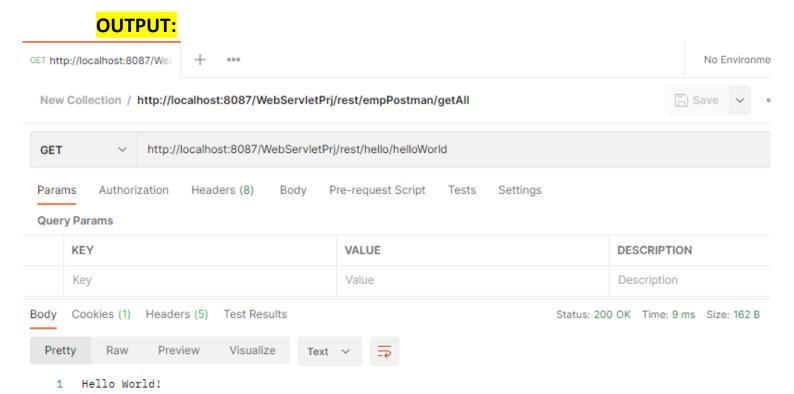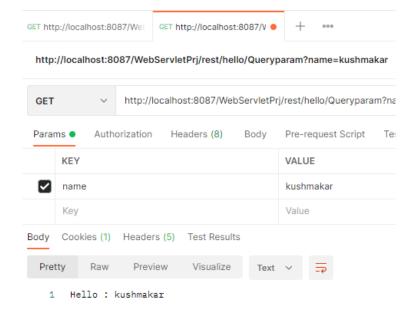**RESTFul ASSIGNMENT Day_1**

Harshit Kushmakar| 16896

1. **Create a Restful WebService to send "Hello World" back to the user. Use Postman API to test your Service.**

```java
package com.rest;
import javax.ws.rs.GET;
import javax.ws.rs.Path;
import javax.ws.rs.PathParam;
import javax.ws.rs.Produces;
import javax.ws.rs.core.MediaType;
import javax.ws.rs.core.Response;
@Path("/hello")
public class HelloWorldService{

    @GET
    @Path("/helloWorld")
    @Produces(MediaType.TEXT_PLAIN)
    public String getMsg() {
        return "Hello World!";
    }

}
```

OUTPUT:

GET http://localhost:8087/Wel + ooo   No Environme

New Collection / http://localhost:8087/WebServletPrj/rest/empPostman/getAll   💾 Save ∨

GET	∨	http://localhost:8087/WebServletPrj/rest/hello/helloWorld

Params Authorization Headers (8) Body Pre-request Script Tests Settings

Query Params

KEY	VALUE	DESCRIPTION
Key	Value	Description

Body Cookies (1) Headers (5) Test Results   Status: 200 OK Time: 9 ms Size: 162 B

Pretty Raw Preview Visualize Text ∨ ⇄

```
1    Hello World!
```

2. In the same question add a Query String as mentioned below: [">](#) [Web Service Demo](#) **Retrieve this parameter with the help of @QueryParam annotation and send a message "Hello " + username back to client application. Use Postman API to test your service ➔ In both the above questions, Rest Service produces a String response.**

```java
package com.rest;
import javax.ws.rs.*;
import javax.ws.rs.core.MediaType;
import javax.ws.rs.core.Response;


@Path("/hello")
public class HelloWorldService {



    @GET
    @Path("/Queryparam")
    @Produces(MediaType.TEXT_PLAIN)
    public Response getMsgNAme(@QueryParam("name") String msg) {
        String output = "Hello : " + msg;
        return Response.status(200).entity(output).build();
    }


}
```

GET http://localhost:8087/We   GET http://localhost:8087/V ●   +   ○○○

http://localhost:8087/WebServletPrj/rest/hello/Queryparam?name=kushmakar

| GET | ⌄ | http://localhost:8087/WebServletPrj/rest/hello/Queryparam?na |

Params ●   Authorization   Headers (8)   Body   Pre-request Script   Te:

	KEY	VALUE
☑	name	kushmakar
	Key	Value

Body   Cookies (1)   Headers (5)   Test Results

| Pretty | Raw | Preview | Visualize | Text ⌄ | ⇄ |

```
1   Hello : kushmakar
```

3. Change the above question to use @PathParam annotation.
[WebService Demo](#).

```java
package com.rest;
import javax.ws.rs.*;
import javax.ws.rs.core.MediaType;
import javax.ws.rs.core.Response;


@Path("/hello")
public class HelloWorldService {



@GET
@Path("/{param}")
@Produces(MediaType.TEXT_PLAIN)
public Response getMsg(@PathParam("param") String msg) {
    String output = "Hello : " + msg;
    return Response.status(200).entity(output).build();
    }
}
```

GET http://localhost:8087/Wel   GET http://localhost:8087/Wel   GET http://localhost:8087/V ●   +   ◦◦

http://localhost:8087/WebServletPrj/rest/hello/harshit

| GET | ∨ | http://localhost:8087/WebServletPrj/rest/hello/harshit |

Params   Authorization   Headers (8)   Body   Pre-request Script   Tests   Settings

Query Params

KEY	VALUE
Key	Value

Body   Cookies (1)   Headers (5)   Test Results   ⊕

Pretty   Raw   Preview   Visualize   Text ∨   ⇄

```
1   Hello World : harshit
```