1.Spring Boot Exception Handling.

productController.java

```
package com.example.demo;

import org.springframework.stereotype.Controller;
import org.springframework.ui.Model;
import org.springframework.web.bind.annotation.GetMapping;
import org.springframework.web.bind.annotation.PathVariable;

@Controller
public class ProductController {

        @GetMapping("/product/{id}")
        public String getProduct(@PathVariable("id") int id, Model model) {

                // Business logic
                if (id == 0)
                        throw new ProductNotFoundException();
                else {
                        model.addAttribute("id", id);
                }

                return "product-found"; // go to product-found.jsp

        }

        // First approach: we can write ExceptionHandler in the controller
        // where the exception happens
//      @ExceptionHandler(value = ProductNotFoundException.class)
//      public ResponseEntity<Object> abc(ProductNotFoundException exception) {
//              return new ResponseEntity<>("Product not found. Make sure your id > 0.",
HttpStatus.NOT_FOUND);
//      }


}
```

ProductExceptionHandler.java

```
package com.example.demo;
import org.springframework.http.HttpStatus;
import org.springframework.http.ResponseEntity;
import org.springframework.web.bind.annotation.ControllerAdvice;
import org.springframework.web.bind.annotation.ExceptionHandler;

@ControllerAdvice
public class ProductExceptionHandler {

	// Second approach: we can write ExceptionHandler in a separate class like this
	// one.
	@ExceptionHandler(value = ProductNotFoundException.class)
	public ResponseEntity<Object> abc(ProductNotFoundException exception) {
		return new ResponseEntity<>("Product not found! Make sure your id > 0.",
HttpStatus.NOT_FOUND);
	}

}
```

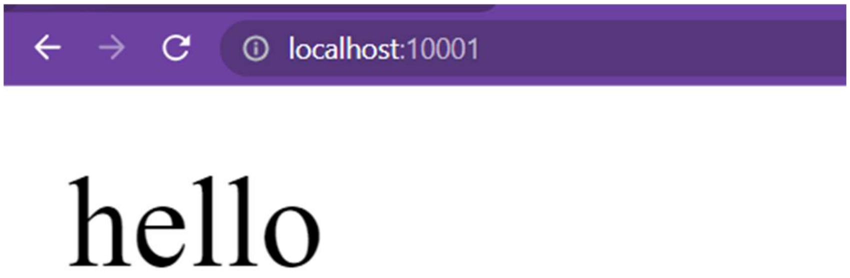
ProductNotFoundException.java

```
package com.example.demo;


public class ProductNotFoundException extends RuntimeException {

}
```

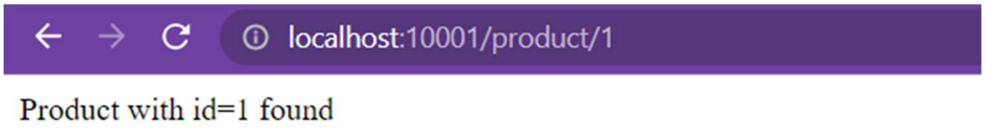
ProductNotFound.jsp

```
Product with id=${id} found
```

Output



```
hello
```



```
Product with id=1 found
```

2.Create a project to Consume a RESTful web service

Quote.java

```
package com.ecommerce.beans;
import com.fasterxml.jackson.annotation.JsonIgnoreProperties;
@JsonIgnoreProperties(ignoreUnknown = true)
public class Quote {

   private String type;
   private Value value;

   public Quote() {
   }

   public String getType() {
      return type;
   }

   public void setType(String type) {
      this.type = type;
   }

   public Value getValue() {
      return value;
   }

   public void setValue(Value value) {
      this.value = value;
   }

   @Override
   public String toString() {
      return "Quote{" +
            "type='" + type + '\'' +
            ", value=" + value +
            '}';
   }
}
```

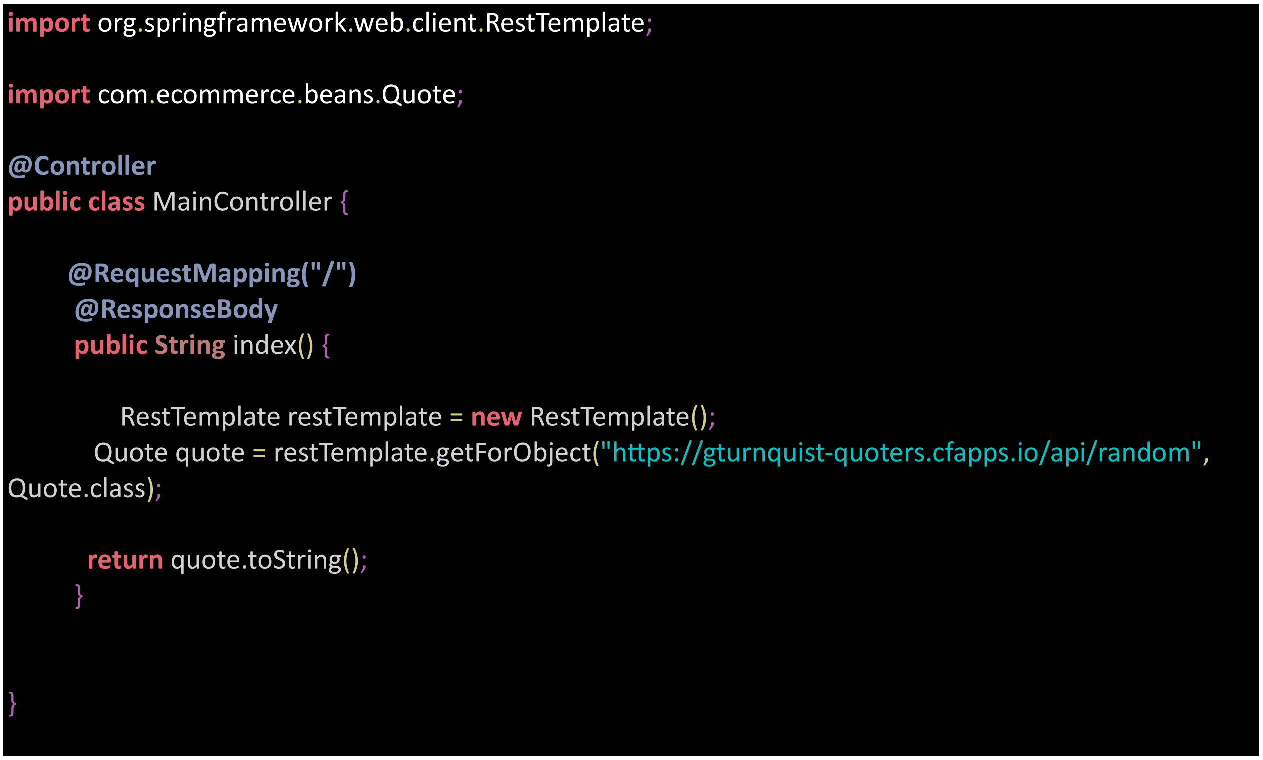
Value.java

```
package com.ecommerce.beans;
```

```
import com.fasterxml.jackson.annotation.JsonIgnoreProperties;

@JsonIgnoreProperties(ignoreUnknown = true)
public class Value {

    private Long id;
    private String quote;

    public Value() {
    }

    public Long getId() {
        return this.id;
    }

    public String getQuote() {
        return this.quote;
    }

    public void setId(Long id) {
        this.id = id;
    }

    public void setQuote(String quote) {
        this.quote = quote;
    }

    @Override
    public String toString() {
        return "Value{" +
                "id=" + id +
                ", quote='" + quote + '\'' +
                '}';
    }
}
```

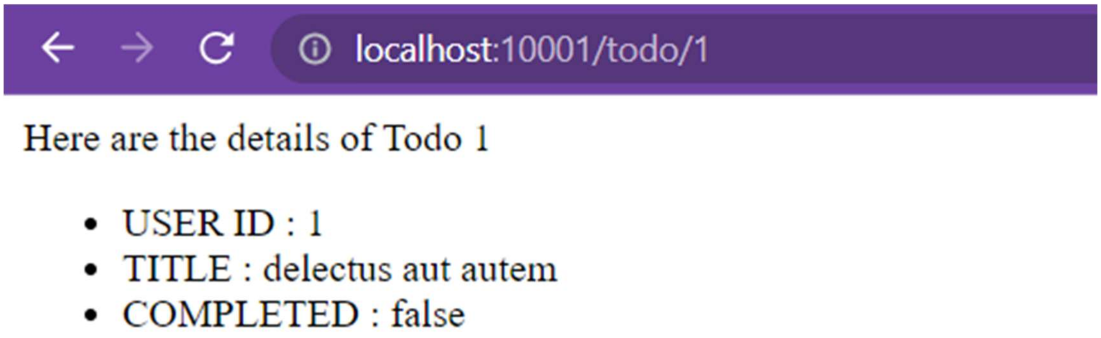
MainController.java

```
package com.ecommerce.controllers;
import org.springframework.http.HttpStatus;
import org.springframework.http.ResponseEntity;
import org.springframework.stereotype.Controller;
import org.springframework.web.bind.annotation.PathVariable;
import org.springframework.web.bind.annotation.RequestBody;
import org.springframework.web.bind.annotation.RequestMapping;
import org.springframework.web.bind.annotation.RequestMethod;
import org.springframework.web.bind.annotation.ResponseBody;
```



```
import org.springframework.web.client.RestTemplate;

import com.ecommerce.beans.Quote;

@Controller
public class MainController {

        @RequestMapping("/")
         @ResponseBody
         public String index() {

                RestTemplate restTemplate = new RestTemplate();
             Quote quote = restTemplate.getForObject("https://gturnquist-quoters.cfapps.io/api/random",
Quote.class);

           return quote.toString();
         }



}
```

Output



Here are the details of Todo 1

- USER ID : 1
- TITLE : delectus aut autem
- COMPLETED : false

3.Create a project to upload and download a file in Spring Boot.

index.html

```
<html>
<head><title>File Upload</title></head>
<body>

     <form  method="post" enctype="multipart/form-data" action="/upload">
      Upload file&nbsp;
       <input type="file" name="fileToUpload" id="fileToUpload"><br><br>
   <input type="submit" value="Upload " name="submit">
     </form>
</body>
</html>
```

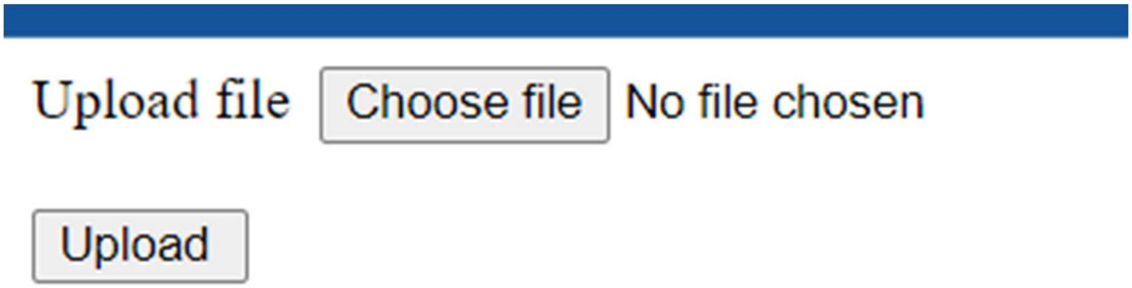
}

MainController.java

```
package com.ecommerce.controllers;

import java.io.File;
import java.io.FileInputStream;
import java.io.FileOutputStream;
import java.io.IOException;

import org.springframework.core.io.ClassPathResource;
import org.springframework.core.io.InputStreamResource;
import org.springframework.core.io.Resource;
import org.springframework.http.HttpHeaders;
import org.springframework.http.HttpStatus;
import org.springframework.http.MediaType;
import org.springframework.http.ResponseEntity;
import org.springframework.stereotype.Controller;
import org.springframework.util.ResourceUtils;
import org.springframework.web.bind.annotation.PathVariable;
import org.springframework.web.bind.annotation.RequestBody;
import org.springframework.web.bind.annotation.RequestMapping;
import org.springframework.web.bind.annotation.RequestMethod;
import org.springframework.web.bind.annotation.RequestParam;
import org.springframework.web.bind.annotation.ResponseBody;
import org.springframework.web.multipart.MultipartFile;
```

```
@Controller
public class MainController {

      @RequestMapping(value = "/")
   public String index() {
      return "index.html";
   }

      @RequestMapping(value = "/upload", method = RequestMethod.POST, consumes =
MediaType.MULTIPART_FORM_DATA_VALUE)
      public String fileUpload(@RequestParam("file") MultipartFile file) {
            String result = "File was uploaded successfully";

            try {
             File convertFile = new File("/var/tmp/"+file.getOriginalFilename());
          convertFile.createNewFile();
          FileOutputStream fout = new FileOutputStream(convertFile);
          fout.write(file.getBytes());
          fout.close();

            } catch (IOException iex) {
                  result = "Error " + iex.getMessage();

            } finally {
                  return result;
            }
      }

      @RequestMapping(value = "/download", method = RequestMethod.GET)
      public ResponseEntity<Object> downloadFile() throws IOException  {
            String fileName = "static/dump.txt";
      ClassLoader classLoader = new MainController().getClass().getClassLoader();

      File file = new File(classLoader.getResource(fileName).getFile());

        InputStreamResource resource = new InputStreamResource(new FileInputStream(file));
        HttpHeaders headers = new HttpHeaders();

        headers.add("Content-Disposition", String.format("attachment; filename=\"%s\"",
file.getName()));
        headers.add("Cache-Control", "no-cache, no-store, must-revalidate");
        headers.add("Pragma", "no-cache");
        headers.add("Expires", "0");

        ResponseEntity<Object>
        responseEntity = ResponseEntity.ok().headers(headers).contentLength(file.length()).contentType(
          MediaType.parseMediaType("application/txt")).body(resource);
        return responseEntity;
    }
```

Output



4. Create a project to enable HTTPS and display in browser.

Maincontroller.java

```
package com.ecommerce.controllers;

import org.springframework.beans.factory.annotation.Autowired;
import org.springframework.http.HttpStatus;
import org.springframework.http.ResponseEntity;
import org.springframework.stereotype.Controller;
import org.springframework.web.bind.annotation.PathVariable;
import org.springframework.web.bind.annotation.RequestBody;
import org.springframework.web.bind.annotation.RequestMapping;
import org.springframework.web.bind.annotation.RequestMethod;
import org.springframework.web.bind.annotation.ResponseBody;

@Controller
public class MainController {

        @Autowired
        private ProductRepository repository;

        @RequestMapping("/")
         @ResponseBody
         public String index() {
           return "This is running under SSL";
         }
}
```

Application.properties

```
server.port=8443

server.ssl.key-alias=selfsigned_localhost_sslserver

server.ssl.key-password=changeit

server.ssl.key-store=classpath:ssl-server.jks

server.ssl.key-store-provider=SUN

server.ssl.key-store-type=JKS
```