***Categorization of Methods***

|  |  |  |
| --- | --- | --- |
| Method | Belongs To | Explanation(Hinglish) |
| Authenticate | Business Logic | Ye check karta hai ki user ka username/password sahi hai ya nahi – yeh ek rule hai, toh business logic mein aata hai. |
| Search | Data Access + Business Logic | Agar simple database se data nikalna hai, toh Data Access Logic. Agar koi filter, sorting, ya custom rule lage toh Business Logic bhi involved ho sakti hai. |
| Change Password | Business + Integration Logic | Password validate karna Business Logic; external service (like OTP API or email API) ko call karna Integration Logic. |
| Forgot Password | Business + Integration Logic | Token generate karna Business Logic; reset link email bhejna ya SMS API call karna Integration Logic. |
| Add (Create) | Data Access + Business Logic | Validate karna (jaise email unique hai ya nahi) business logic hai. Lekin record database mein insert karna Data Access Logic. |
| Update | Data Access + Business Logic | Same – validate/update rules business logic, actual DB update = Data Access Logic. |
| Delete | Data Access + Business Logic | Agar directly delete kar rahe ho, toh Data Access. Lekin agar soft delete ya kuch check lagana ho before delete (jaise "order already shipped, delete not allowed") – toh business logic bhi. |
| User Registration | Integration Logic | External API (like email verification service ya CRM system) ke saath integrate karna Registration ke dauraan hota hai.  S |

***Summary in Simple Hinglish***

* Rules check karna, validation lagana, decision lena = Business Logic
* Database se data lena, save karna, update/delete karna = Data Access Logic
* Often, real-world apps mein dono logic saath mein kaam karte hain, but their responsibilities clearly alag hoti hain.
* Agar chaho, main ek simple code structure bhi bana sakta hoon to show how they’re separated.

**Q:- What is Maven?**

**Ans:-** Maven is a powerful build Automation tools.

**Q:- What is POM.XML?**

**Ans:-**  It is XML File that contain Information about the project and contain configuration detail used by Maven to build Project. The full form of POM.XML is “**Project Object Model . Extensible Markup Language**”.

**NOTE:-** Short Ans.

It Contains Project Information and Configuration

**Q:-** **What are dependencies in pom.xml?**

**Ans:-** In Maven, a dependency is an archive file like JAR, ZIP, etc., that your project needs to compile, build, test, and run. These project dependencies are defined in the pom. xml file.

**Q:-How many Dependencies used in project?**

**Ans:-**

1. log4j :- 1.2.17
2. My SQL :- 8.0.29
3. JavaX Mail :- 1.4.7
4. Javax Servlet :- 3.0.1
5. C3P0 :- 0.9.1.2
6. Junit :- 3.8.1

**Q:- What is WEB.XML?**

**Ans:-** it is XML file that contains application configuration and it is also called deployment descriptor.

**Q:- What is Utility Classes?**

**Ans:-** Utility classes are used to reusability and reuseable service.

**Q:- what are the Utility Classes you have in your project?**

**Ans:-**

|  |  |  |
| --- | --- | --- |
| S.NO. | Utility Class | Defination |
| 1. | 1. **Data Validator** | Used to Validate input Data Entered by User. |
| 2. | 1. **Data Utility** | Used to Format the Data from one format to another. |
| 3. | 1. **Email Utility** | Provide Email Service. |
| 4. | 1. **Email Builder** | Used to build Email Message. |
| 5. | 1. **Email Message** | Contains Email Message. |
| 6. | 1. **HTML Utility** | Used to produce HTML content like DropDown List. |
| 7. | 1. **JDBCDataSource** | For DCP (Data Connection Pool) |
| 8. | 1. **Property Reader** | Used to read the property values from application properties file using Resource Bundle |
| 9. | 1. **Servlet Utility** | This Class provides Utility operation for Servlet Container like Forward, Redirect, Handle Generic Exception, manage Success and Error Message, manage default Bean and List, manage Pagination Parameters |

**Q:- What is URL and URI?**

**Ans:-** The Full Form of URL is “UNIFORM RESOURCE LOCATOR”

And it Contains Protocol, Socket, and URI.

The Full Form of URI is “UNIFIED RESOURCE IDENTIFIER”

And It checks resource exists or not

**Q:- What is Session?**

**Ans:-** Session is conversation between browser web client and web server.

**Q:- How to Destroy Session?**

**Ans:-**

* Session.invalidate(); (Minutes).
* <Session config> (Minutes)

<Session timeout>20</ Session timeout>

</Session config>

* Session.setMaxInactiveInterval(); (Seconds)

**Q:- How to set session timeout in web.xml?**

**Ans:-**  <Session config> **(Minutes)**

<Session timeout>20</ Session timeout>

</Session config>

**Q:- What is default timeout for session?**

**Ans:-** 30 Minutes.

**Q:-What is Design Pattern?**

**Ans:-** Standard Problem Has Standard Solution that is Knows as Design Pattern.

There are 4 types of Design Patter.

* **Singleton Design Pattern:-** The class that has only one instance in their lifetime.
* **Builder Design Pattern:**-Create complex object using simple object using step by step approach.
* **Factory Design Pattern**:- The class that has ability to create object of another class is called Factory **Design Pattern**.
* **Front Controller Design Pattern**:- Main Controller perform session checking and login operation before calling any application controller. It prevent any user to access an application without login.

**Q:-How Many types of Logics?**

**Ans:-**

* **Presentation Logic:-**  This is a set of control statement that decide look & feel of User Interface(JSP).
* **Control Logic:-** This is a set of control statement that decide navigation of User Interface it is also called **“Navigation Logic”**.
* **Business Logic:-** This is a set of control statement that perform Business Operations like (Change passwords and forget password).
* **Data Access Logic:-** This is a set of control statement that perform and Make Changes in Database with the help of JDBC like(CRUD Operations).
* **Integration Logic:-** This is a set of control statement that integrate application with another application or server(Send E-Mail).

**Q:-what is MVC ? and its Guidelines?**

**Ans:-** MVC is a Framework Methodology. That separates code implementation of an application into three components

1. Model
2. View
3. Controller

* **Model:->** Model contain Business Logic, Data Access Logic, and Integration Logic.
* **View:->** View Contain Presentation Logic
* **Controller:->** Controller contain Control(Navigation) Logic.

**Guidelines Of MVC**

* One Screen One View.
* One View One Controller.
* View is always access by Controller.
* View always submit request to its own Controller.
* Controller will perform Business Operations by Calling Model Business Method.

**Q:- What is Servlet Lifecycle?**

**Ans:-**

* init();
* service();
* destroy();

**Q:-Difference between Prepared Statement (PSTMT) and Statement (STMT) ?**

**Ans:-** Prepared Statement preserve the parsed Query and reuse that Query with different parameters. Whereas Statement Query are passed in every Database call.

**Q:-What is Callable Statement ?**

**Ans:-** It is Child Interface of Prepared Statement . it is used to execute stored procedure and stored function by JDBC.

**Q:-What is Validation & its Types ?**

**Ans:-**Validate the data Entered By User and through Database.

There are two types of Validations

1. Server Side Validation

* Input Validations:-> Checks input data entered by User.
* Business Validations:-Checks the Business Condition that may need Database communication.

1. Client Side Validation

Uses JavaScript.

**Q:- What is Variable ?**

**Ans:-** Variables allows Java program to store the data during the runtime of a program.

**Q:- What is Object ?**

**Ans:-** Object is an Instance of an Class.

**Q:- What is Constructor ?**

**Ans:-** Constructor is an Instance method that has the same name as the Class name. Constructor do not have any return type.

**Q:- What is Method ?**

**Ans:-** Method is a set of instructions grouped together to perform operations it is executed when it is called.

**Q:- What is final Keyword ?**

**Ans:-** The Final Keyword is used to restrict the Variable, Class, and Method.

* **Final Variable: -** If the Variable is defined as final then it’s value can not be changed and variable become constant.
* **Final Class: -** If the Class is defined as final then it cannot be inherited.
* **Final Method:** - If the Method is defined as final then it cannot be overridden by child class.

**Q:- What is Finally Block ?**

**Ans:-** Finally Block is always executed after the Try Catch block.

**Q:-What is Finalize ?**

**Ans:-** Finalize is called by the Garbage Collector when object is removed from memory.

**Q:- Difference between DoGet and DoPost ?**

**Ans:-**

|  |  |
| --- | --- |
| DoGet | DoPost |
| Perform View Logic | Perform Submit Logic |
| Can be Cached | Can not be Cached |
| Can be Bookmarked | Can not be Bookmarked |
| Can send only text and number | Can send Binary Data |
| Parameter are saved in browser history  because they are part of URL | Parameters are not saved in browser history |

**Q:- What is JSP ?**

**Ans:-** JSP Stands for **Java Server Page** it is a special form of servlet to develop web pages. JSP allows you to embedded java code inside HTML pages using JSP tags.

**Q:- What is Servlet ?**

**Ans:-** Servlet are special types of java classes to produce dynamic web content to develop web applications.

**Q:-What is JSP/Servlet ?**

**Ans:-** JSP and Servlet are the server side program to produce dynamic response on a client request these are written in java language.

**Q:- Difference Between Overloading and Overriding ?**

**Ans:-**

|  |  |
| --- | --- |
| OverLoading | OverRiding |
| Define multiple methods with same name  But different parameter in the same class | Redefine the Method of an Parent Class in  The Child Class |
| Inheritance is not required | Inheritance is required |
| Happen in the same class | Happen in the sub class |

**Q:- What is OOP ?**

**Ans:-** Object Oriented Programming represents a programming methodology based on Object instead of just procedures and functions.

**Q:-What are key concepts of OOP ?**

**Ans:-** There are three key concepts of OOP.

* Encapsulation :-> Create Expert Classes.
* Inheritance :-> Cerate Specialized Classes.
* Polymorphism:-> Provides Dynamic Behaviour.
* **Encapsulation :->** Gathering all related method and attributes in a class is called Encapsulation
* **Inheritance** :-> A class may inherit or reuse the properties