Date : 13/03/2021
Spring boot 9AM
Mr. RAGHU

POSTMAN Tool:
https://www.postman.com/downloads/

*) ResponseEntity<T> :
Entity(Final Output) Placed inide Response.
This is used as Controller#Method ReturnType
T = Type can be String/Class/Collection

=> If return type is String , then Http Response is 'Text Format'
(Content-Type : text/plain).
=> If return type is non-String , then Http Response is 'JSON'
(Content-Type : application/json).
=> *** Latest Spring Boot version 2.x supports only JSON Format.
XML Supports default is not present, but we can add additional
dependency to enable this.

Q) Which is prefered for Object Format in Webservices JSON or XML?
A) JSON.
-> Representation is easy { key:val }
-> Need less memory (compared to XML)
-> Execution/Trasnfer is very fast.
-> XML is more complex in design if data gets increased

ex:
XML:
```
<employee>
 <eid>10</eid>
 <ename>A</ename>
</employee>
```

JSON:
```
{eid:10,ename:"A"}
```

MEDIA TYPE ANNOTATIONS

ReST has provided two mediatype annotations.
a. @RequestBody (Input MediaType)
b. @ResponseBody (Output MediaType)

a. @RequestBody : It reads Http Request first.
Checks Header Param 'Content-Type'
If Type is non-string(not text/plain)
ex: application/json , application/xml
then data/Content is converted into Object Format,
given to Controller#method Param.

=> Spring MVC @ModelAttribute (FORM DATA) === Spring ReST
@RequestBody

b. @ResponseBody : It will take Controller#method return type first.
If type is non-String Ex: Employee, List<Product>
then data is converted into JSON/XML (based on request type)

and give it to Http Response Body Section and update
Content-Type Header Param even.

=> Spring MVC Model === Spring ReST @ResponseBody
---

*) @ResponseBody Annotation is optional to apply over method.
  Bcoz internally @RestController is appling @ResponseBody Annotation.
  It works only for non-String.

*) JSONView chrome Extesion.
*) We can directly use String/Class/Collection as ReturnType
   ResponseEntity is optional, but recomanded in realtime.

-------code---

Name: SpringBoot2OutputMediaType
Dep : Spring web, Lombok, Devtools

Model:

```
package in.nareshit.raghu.model;

import lombok.AllArgsConstructor;
import lombok.Data;
import lombok.NoArgsConstructor;

@Data
@NoArgsConstructor
@AllArgsConstructor
public class Employee {

	private Integer eid;
	private String ename;
	private Double esal;
}
```

RestController

```
package in.nareshit.raghu.rest;

import org.springframework.http.HttpStatus;
import org.springframework.http.ResponseEntity;
import org.springframework.web.bind.annotation.GetMapping;
import org.springframework.web.bind.annotation.RestController;

import in.nareshit.raghu.model.Employee;

@RestController
public class EmployeeRestController {

	@GetMapping("/msg")
	public ResponseEntity<String> showDataA() {
		String body = "Welcome to Simple data";
		return new ResponseEntity<String>(body,
HttpStatus.OK);
	}

	@GetMapping("/obj")
	public ResponseEntity<Employee> showDataB() {
		Employee body = new Employee(106, "SYED", 600.0);
```

```
                return new ResponseEntity<Employee>(body,
HttpStatus.OK);
        }

        @GetMapping("/obja")
        public Employee showDataC() {
                Employee body = new Employee(190, "AHMED SYED",
800.0);
                return body;
        }
}
```

URL:
http://localhost:8080/obj
http://localhost:8080/obja
