JSON

(JavaScript Object Notation)

& JSON Path

What is JSON?

- JSON – **Java Script Object Notation**
- JSON is a syntax for storing and exchanging data.
- Basically It was designed for human-readable data interchange.
- JSON is text, written with Java Script Object Notation.
- It has been extended from the JavaScript scripting language
- The filename extension is **.json**
- JSON internet Media type is **application/json**

JSON Data Types

- Number
- String
- Boolean
- Null
- Object
- Array

Data Types

- **String**
- Strings in JSON must be written in double quotes.
- Example:

```
{ "name":"John" }
```

- **Numbers**
- Numbers in JSON must be an integer or a floating point.
- Example:

```
{ "age":30 }
```

- **Object**
- Values in JSON can be objects.
- Example:

```
{
"employee":{ "name":"John", "age":30, "city":"New York" }
}
```

Data Types

- **Array**
- Values in JSON can be arrays.
- Example:

```
{
"employees":[ "John", "Anna", "Peter" ]
}
```

- **Boolean**
- Values in JSON can be true/false.
- Example:

```
{ "sale":true }
```

- **Null**
- Values in JSON can be null.

```
{ "middlename":null }
```

JSON – Syntax

- Data should be in name/value pairs
- Data should be separated by commas
- Curly braces should hold objects
- Square brackets hold arrays

```
{
  "student": [

     {
        "id":"01",
        "name": "Tom",
        "lastname": "Price"
     },

     {
        "id":"02",
        "name": "Nick",
        "lastname": "Thameson"
     }
  ]
}
```

JSON vs XML

JSON	XML
JSON is simple to read and write.	XML is less simple as compared to JSON.
It also supports **array**.	It doesn't support array.
JSON files are more **human-readable** than XML.	XML files are **less human readable**.
It supports only **text** and **number** data type	It supports many data types such as **text, number, images, charts, graphs**, etc.

Examples

JSON Example

```
{"employees":[
   {"name":"Vimal", "email":"vjaiswal1987@gmail.com"},
   {"name":"Rahul", "email":"rahul12@gmail.com"},
   {"name":"Jai", "email":"jai87@gmail.com"}
]}
```

XML Example

```
<employees>
   <employee>
      <name>Vimal</name>
      <email>vjaiswal1987@gmail.com</email>
   </employee>
   <employee>
      <name>Rahul</name>
      <email>rahul12@gmail.com</email>
   </employee>
   <employee>
      <name>Jai</name>
      <email>jai87@gmail.com</email>
   </employee>
</employees>
```

JSON Object & JSON Array

JSON Object

- JSON object holds key/value pair. Each key is represented as a string in JSON and value can be of any type.
- The keys and values are separated by colon. Each key/value pair is separated by comma.
- The curly brace { represents JSON object.
- Example:

```
{
   "employee": {
      "name":      "sonoo",
      "salary":    56000,
      "married":   true
   }
}
```

- JSON Object with Strings

```
{
"name":      "Scott",
"email":     "Scottjaiswal1987@gmail.com"

}
```

- JSON Object with Numbers

```
{
"integer": 34,
"fraction": .2145,
"exponent": 6.61789e+0
}
```

- JSON Object with Booleans

```
{
"first": true,
"second": false
}
```

- JSON Nested Object

```
{
    "firstName": "Scott",
    "lastName": "Jaiswal",
    "age": 27,
    "address" : {
        "streetAddress": "Plot-6, Mohan Nagar",
        "city": “Hyderabad",
        "state": “TL",
        "postalCode": “500090"
    }
}
```

JSON Array

- JSON array represents ordered list of values.
- JSON array can store multiple values. It can store string, number, boolean or object in JSON array.
- In JSON array, values must be separated by comma.
- The [(square bracket) represents JSON array.

JSON Array of Strings

["Sunday", "Monday", "Tuesday", "Wednesday", "Thursday", "Friday", "Saturday"]

JSON Array of Numbers

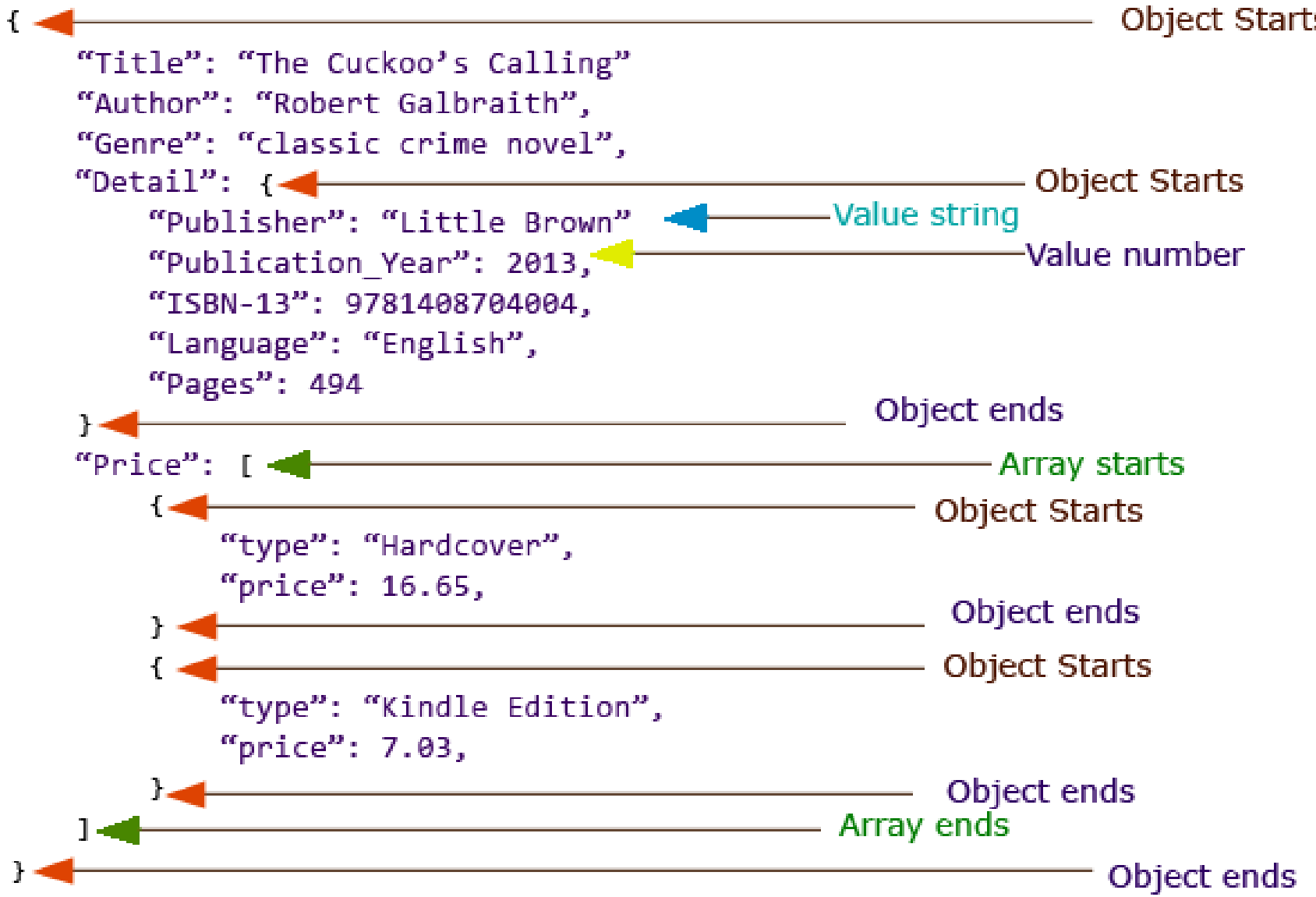
[12, 34, 56, 43, 95]

JSON Array of Booleans

[true, true, false, false, true]

JSON Array of Objects

```
{"employees":[
    {"name":"Ram", "email":"ram@gmail.com", "age":23},
    {"name":"Shyam", "email":"shyam23@gmail.com", "age":28},
    {"name":"John", "email":"john@gmail.com", "age":33},
    {"name":"Bob", "email":"bob32@gmail.com", "age":41}
]}
```


{
Object Starts
"Title": "The Cuckoo's Calling"
"Author": "Robert Galbraith",
"Genre": "classic crime novel",
"Detail": {
Object Starts
"Publisher": "Little Brown"
Value string
"Publication_Year": 2013,
Value number
"ISBN-13": 9781408704004,
"Language": "English",
"Pages": 494
}
Object ends
"Price": [
Array starts
{
Object Starts
"type": "Hardcover",
"price": 16.65,
}
Object ends
{
Object Starts
"type": "Kindle Edition",
"price": 7.03,
}
Object ends
]
Array ends
}
Object ends

Capture & Validate JSON Path

https://jsonpathfinder.com/

https://jsonpath.com/