**GIT**

Que. How to create a new branch using git command?

Ans. git branch

git branch <branch-name> e.g branch name is “feature-branch” then git branch feature-branch

after creating branch switch to new created branch using the command git checkout feature-branch.

Alternatively you can combine branch creation and checkout steps into single command

**Spring Boot**

Que. Dao full form

Ans. Data Access Object

Usually, the DAO class is responsible for two concepts encapsulating the details of the persistence layer and providing a CRUD interface for a single entity.

Que. Difference between get post put update delete request type

Ans. GET – Reads a resource

PUT – Replaces a resource

Post – Creates a resource ,is used to adda child resource ,it can be catched

UPDATE – It executes an update statement

DELETE – Delete a resource

Que. POJO, Model, Service

Ans. POJO- POJO is a Plain Old Java Object . It does not require any special classpath. POJO class is used to define the object entities. The POJO class is an object class that encapsulates the Business logic. the Controller interacts with the business logic, which contacts with POJO class to access the data.

The major advantage of the POJO class is we will not have to create objects every time in other Java programs. Simply we can access the objects by using the get() and set() methods.

To access the objects from the POJO class, follow the below steps Create a POJO class objects, Set the values using the set() method, Get the values using the get() method ,

Does not have any naming conventios for properties and methods

Model – In spring model works as a container that contains data of the application, data can be in any form such as objects, strings, information from database. It is required to place model interface in controller part of the application. Object of the HttpServletRequest reads info. Provided by user and pass it to the model interface.

Service - Spring boot service component is used as a class file that includes the @Service annotation and allows developers to add business functionalities. The annotation is used with the classes that provide these business functionalities.

@Service annotation is used in your service layer and annotates classes that perform service tasks

Que.JDBC template

Ans. JDBC is a Java database connectivity . JDBC is a java API to connect and execute query with the database , JDBC API uses JDBC drivers to connect with the database

We can perform all the database operations by the help of JdbcTemplate class such as insertion, updation, deletion and retrieval of the data from the database.

Spring Boot automatically creates a JdbcTemplate . JdbcTemplate class is the **central class in the JDBC core package**.

We have to import org.springframework.jdbc.core.JdbcTemplate in dao class

provides you methods to write the queries directly, so it saves a lot of work and time.  
 We have to import org.springframework.jdbc.core.JdbcTemplate in dao class

Commonly user **methods of JdbcTemplate class**:  
1) public int update(String query) is used to insert, update and delete records.  
2) public int update(String query,Object... args) is used to insert, update and delete records using PreparedStatement using given arguments.  
3) public void execute(String query) is used to execute DDL query.  
4) public T execute(String sql, PreparedStatementCallback action) executes the query by using PreparedStatement callback.  
5) public T query(String sql, ResultSetExtractor rse) is used to fetch records using ResultSetExtractor.  
6) public List query(String sql, RowMapper rse) is used to fetch records using RowMapper