1) What is Spring ?

Spring is a framework to develop standalone and web based application and its provide some modules to develop the application.

2)What are the features is provided by spring ?

Spring provided Dependency injection and loose coupling .

3)If there is JDBC why Spring JDBC is came and what are the benifities ?

Spring JDBC is the abstraction of plain JDBC like we can say avoid boilerplate code .

4) What is Spring?

Is a dependency injection framework.

It is basically used to manage the life cycle of java classes (beans).

It consists of a lot of boilerplate configuration.

Uses XML based configuration. It takes time to have a spring application up and running and it’s mainly

5)what is IOC ?

Inversion of control to manage our life cycle of beans .

6)what are the diff b/w java bean and spring bean and pojo ?

Java beans is only plain class like -DTO so we can transfer the data into one layer to other layer

Spring beans is nothing but those beans 7is managed by IOC contaioner is called Spring Beans

Pojo is stands for Plan old java class .

7)what is tight coupling?

Degree of dependency between two components is high is called tight coupling

ex-Inheritance

8)what is loose coupling ?

Degree of dependency between two components is less is called tight coupling

ex-interface model programming

9)what is dependency injection in spring?

Dependency injection is nothing but to injecting one bean to other bean by using setter injection and constructor injection.

10)what is target class ?

A class which is getting and pulling the object from different resources is called target class.

11)what is dependent class ?

A class provided the services to other class is called dependent class.

12)how we can achieve dependency injection ?

we can achieve dependence injection in 2 way like -setter and constructor injection .

13)what is configuration file and what is the use of it ?

Confuguration file contains confuguration of bean class so that the IOC contaioner will manage our application by taking the help of Configuration file .

14)what is spring jdbc ?

Spring JDBC is the abstraction of JDBC.

15)what is jdbcTemplete?

JdbcTemplate is a powerful mechanism to connect to the database and execute SQL queries. It internally uses JDBC api .

16)What is SpringBoot?

SpringBoot is a framework of frameworks like as Hibernate, JPA,security.

It is used to create a stand-alone Spring-based application that you can just run because it needs minimal Spring configuration.

17)Explain the need of dev-tools dependency

DevTools when we make changes to Java code or properties file, the application gets updated with new changes.

It monitors for changes and automatically restarts the application

18)what is spring starter ?

**Spring Boot** provides a number of **starters** that allow us to add jars in the classpath. Spring Boot built-in**starters** make development easier and rapid.

Spring Boot Starters are dependency descriptors that can be added under the <dependencies> section in pom. xml.

19)what are the starter provided by spring boot ?

spring-boot-starter-thymeleaf : It is used to build MVC web applications using Thymeleaf views.

spring-boot-starter-web-services : It is used for Spring Web Services.

spring-boot-starter-web : It is used for building the web application, including RESTful applications using Spring MVC. It uses Tomcat as the default embedded container.

spring-boot-starter-test : It is used to test Spring Boot applications with libraries, including JUnit, Hamcrest, and Mockito.

spring-boot-starter-jdbc: It is used for JDBC with the Tomcat JDBC connection pool.

spring-boot-starter-security: It is used for Spring Security.

20)what are the features available in springboot?

* Web Development.
* SpringApplication.
* Application events and listeners.
* Admin features.
* Externalized Configuration.
* Properties Files.
* YAML Support.
* Type-safe Configuration.

21) what is entity and what is the use of it ?

Entity is meant for to mapping one class to Db with the help of Data jpa.

22)is there any facilities to change the server running port number dynamically ?

Yes,Server.port in application.properties file

23)what is @Repository annotation and when we can use and which layer it is applicable ?

@Repository is applicable in DAO layer and it will commnucate to DB by and it will internally execute the some query .

25)what are the annotation uses in entity class ?

@id,@Entity,@Table,@Column

26)what is @Autowired ?

Autowired annotation is ment for linking to class and creating one class object in side the other class.

27)what is @Controller and @RestController ?

|  |  |
| --- | --- |
| @Controller | @RestController |
| @Controller is used to mark classes as Spring MVC Controller. | @RestController annotation is a special controller used in RESTful Web services, and it’s the combination of @Controller and @ResponseBody annotation. |
| It is a specialized version of @Component annotation. | It is a specialized version of @Controller annotation. |
| @Controller annotation indicates that the class is a “controller” like a web controller. | @RestController annotation indicates that class is a controller where @RequestMapping methods assume @ResponseBody semantics by default. |
| In @Controller, we need to use @ResponseBody on every handler method. | In @RestController, we don’t need to use @ResponseBody on every handler method. |
| It was added to Spring 2.5 version. | It was added to Spring 4.0 version. |

28)What is scp?

String Pool, also known as SCP (String Constant Pool), is a special storage space in Java heap memory that is used to store unique string objects. Whenever a string object is created, it first checks whether the String object with the same string value is already present in the String pool or not, and if it is available, then the reference to the string object from the string pool is returned. Otherwise, the new string object is added to the string pool, and the respective reference will be returned.

29) How to declare a string in Java?

String declaration in Java can be done in two ways:

By string literal: Double quotes are used to create Java String literals.

Example: String str= "Rohini";

By new keyword: Keyword "new" is used to create a Java string.

Example: String str=new String ("Rohini");

### 30) Is String immutable or final in Java? If so, then what are the benefits of Strings being Immutable?

Yes, Strings are immutable in Java. Immutable objects mean they can't be changed or altered once they've been created. However, we can only modify the reference to the string object. The String is immutable in Java because of many reasons like security, caching, synchronization and concurrency, and class loading

31) String Vs StringBuffer Vs StringBuilder

|  |  |  |  |
| --- | --- | --- | --- |
|  | String | StringBuffer | StringBuilder |
| Storage Area | Heap, SCP | Heap | Heap |
| Mutability | Immutable | Mutable | Mutable |
| Thread Safe | Not Thread safe | Thread safe | Not Thread safe |
| Performance | Slow | Faster than String | Faster than StringBuffer |
| Memory | Consumes more memory | Less | Less |

32)Pillars of OOP’s

1. Abstraction :

Hiding the implementation from the user but shows only essential information to the user.

2. Encapsulation:

[Encapsulation](https://www.javatpoint.com/encapsulation) is a mechanism that allows us to bind data and functions of a class into an entity. It protects data and functions from outside interference and misuse. Therefore, it also provides security. A class is the best example of encapsulation.

3. Inheritance: The concept allows us to inherit or acquire the properties of an existing class (parent class) into a newly created class (child class). It is known as [**inheritance**](https://www.javatpoint.com/inheritance-in-java). It provides code reusability.

4. Polymorphism:

It allows us to create methods with the same name but different method signatures. It allows the developer to create clean, sensible, readable, and resilient code.

33)Array Vs ArrayList

| Basis of Comparison | Array | Array List |
| --- | --- | --- |
| Static/Dynamic | Arrays are static | ArrayList is dynamic |
| Resizable | Fixed Length | Can be Resizable |
| Initialization | When performing initialization for an array, it is required to specify the size of the array. | It is not necessary to mention the size of an ArrayList. |
| Performance | Arrays are faster | ArrayList is slower |
| Generic Type | An array can store primitive data as well as objects, but it cannot store generics. | ArrayList is able to store generics as well as objects, but it cannot store data of primitive types. |
| Iteration | Only loops are permitted in this area. | It is acceptable to use loops and iterators. |
| Type Safety | It is not type-safe. | It is type-safe. |
| Length | Makes use of the length object | Make use of size() function |
| Adding Elements | The additions are done with the assignment operator. | ArrayList uses add() method for performing additions |
| Single/Multi- Dimensional | Single and multiple dimensions are also a possibility. | You are only permitted to use single dimension. |

34)Unchecked exception vs Checked Exception

|  |  |
| --- | --- |
| Checked Exception | Unchecked exception |
| Checked exceptions occur at compile time. | Unchecked exceptions occur at runtime. |
| The compiler checks a checked exception. | The compiler does not check these types of exceptions. |
| These types of exceptions can be handled at the time of compilation | These types of exceptions cannot be a catch or handle at the time of compilation, because they get generated by the mistakes in the program. |
| They are the sub-class of the exception class. | They are runtime exceptions and hence are not a part of the Exception class. |
| Here, the JVM needs the exception to catch and handle. | Here, the JVM does not require the exception to catch and handle. |
| Examples of Checked exceptions:   * File Not Found Exception * No Such Field Exception * Interrupted Exception * No Such Method Exception * Class Not Found Exception | Examples of Unchecked Exceptions:   * No Such Element Exception * Undeclared Throwable Exception * Empty Stack Exception * Arithmetic Exception * Null Pointer Exception * Array Index Out of Bounds Exception * Security Exception |

35)HashMap Vs HashSet

|  |  |  |
| --- | --- | --- |
| Basis | HashMap | HashSet |
|  | Java HashMap is a hash table based implementation of Map interface. | HashSet is a Set. It creates a collection that uses a hash table for storage. |
| Implementation | HashMap implements Map, Cloneable, and Serializable interface es. | HashSet implements Set, Cloneable, Serializable, Iterable and Collection interfaces. |
| Stores | In HashMap we store a **key-value pair**. It maintains the mapping of key and value. | In HashSet, we store objects. |
| Duplicate Values | It does not allow duplicate keys, but duplicate values are allowed. | It does not allow duplicate values. |
| Null values | It can contain a single null key and multiple null values. | It can contain a single null value. |
| Storing Mechanism | HashMap internally uses hashing to store objects. | HashSet internally uses a HashMap object to store objects. |
| Method of insertion | HashMap uses the put() method to add the elements in the HashMap. | HashSet uses the add() method to add elements in the HashSet. |
| Performance | HashMap is faster/ than HashSet because values are associated with a unique key. | HashSet is slower than HashMap because the member object is used for calculating hashcode value, which can be same for two objects. |
| The Number of objects | Only one object is created during the add operation. | There are two objects created during put operation, one for key and one for value. |

36) What is Junit Testing?

It is an open-source testing framework for java programmers. The java programmer can create test cases and test his/her own code. It is used to check the correctness of the software.

To perform unit testing, we need to create test cases. The unit test case is a code which ensures that the program logic works as expected.

37)What is Mokito?

Mockito is a Java-based mocking framework used for unit testing of Java application. Mockito plays a crucial role in developing testable applications.

The main purpose of using the Mockito framework is to simplify the development of a test by mocking external dependencies and use them in the test code. As a result, it provides a simpler test code that is easier to read, understand, and modify.

38)What is @Mock?

Using this annotation it will create the mock objects. Mock objects act as a dummy or clone of the real object in testing. They are generally created by an open-source library or a mocking framework like Mockito, EasyMock, etc. Mock objects are typically used for behavior verification.

39)What is Stub?

Stub objects hold predefined data and provide it to answer the calls during testing. They are referred to as a dummy object with a minimum number of methods required for a test. It also provides methods to verify other methods used to access the internal state of a stub, when necessary. Stub object is generally used for state verification.

40)What is @Spy?

A Spy is like a partial mock, which will track the interactions with the object like a mock. Additionally, it allows us to call all the normal methods of the object. Whenever we call a method of the spy object, the real method will be invoked. It will directly call the real objects without changing its behaviour.

41) Difference between Spring and Spring Boot?

|  |  |
| --- | --- |
| Spring | Spring Boot |
| Spring Framework is a widely used Java EE framework for building applications. | Spring Framework is a widely used Java EE framework for building applications. |
| It aims to simplify Java EE development that makes developers more productive. | It aims to shorten the code length and provide the easiest way to develop Web Applications. |
| The primary feature of the Spring.Framework is dependency injection. | The primary feature of Spring Boot is Autoconfiguration. It automatically configures the classes based on the requirement. |
| It helps to make things simpler by allowing us to develop loosely coupled applications. | It helps to create a standalone application with less configuration. |
| The developer writes a lot of code (boilerplate code) to do the minimal task. | It reduces boilerplate code. |
| To test the Spring project, we need to set up the sever explicitly. | Spring Boot offers embedded server such as Jetty and Tomcat, etc. |
| It does not provide support for an in memory database. | It offers several plugins for working with an embedded and in-memory database such as H2. |
| Developers manually define dependencies for the Spring project in pom.xml. | Spring Boot comes with the concept of starter in pom.xml file that internally takes care of downloading the dependencies JARs based on Spring Boot Requirement. |