

# **HSBC Technology Graduate Training**

## Spring Boot

Day 6 (Morning)

Friday 30 October 2020 | 9am

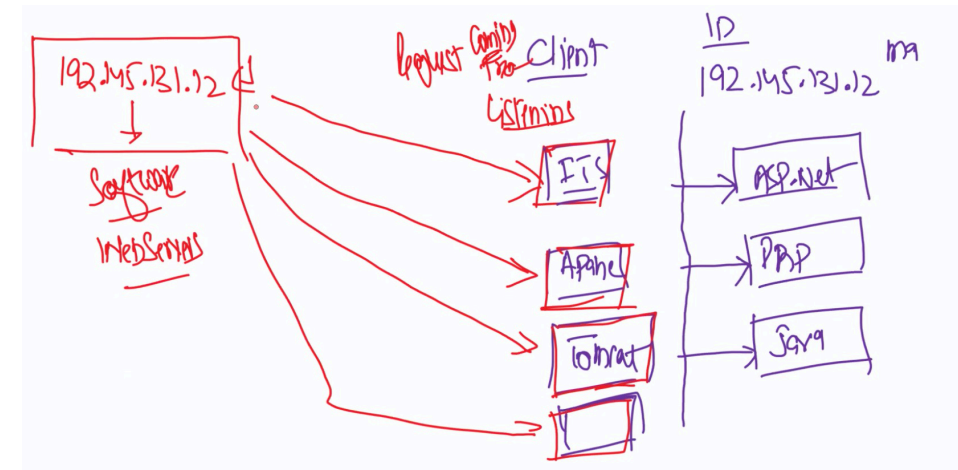
# Contents

- **Spring Boot**
- **MVC**
- **REST**
- **Packages**

**Spring Boot**

- Spring boot is a Java Framework.
- When we use web development we are referring to applications for the network.
- Web development projects are often browser based.
- EE (Enterprise Edition) applications are web development projects that are for use by all environments (even without a browser).
- Spring Boot applications will be running at some machine with some IP address.
- A client with some IP address will make requests to Spring Boot.

- You cannot send a request to a machine.
- You can only send requests to a web server software.
- A server-side machine may have many web server software 'listening' for requests.
- Each software will listen at different port numbers.
- Port numbers are unique for applications that exist on a machine.
  - You cannot have one machine with more than one app running on same port.
- From the client, we send a request to the IP address followed by Port number.
  - 192.145.121.12:80 where 80 is the port number.



- In general, many people will not know what IP addresses/port numbers they need.
- When making a request from the browser, it will make a request on port 80 by default.
  - This is a 'known' port and is a universal standard for web servers hosting websites.
- There are a list of known ports:

**HTTP: Port 80**

**SMTP: Port 25**

**Telnet: Port 23**

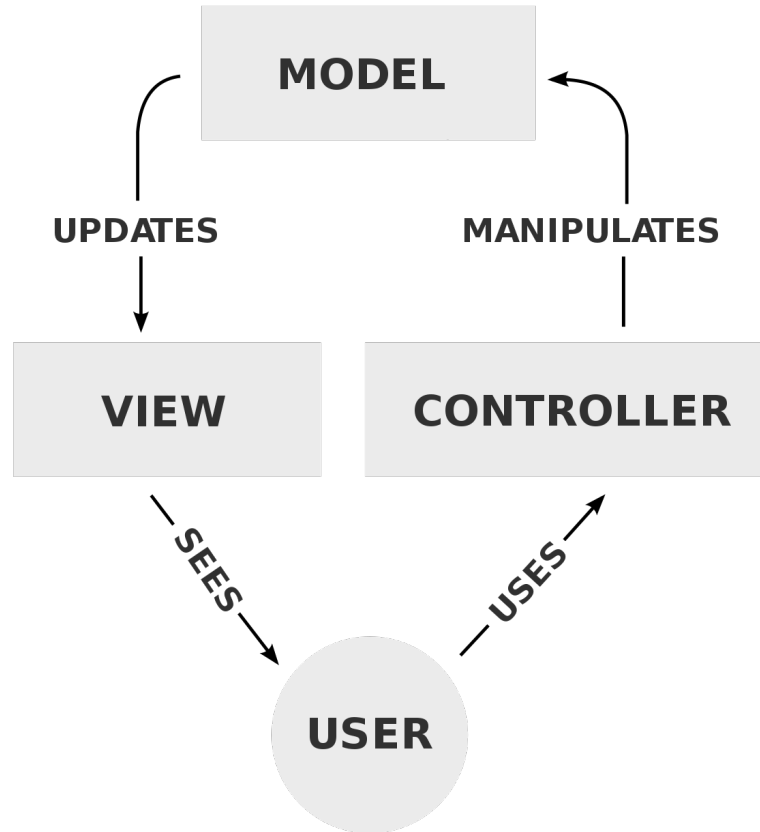
**FTP: Port 21**

**MVC**

- MVC stands for Model-View-Controller.
- It is an application design pattern.
- MVC is not specific to any language or technology.
- An application may consist of scripts, databases, webpages etc.
- We can divide our application in a logical manner using MVC.
  - Model refers to **data**. It is the model's job to handle communication to the database and make queries where required.
  - View refers to the **user interface**. It is the view's job to handle the design and display of components to allow user interaction with the application.
  - Controller refers to the **logic**. It is also the controller's job to listen for and receive requests and determine the action to be taken. It is the controller's job to handle the logic and is the 'core' of any application.



- There are various standards for how components of an MVC application should interact with each other.



## MVC: CONTROLLER

- A controller will handle your request.
- In our context, it is a Java Class.
- The controller receives a request from a browser:
  - For example, a request for **localhost:8080/login**
- In the Java Controller Class, there may be many methods, we have to perform request mapping so that the controller knows which method will be called upon which request.

**REST**

- REST stands for Representational State Transfer.
- A server may provide various servers accessible at different endpoints:
  - **/timestable**
  - **/add**
  - **/login**
  - **/subtract**
- In each of these endpoints or services, there may be different methods. For example GET or POST or DELETE or PUT.
- The endpoint indicates what resource you want and the method indicates how you should execute the request.
  - If someone requests **/welcome** using GET method, then **doGet()** should be called.
  - If someone requests **/welcome** using POST method, then **doPost()** should be called.

- **GET is usually used to retrieve data, assets, files or a webpage.**
  - **We usually send GET requests from the URL (when we access a website).**
- **POST is usually used to send data to a server to store something.**
  - **We usually send POST requests from a form on a website.**

# Packages

## PACKAGES

- A package is similar to a folder.
- For example, we organize our documents into folders.
- We can create folders inside folders and generate a hierarchy of documents & folders.
- Packages are folders which contain Java Class files.
- When we create a class in Java, it exists in some package.
- By default, the class is created in the default package.
- We can import packages in our Java projects to introduce code from other packages.

## PACKAGES

- A class that contains members without a scope identifier (e.g. public, private, protected) is automatically assumed to be protected.
- A protected data member or method is a member that is accessible from other classes, as long as that class is in the same package.