**Rest API=>**

Rest API is used to transfer raw data or raw information between service receiver and service provider.

Or

Rest api is a architectural style of writing code to exchange information between service receiver and service provider.

Or

Rest api is interface or mediator to exchange information between two system irrespective of their language.

**Note:**

Rest API is not a programming language.

**What JSON?**

JSON stands for Java Script Object Notation.

JSON is data interchange format.

**Note=>**

JSON object and Javascript object are not same but JSON object syntax is similar to Javascript object.

Different ways to transfer information in rest api =>

1. Text format
2. JSON
3. XML

Rest API converts the data into raw data.

Raw data doesn’t belong to any language.

1. Text format
2. **JSON=>**

Java Script Object Notation

JSON is data interchange format.

1. **Xml=>**

They create multiple tags.

Springboot save the memory for application.

We cannot create Springboot in eclipse.

**Spring Boot=>**

Spring Boot is module of spring from which we speed up the development.

**Spring boot is an important module in spring framework . It is widely used to build rest api’s.**

1. Our spring boot is completely based on backend.
2. In spring boot we use postman.

.

Whatever we see the data in browser is called as frontend.

**Postman=>**

1. Postman is tool which is used to check the restapi
2. Postman is used to check wheather the code is working or not .
3. In postman we will send the request our response is same as request.

**Advantage of spring boot=>**

1. Inside our boot application we have restapi.
2. Inside rest api we have controller classes.
3. Whatever the methods are created in controller class each method acts as servlet.
4. All the controller classes are stored in the rest api
5. User data is stored inside the database
6. **Post =>**

It is used to save the data.

1. **Get =>**

It is used to fetch the data.

1. **Put=>**

It is used to update the data.

1. **Delete=>**

It is used to delete the data.

Each method in the controller class is servlet.

**Steps for Spring-boot project=>**

1. First click on browser

2. Click on spring initializer

3. Then click on project= Maven, Language =java

4. Spring boot 3.06

5. Add dependency

i] Mys SQL driver

ii] Spring boot dev tools

iii] Spring web

iv] Spring data JPA

6. Click on generate and download zip file

7. Then exatract all

8. Open eclipse and import

**Data source autoconfiguration=>**

It is used to configure database details to hibernate.

**Interface Hierachy=>**

Repository-Crud Repository-Paging and sorting repository-JPA repository

**Crud repository=>**

Crud repository is used to perform crud operation.

**Paging and sorting repository=>**

We can sort the data

We divide into multiple pages.

Springboot provides implemntion to all the interfaces with the help of hibernate.

**JPA Repository=>**

It allow to access and persist data between java class and relational database.

**@restcontroller=>**

1.It is used to create controllers for the rest api.

2.@restcontroller converts the response into json format.

3.@Restcontroller annotation is the combination of @controller annotation and @resposnse body.

**@responsebody=>**

It is used to convert the response into json format.

**Note=>**

@restcontroller internally uses @responsibility.

**@Controller=>**

1. It is used to create controllers for the web application.
2. It is used to send the response.

**scanBasePackage=>**

It is an attribute which is used to specify the package name of controller to rest-api.

**@postmapping=>**

It is used to handle post requests.

**@getmapping=>**

It is used to handle the get requests.

**@putmapping=>**

It is used to handle put products.

**@deletemapping=>**

It is used to handle delete requests.

**@Entityscan=>**

This annotation is used to specify the package name of entity classes.

**@enablejparepository=>**

This annotation is used to specify the package name of our repository .It allows the programmer to use the jparepository of springboot.

**Note=>**

Whenever we create sub -interface of jpa repository .our spring boot internally creates a implementing classs of it and provided implemention for all the abstract methods with the help of hibernate.

**@DataSourceConfiguration=>**

Spring boot internally uses DataSourceAutoConfiguration class to connect our boot application with the database .DataSourceAutoConfiguration class uses hibernate internally.

**Note=>**

Spring boot-DataSourceAutoConfiguration-Hibernate-jdbc-jdbcdriver-database.

**Application.properties=>**

This file is used to specify the database configuration details and application’s properties to the spring boot.

In our spring boot save method internally call the merge method.

**@RequestBody=>**

This annotation is used to convert json object into java object.

**@ResponseBody=>**

**Optional class=>**

1. Optional class is a container which is used to store the not null values.
2. Optional class does not accept null values.

**Hwo to fetch the data from optional class?**

We can fetch the data from optional class with the help of get() method.

**Notes:**

Whenever we fetch the data from database in spring boot, spring boot internally uses optional class

**Url rewriting=>**

The process of storing the information in the url is called as url -rewriting.

**Query Variable=>**

It is a variable which is used to store the data in the url.

**@RequestBody=>**

It is used to convert json object into java object.

**@RequestParam=>**

Fetch the data from the url.