gson = new GsonBuilder().create();
```

```
Type type = new TypeToken<List<Employee>>().getType();
```

```
List<Employee> employees = gson.fromJson(json, type);
```

PITANJA?

Nikola Milikić

Email: nikola.milikic@fon.bg.ac.rs

URL: <http://nikola.milikic.info>