

1. What is Java?

Ans. Java is a object oriented programming language where we can write our code inside a class.

2. Java is a compiler or interpreted language?

Ans. Java is both compiled and interpreted language. It compiles using javac and interprets using Java virtual machine(JVM). JVM converts the Java code to machine understandable code. That is byte code.

3. What is Spring Boot?

Ans. Springboot is a framework of java.

4. What is Framework?

Ans. Framework is a collection of Packages where we can use the packages directly into our code rather than manually write those codes.

5. What is Packages?

Ans. Packages are the collections of classes and interfaces.

6. What is a class?

Ans. Class is the collection of properties and methods.

7. What is Objects?

Ans. Objects are the physical representation of class having also properties and methods.

8. What is POM.xml?

Ans. POM stands for Project Object Model . It is an XML file that contains information about the project and configuration details used by Maven to build the project.

9. What is XML?

Ans. XML stands for extensible markup Language. It is used to Transport and save data.

10. What is the difference between XML and HTML?

Ans. HTML is stands for Hyper text markup language used to represent the data in browser where xml is used to store data and transport data.

11. What is JDBC?

JDBC stands for Java Database Connectivity. JDBC is a Java API to connect and execute the query with the database.

12. What is Entity and attribute?

Ans. Entity means definition of the data that is table. Where as attributes means the data of that table.

13. What is @Entity? And why you write @enity?

Ans. @Entity is basically used to let the system understand that, this is class is not a normal class, this is refered as a entity class where the class is linked with a database table.

14. What is @column?

Ans. @Column is used to set the column name based on properties of a class.

15. What is @Id?

Ans. @id is refered a property or column as a primary key for that table.

16. What is Primary key?

Ans. Primary is used to uniquely identify each row of a table. This can not be null and each value must be unique.

17. What is @GeneratedValue?

Ans. This is used to auto increment a column by 1.

18. Can we change the auto increment from 1 to 2 or any other value?

Ans. Yes. We have to specify.

19. Can I only set the auto increment from java?

Ans. No we can also set it from the database. We can use identity near the column to make it auto increment. This is always the best practice to do it in database.

20. What is lombok?

Ans. Lombok is an extension of java which automatically create the getter and setter method for all the properties.

21. What is Private?

Ans. Private is an access specifier that allow the properties to access within the same class only. If we need to pass or get value from that property, then we can create the getter and setter functions for that property which shall be public.

22. What are the different access specifiers available?

Ans. Private- Access within the same class, where it is declared.

Protected- Access within the same class as well as in the child class.

Default- Access within every class but inside same package.

Public- Access everywhere.

23. Why we declare getter and setter methods in model class?

Ans. Because each the properties inside our model class are linked with the column of a database table.

To insert a value to the database or to fetch the value from the database, we need to use those property with the help of getter and setter method. Getter method is used to fetch the data and setter method are used to set the data into that property to make insertion or updation.

24. What is a repository?

Ans. Repository is basically a storage to store something. For example a data repository where various type of data can be store. Another example like a project repository where we can store and access those project. Like in our organization we used SVN and Git to store the projects.

25. What is compilation and run?

Ans. Compilation means conversion of our source code to the executable code or machine understanding code(byte code). run means if compilation is successful it will be in running mode, so that user can interact with our application.

26. Why here we create the repository?

Ans. We created a repository class to extends the Jpa repository. JPA repository is a package of spring framework that contains it's different functions to access the database such as insert data to database, fetch data from database. For example save() is a function to insert/update the data to the table. findAll() is a method to fetch all record from a table.

27. Why we declare @Repository?

Ans. We declare @Repository to let the system understand during compilation that, this interface is a repository interface for this particular model class.

28. What is controller?

Ans. Controller is used to handle the client request based on it's path and http requests like get/post/put , so that it can route to it's associated method and perform operation.

29. What is http?

Ans. Http is Hypertext transport protocol. Whatever we see in the browser is a hyper text. This set some protocol to exchange data between client and server.

30. Why we write @RestController there? What is Rest?

Ans. Rest stands for Representational State Transfer. This means We are creating APIs that just transfer the data based on our representation, this means not sharing the database. Here we write @RestController to let the System understand during run time that this is the controller class, so that when ever a request will send this will go to the controller class method.

31. What is API?

Ans. API is a communication bridge between client(User interface/Application(web/Mobile)) and server(Database). Where request will be send from the client that will be processed at API and then API will request to the database to get the information as a response.

32. What is the different format of response?

Ans. -> JSON(JavaScript Object Notation)- Allow to represent the data as a key and value format like an object

->XML (Extensible Markup Language)

33. What is @CrossOrigin?

Ans. We set the cross origin here to specify our user to access our api. For example here we mention * this means api can be accessible from anywhere. If we want only from specific server and port our api shall access, then we need to specify that server ip in place of *.

34. What is @AutoWired?

Ans. Autowired basically used to connect all the methods of a repository that we mentioned.

35. What is @GetMapping?

Ans. Get mapping is used to fetch the data from the database, that work like the select statement of the database.

36. What is @PostMapping?

Ans. Post mapping is used to insert the data to the database, that works like the insert statement in the database.

37. What is @putMapping?

Ans. Put mapping is used to update the data to the database.

38. What is @deleteMapping?

Ans. This is used to delete the record from the database.

39. What is List?

Ans. List a collection to store multiple data in a single variable.

40. What is Encapsulation and abstraction?

Ans. Encapsulation is used to hide the data and abstraction is used to hide the implementation. For example our model class is a encapsulated class and the method like findAll() and Save() of the JPA repository is abstraction because this just lets us know what to execute...but how it is executed?this hides the implementation. However we can also check the implementation by using go to definition feature of the framework.

41. What is Inheritance?

Ans. Inheritance is a concept of OOPS where child class extends the properties and methods of parent class.

Types of inheritance includes- single, multilevel, heirarchical

42. What is Interface?

Ans. Interface is used to specify behaviour of a class which is going to implement that interface.

By default inside an interface all the properties and methods are public and abstract. Function only have declaration and does not have body.