```
Yesterday's session : REST Controller Unit Testing
-------------------------------------------------

-> We can perform Unit testing for rest controller using MockMvc.


MockMvcRequestBuilders reqBuilder =
                        MockMvcRequestBuilders.get("");

MvcResult result = mockMvc.perform(reqBuilder).andReturn();

-> We can validate rest api method results using response body conte

        int status = result.getResponse().getStatus();

-----------------------------------------------------------------

What is Mocking?
----------------
-> It is the process of creating substitute/alternate object for the

-----------------------------------------------------------------
package com.ashokit.service;

import org.springframework.stereotype.Service;

import com.ashokit.bindings.Book;

@Service
public class BookService {

        public boolean saveBook(Book book) {
                return true;
        }

}
-----------------------------------------------------------------
package com.ashokit.rest;

import org.springframework.beans.factory.annotation.Autowired;
import org.springframework.http.HttpStatus;
import org.springframework.http.ResponseEntity;
import org.springf                    nd.annotation.PostMapping;
import org.springf                    nd.annotation.RequestBody;
import org.springf                    nd.annotation.RestController;
```

```
Yesterday's session : REST Controller Unit Testing
-------------------------------------------------

-> We can perform Unit testing for rest controller using MockMvc.


MockMvcRequestBuilders reqBuilder =
                        MockMvcRequestBuilders.get("");

MvcResult result = mockMvc.perform(reqBuilder).andReturn();

-> We can validate rest api method results using response body conte

        int status = result.getResponse().getStatus();

-----------------------------------------------------------------

What is Mocking?
----------------
-> It is the process of creating substitute/alternate object for the

-----------------------------------------------------------------
package com.ashokit.service;

import org.springframework.stereotype.Service;

import com.ashokit.bindings.Book;

@Service
public class BookService {

        public boolean saveBook(Book book) {
                return true;
        }

}
-----------------------------------------------------------------
package com.ashokit.rest;

import org.springframework.beans.factory.annotation.Autowired;
import org.springframework.http.HttpStatus;
import org.springframework.http.ResponseEntity;
import org.springf                .nd.annotation.PostMapping;
import org.springf                .nd.annotation.RequestBody;
import org.springf                .nd.annotation.RestController;
```

```
Yesterday's session : REST Controller Unit Testing
-------------------------------------------------

-> We can perform Unit testing for rest controller using MockMvc.


MockMvcRequestBuilders reqBuilder =
                        MockMvcRequestBuilders.get("");

MvcResult result = mockMvc.perform(reqBuilder).andReturn();

-> We can validate rest api method results using response body conte

        int status = result.getResponse().getStatus();

-----------------------------------------------------------------

What is Mocking?
----------------
-> It is the process of creating substitute/alternate object for the

-----------------------------------------------------------------
package com.ashokit.service;

import org.springframework.stereotype.Service;

import com.ashokit.bindings.Book;

@Service
public class BookService {

        public boolean saveBook(Book book) {
                return true;
        }

}
-----------------------------------------------------------------
package com.ashokit.rest;

import org.springframework.beans.factory.annotation.Autowired;
import org.springframework.http.HttpStatus;
import org.springframework.http.ResponseEntity;
import org.springf                    nd.annotation.PostMapping;
import org.springf                    nd.annotation.RequestBody;
import org.springf                    nd.annotation.RestController;
```

```
Yesterday's session : REST Controller Unit Testing
-------------------------------------------------

-> We can perform Unit testing for rest controller using MockMvc.


MockMvcRequestBuilders reqBuilder =
                        MockMvcRequestBuilders.get("");

MvcResult result = mockMvc.perform(reqBuilder).andReturn();

-> We can validate rest api method results using response body conte

        int status = result.getResponse().getStatus();

-----------------------------------------------------------------

What is Mocking?
----------------
-> It is the process of creating substitute/alternate object for the

-----------------------------------------------------------------
package com.ashokit.service;

import org.springframework.stereotype.Service;

import com.ashokit.bindings.Book;

@Service
public class BookService {

        public boolean saveBook(Book book) {
                return true;
        }

}
-----------------------------------------------------------------
package com.ashokit.rest;

import org.springframework.beans.factory.annotation.Autowired;
import org.springframework.http.HttpStatus;
import org.springframework.http.ResponseEntity;
import org.springf                    nd.annotation.PostMapping;
import org.springf                    nd.annotation.RequestBody;
import org.springf                    nd.annotation.RestController;
```

```
Yesterday's session : REST Controller Unit Testing
-------------------------------------------------

-> We can perform Unit testing for rest controller using MockMvc.


MockMvcRequestBuilders reqBuilder =
                        MockMvcRequestBuilders.get("");

MvcResult result = mockMvc.perform(reqBuilder).andReturn();

-> We can validate rest api method results using response body conte

        int status = result.getResponse().getStatus();

-----------------------------------------------------------------

What is Mocking?
----------------
-> It is the process of creating substitute/alternate object for the

-----------------------------------------------------------------
package com.ashokit.service;

import org.springframework.stereotype.Service;

import com.ashokit.bindings.Book;

@Service
public class BookService {

        public boolean saveBook(Book book) {
                return true;
        }

}
-----------------------------------------------------------------
package com.ashokit.rest;

import org.springframework.beans.factory.annotation.Autowired;
import org.springframework.http.HttpStatus;
import org.springframework.http.ResponseEntity;
import org.springf                    nd.annotation.PostMapping;
import org.springf                    nd.annotation.RequestBody;
import org.springf                    nd.annotation.RestController;
```

```
Yesterday's session : REST Controller Unit Testing
-------------------------------------------------

-> We can perform Unit testing for rest controller using MockMvc.


MockMvcRequestBuilders reqBuilder =
                        MockMvcRequestBuilders.get("");

MvcResult result = mockMvc.perform(reqBuilder).andReturn();

-> We can validate rest api method results using response body conte

        int status = result.getResponse().getStatus();

-----------------------------------------------------------------

What is Mocking?
----------------
-> It is the process of creating substitute/alternate object for the

-----------------------------------------------------------------
package com.ashokit.service;

import org.springframework.stereotype.Service;

import com.ashokit.bindings.Book;

@Service
public class BookService {

        public boolean saveBook(Book book) {
                return true;
        }

}
-----------------------------------------------------------------
package com.ashokit.rest;

import org.springframework.beans.factory.annotation.Autowired;
import org.springframework.http.HttpStatus;
import org.springframework.http.ResponseEntity;
import org.springf                    nd.annotation.PostMapping;
import org.springf                    nd.annotation.RequestBody;
import org.springf                    nd.annotation.RestController;
```

```
Yesterday's session : REST Controller Unit Testing
--------------------------------------------------

-> We can perform Unit testing for rest controller using MockMvc.


MockMvcRequestBuilders reqBuilder =
                        MockMvcRequestBuilders.get("");

MvcResult result = mockMvc.perform(reqBuilder).andReturn();

-> We can validate rest api method results using response body conte

        int status = result.getResponse().getStatus();

-----------------------------------------------------------------

What is Mocking?
----------------
-> It is the process of creating substitute/alternate object for the

-----------------------------------------------------------------
package com.ashokit.service;

import org.springframework.stereotype.Service;

import com.ashokit.bindings.Book;

@Service
public class BookService {

        public boolean saveBook(Book book) {
                return true;
        }

}
-----------------------------------------------------------------
package com.ashokit.rest;

import org.springframework.beans.factory.annotation.Autowired;
import org.springframework.http.HttpStatus;
import org.springframework.http.ResponseEntity;
import org.springf                    nd.annotation.PostMapping;
import org.springf                    nd.annotation.RequestBody;
import org.springf                    nd.annotation.RestController;
```

```
Yesterday's session : REST Controller Unit Testing
-------------------------------------------------

-> We can perform Unit testing for rest controller using MockMvc.


MockMvcRequestBuilders reqBuilder =
                        MockMvcRequestBuilders.get("");

MvcResult result = mockMvc.perform(reqBuilder).andReturn();

-> We can validate rest api method results using response body conte

        int status = result.getResponse().getStatus();

-----------------------------------------------------------------

What is Mocking?
----------------
-> It is the process of creating substitute/alternate object for the

-----------------------------------------------------------------
package com.ashokit.service;

import org.springframework.stereotype.Service;

import com.ashokit.bindings.Book;

@Service
public class BookService {

        public boolean saveBook(Book book) {
                return true;
        }

}
-----------------------------------------------------------------
package com.ashokit.rest;

import org.springframework.beans.factory.annotation.Autowired;
import org.springframework.http.HttpStatus;
import org.springframework.http.ResponseEntity;
import org.springf                    nd.annotation.PostMapping;
import org.springf                    nd.annotation.RequestBody;
import org.springf                    nd.annotation.RestController;
```

```
Yesterday's session : REST Controller Unit Testing
-------------------------------------------------

-> We can perform Unit testing for rest controller using MockMvc.


MockMvcRequestBuilders reqBuilder =
                        MockMvcRequestBuilders.get("");

MvcResult result = mockMvc.perform(reqBuilder).andReturn();

-> We can validate rest api method results using response body conte

        int status = result.getResponse().getStatus();

-----------------------------------------------------------------

What is Mocking?
----------------
-> It is the process of creating substitute/alternate object for the

-----------------------------------------------------------------
package com.ashokit.service;

import org.springframework.stereotype.Service;

import com.ashokit.bindings.Book;

@Service
public class BookService {

        public boolean saveBook(Book book) {
                return true;
        }

}
-----------------------------------------------------------------
package com.ashokit.rest;

import org.springframework.beans.factory.annotation.Autowired;
import org.springframework.http.HttpStatus;
import org.springframework.http.ResponseEntity;
import org.springf                    nd.annotation.PostMapping;
import org.springf                    nd.annotation.RequestBody;
import org.springf                    nd.annotation.RestController;
```

```
Yesterday's session : REST Controller Unit Testing
--------------------------------------------------

-> We can perform Unit testing for rest controller using MockMvc.


MockMvcRequestBuilders reqBuilder =
                        MockMvcRequestBuilders.get("");

MvcResult result = mockMvc.perform(reqBuilder).andReturn();

-> We can validate rest api method results using response body conte

        int status = result.getResponse().getStatus();

-----------------------------------------------------------------

What is Mocking?
----------------
-> It is the process of creating substitute/alternate object for the

-----------------------------------------------------------------
package com.ashokit.service;

import org.springframework.stereotype.Service;

import com.ashokit.bindings.Book;

@Service
public class BookService {

        public boolean saveBook(Book book) {
                return true;
        }

}
-----------------------------------------------------------------
package com.ashokit.rest;

import org.springframework.beans.factory.annotation.Autowired;
import org.springframework.http.HttpStatus;
import org.springframework.http.ResponseEntity;
import org.springf                    nd.annotation.PostMapping;
import org.springf                    nd.annotation.RequestBody;
import org.springf                    nd.annotation.RestController;
```

```
Yesterday's session : REST Controller Unit Testing
--------------------------------------------------

-> We can perform Unit testing for rest controller using MockMvc.


MockMvcRequestBuilders reqBuilder =
                        MockMvcRequestBuilders.get("");

MvcResult result = mockMvc.perform(reqBuilder).andReturn();

-> We can validate rest api method results using response body conte

        int status = result.getResponse().getStatus();

------------------------------------------------------------------

What is Mocking?
----------------
-> It is the process of creating substitute/alternate object for the

------------------------------------------------------------
package com.ashokit.service;

import org.springframework.stereotype.Service;

import com.ashokit.bindings.Book;

@Service
public class BookService {

        public boolean saveBook(Book book) {
                return true;
        }

}
------------------------------------------------------------
package com.ashokit.rest;

import org.springframework.beans.factory.annotation.Autowired;
import org.springframework.http.HttpStatus;
import org.springframework.http.ResponseEntity;
import org.springf                    nd.annotation.PostMapping;
import org.springf                    nd.annotation.RequestBody;
import org.springf                    nd.annotation.RestController;
```

```
Yesterday's session : REST Controller Unit Testing
--------------------------------------------------

-> We can perform Unit testing for rest controller using MockMvc.


MockMvcRequestBuilders reqBuilder =
                        MockMvcRequestBuilders.get("");

MvcResult result = mockMvc.perform(reqBuilder).andReturn();

-> We can validate rest api method results using response body conte

        int status = result.getResponse().getStatus();

-----------------------------------------------------------------

What is Mocking?
----------------
-> It is the process of creating substitute/alternate object for the

-----------------------------------------------------------------
package com.ashokit.service;

import org.springframework.stereotype.Service;

import com.ashokit.bindings.Book;

@Service
public class BookService {

        public boolean saveBook(Book book) {
                return true;
        }

}
-----------------------------------------------------------------
package com.ashokit.rest;

import org.springframework.beans.factory.annotation.Autowired;
import org.springframework.http.HttpStatus;
import org.springframework.http.ResponseEntity;
import org.springf                    nd.annotation.PostMapping;
import org.springf                    nd.annotation.RequestBody;
import org.springf                    nd.annotation.RestController;
```

3 class comments

Ganga
No att

Add

Asho
Nov 2

22-Nov-2020 C

22-Nov

4 class comm

siri 20
Sir , Pl

Add

```
Yesterday's session : REST Controller Unit Testing
--------------------------------------------------

-> We can perform Unit testing for rest controller using MockMvc.


MockMvcRequestBuilders reqBuilder =
                        MockMvcRequestBuilders.get("");

MvcResult result = mockMvc.perform(reqBuilder).andReturn();

-> We can validate rest api method results using response body conte

        int status = result.getResponse().getStatus();

-----------------------------------------------------------------

What is Mocking?
----------------
-> It is the process of creating substitute/alternate object for the

-----------------------------------------------------------------
package com.ashokit.service;

import org.springframework.stereotype.Service;

import com.ashokit.bindings.Book;

@Service
public class BookService {

        public boolean saveBook(Book book) {
                return true;
        }

}
-----------------------------------------------------------------
package com.ashokit.rest;

import org.springframework.beans.factory.annotation.Autowired;
import org.springframework.http.HttpStatus;
import org.springframework.http.ResponseEntity;
import org.springf                    nd.annotation.PostMapping;
import org.springf                    nd.annotation.RequestBody;
import org.springf                    nd.annotation.RestController;
```