

JSON

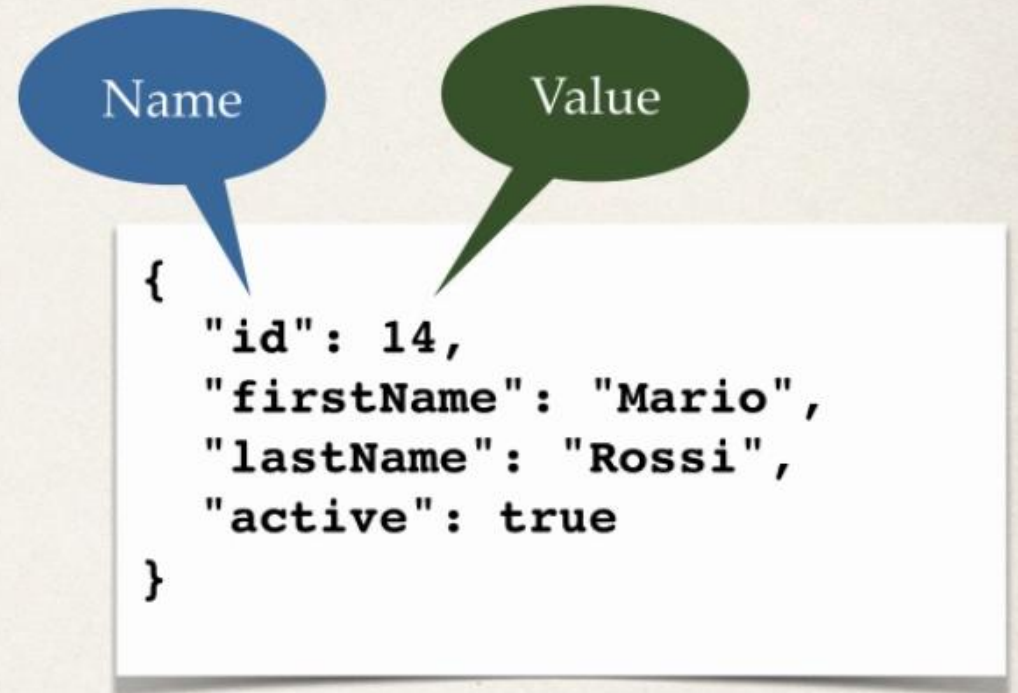
What is JSON?

- JavaScript Object Notation
- Lightweight data format for storing and exchanging data ... plain text
- Language independent ... not just for JavaScript
- Can use with any programming language: Java, C#, Python etc ...

**JSON is just
plain text
data**


Simple JSON Example

- Curley braces define objects in JSON
- Object members are name / value pairs
 - Delimited by colons
- Name is **always** in double-quotes



JSON Values

- Numbers: no quotes
- String: in double quotes
- Boolean: **true**, **false**
- Nested JSON object
- Array
- **null**



```
{  
  "id": 14,  
  "firstName": "Mario",  
  "lastName": "Rossi",  
  "active": true,  
  "courses" : null  
}
```


Nested JSON Objects


```
{  
  "id": 14,  
  "firstName": "Mario",  
  "lastName": "Rossi",  
  "active": true,  
  "address" : {  
    "street" : "100 Main St",  
    "city" : "Philadelphia",  
    "state" : "Pennsylvania",  
    "zip" : "19103",  
    "country" : "USA"  
  }  
}
```



Nested

JSON Arrays

```
{  
  "id": 14,  
  "firstName": "Mario",  
  "lastName": "Rossi",  
  "active": true,  
  "languages" : ["Java", "C#", "Python", "Javascript"]  
}
```

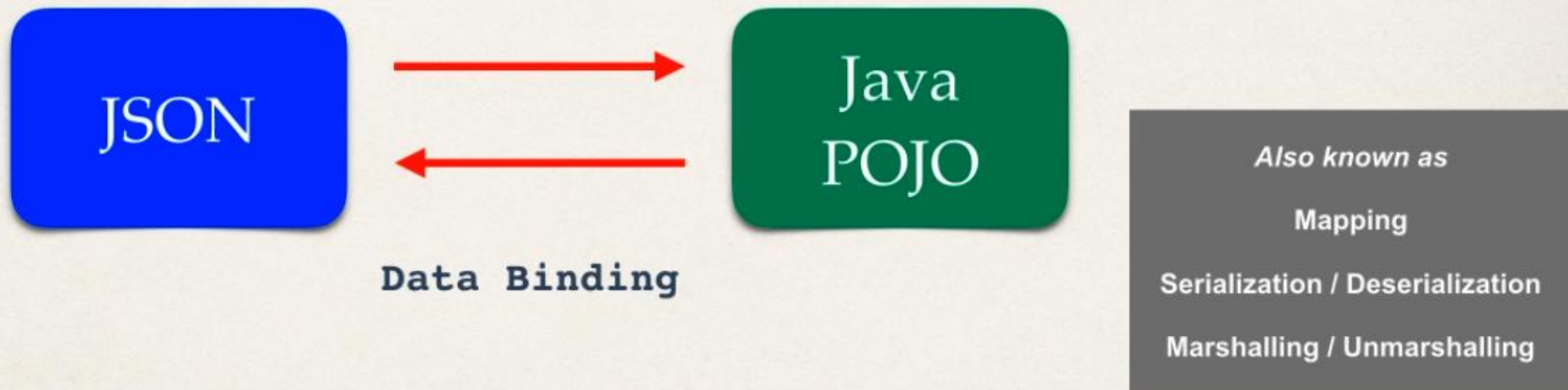


Arrays use square brackets [...]

JSON – Data Binding

Java JSON Data Binding

- Data binding is the process of converting JSON data to a Java POJO



JSON Data Binding with Jackson

- Spring uses the **Jackson Project** behind the scenes
- Jackson handles data binding between JSON and Java POJO
- Details on Jackson Project:

<https://github.com/FasterXML/jackson-databind>

Jackson Data Binding

- Jackson Data Binding API
 - Package: **com.fasterxml.jackson.databind**
- Maven Dependency

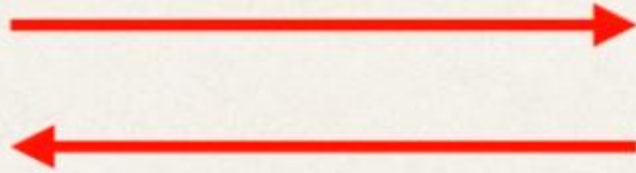
Jackson supports
XML and JSON

```
<dependency>  
  <groupId>com.fasterxml.jackson.core</groupId>  
  <artifactId>jackson-databind</artifactId>  
  <version>2.9.0</version>  
</dependency>
```

Jackson Data Binding

- By default, Jackson will call appropriate getter/setter method

```
{  
  "id": 14,  
  "firstName": "Mario",  
  "lastName": "Rossi",  
  "active": true  
}
```



Java
POJO

Student

JSON to Java POJO

- Convert JSON to Java POJO ... call setter methods on POJO

```
{  
  "id": 14,  
  "firstName": "Mario",  
  "lastName": "Rossi",  
  "active": true  
}
```

*Call
setXXX
methods*



Jackson will do
this work

Java
POJO

Student

JSON to Java POJO

Note: Jackson calls the setXXX methods
It does NOT access internal private fields directly

- Convert JSON to Java POJO ... call setter methods on POJO

```
{  
  "id": 14,  
  "firstName": "Mario",  
  "lastName": "Rossi",  
  "active": true  
}
```

Call setId(...)

Call setFirstName(...)

Call setLastName(...)

Call setActive(...)

```
public class Student {  
  
    private int id;  
    private String firstName;  
    private String lastName;  
    private boolean active;  
  
    public void setId(int id) {  
        this.id = id;  
    }  
  
    public void setFirstName(String firstName) {  
        this.firstName = firstName;  
    }  
  
    public void setLastName(String lastName) {  
        this.lastName = lastName;  
    }  
  
    public void setActive(boolean active) {  
        this.active = active;  
    }  
  
    // getter methods  
}
```

Thank you