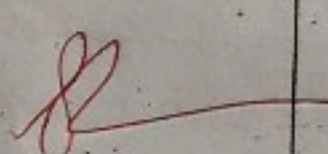


Assignment no :- 06

Topics covered :-
- Selection
- JDBC tutorial
- JSP

Date of performance :- 06-10-22

Evaluation Criteria	Marks (out of 3)	Date	Signature of Instructor
Punctuality	03	07/10/2022	
Problem solving technique	03		
Attainment level (out of 3)	03		

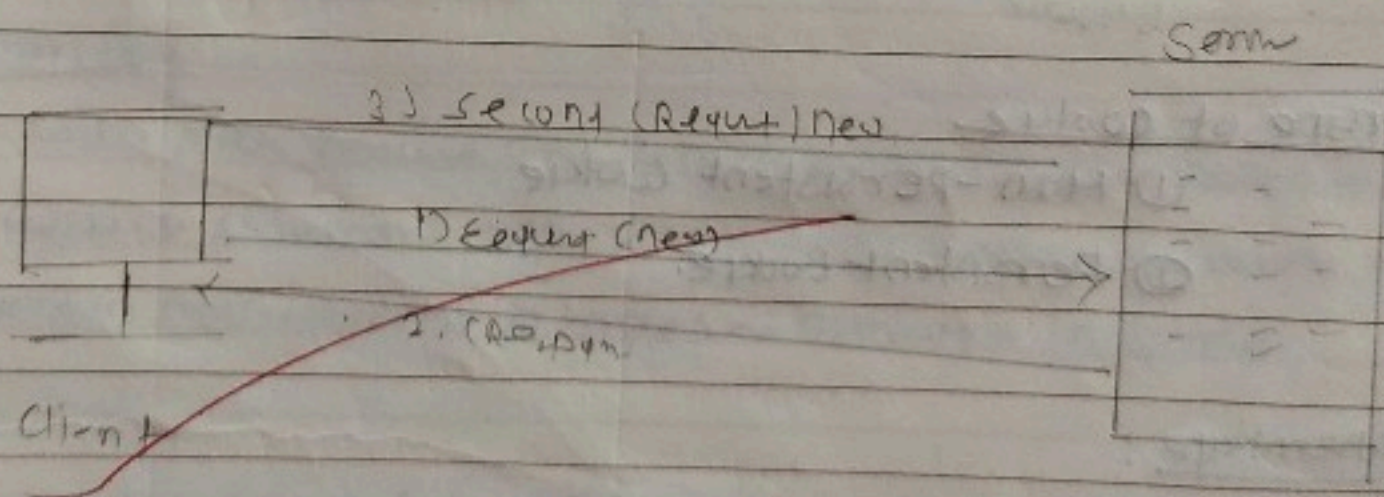
Q.1) What is Session? list session tracking technique

• Session :- Simply means a particular interval of time.

• Session Tracking is way to maintain state (data) of an user. it also known as session management in server.

• HTTP protocol is Stateless so we need to maintain state using session tracking techniques. Each time user request to the server, server treats the request as the new request. so we need to maintain the state of an user to recognize the particular user.

• HTTP is Stateless that means each request is considered as the new request. it show fig.



• Why Use

to recognize the user it is used to recognize the particular user.

Session Tracking techniques:

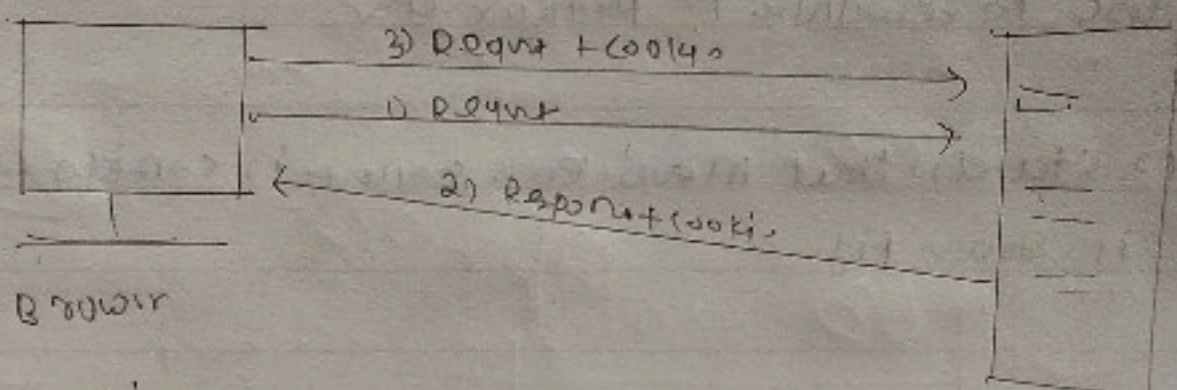
There are four techniques in session tracking:

- i) cookies
- ii) Hidden form field
- iii) URL Rewriting in HttpSession.

i) Cookies

A cookie is small pieces of information that is persisted between multiple client requests.

Each request is considered as a new request. In cookies technique, we add cookie with response from the server. After that it request by default.



Type of cookie

- ① Non-persistent cookie
- ② Persistent cookie.

Advantage:

- ① Simplest technique of maintaining the state.
- ② Cookies are maintained at client side.

Disadvantages:

- ① It will not work if cookies are disabled from browser.
- ② Only textual information can be set in cookie object.

ii) Hidden form field

The information is inserted into the web page via the hidden form field, which is then transfer to the server.

Then field on hidden form the user's view

`<input type="hidden" name="session" value="12345">`

Advantage

- ① It will always work whether cookies is disable or not

Disadvantage

- ① It is maintain on server side
- ② Only textual information can be used

iii) URL Rewriting

With each Request and Return, append some more data via URL as Request parameter. URL rewriting is better technique to keep session management & browser operation in sync.

Advantage

1. It will always work whether cookies is disable or not
2. Extra form submission is not required to on each page

Disadvantage

- i) It will work only with links
- ii) It can send only textual information.

iv) HttpSession

In back end container, create a session id for each user. The container use this id to identify the particular user. An object of HttpSession can be used to perform two things

- i) Get object
- ii) view & manipulate information about session

Illustration

```
HttpSession session = request.getSession();  
session.setAttribute("username", "Anshu");
```

The request must be made before sending any document content to the client, you must first call getSession().

② Write down how you going to do a Database connection (JDBC)

⇒ Java JDBC Tutorial :

JDBC stands for Java Database Connectivity. JDBC is Java API to connect & execute the query with the database. It is a part of Java SE.

JDBC API uses JDBC driver to connect with the database.

• Database Connectivity :-

There are 5 steps to connect any Java application with the database using JDBC. These steps are as follows :-

- Register the Driver Class
- Create connection
- Create statement
- Execute queries
- Close connection.

1) Register the driver class.

The `forName()` method of `Class` class is used to register the driver class. This method is used to dynamically load the driver class.

Syntax of `forName()` method :

`public static void forName() method (String className) throws ClassNotFoundException`

Ex. Conn. for Name ("oracle.jdbc.driver.OracleDriver");

2) Create the Connection Object:

The getConnection method of DriverManager class is used to establish connection with the database.

Syntax

- 1) Public Static Connection getConnection (String url) throws SQLException
- 2) Public Static Connection getConnection (String url, String name, String password) throws SQLException

Ex.

```
Connection con = DriverManager.getConnection ("jdbc:oracle:thin:@localhost:1521:xe", "system", "password");
```

3) Create the Statement Object

The createStatement() method of Connection interface is used to create Statement. The object of Statement is used to execute query with database.

Syntax :-

Public Statement createStatement() throws SQLException

Ex. Statement stmt = con.createStatement();

4) ExecuteQuery

The executeQuery() method of Statement interface is used to execute queries to the database. This method returns the object of ResultSet that can be used to get all the records of a table.

Syntax

```
public ResultSet executeQuery (String sql) throws Exception
```

```
Ex . ResultSet rs = stmt.executeQuery ("select * from emp");
    while (rs.next())
    {
        System.out.println (rs.getInt(1) + " " + rs.getString(2));
    }
```

5) Close the connection object

By closing connection object Statement & ResultSet will be closed automatically. The close() method of Connection interface is used to close the connection.

Syntax of close() method

```
public void close() throws SQLException
```

Ex

```
con.close();
```


Q.3) Explain the all type JDBC driver

JDBC driver is software component that enable java application to interact with the database. There are 4 type of JDBC drivers:

- JDBC-ODBC bridge driver
- Native-API driver
- Network protocol driver
- Thin driver

1) JDBC-ODBC bridge driver

The JDBC-ODBC-bridge driver use ODBC driver to connect to the database. The JDBC-ODBC driver convert JDBC method call into the ODBC function call. This is now discontinue because of thin driver.

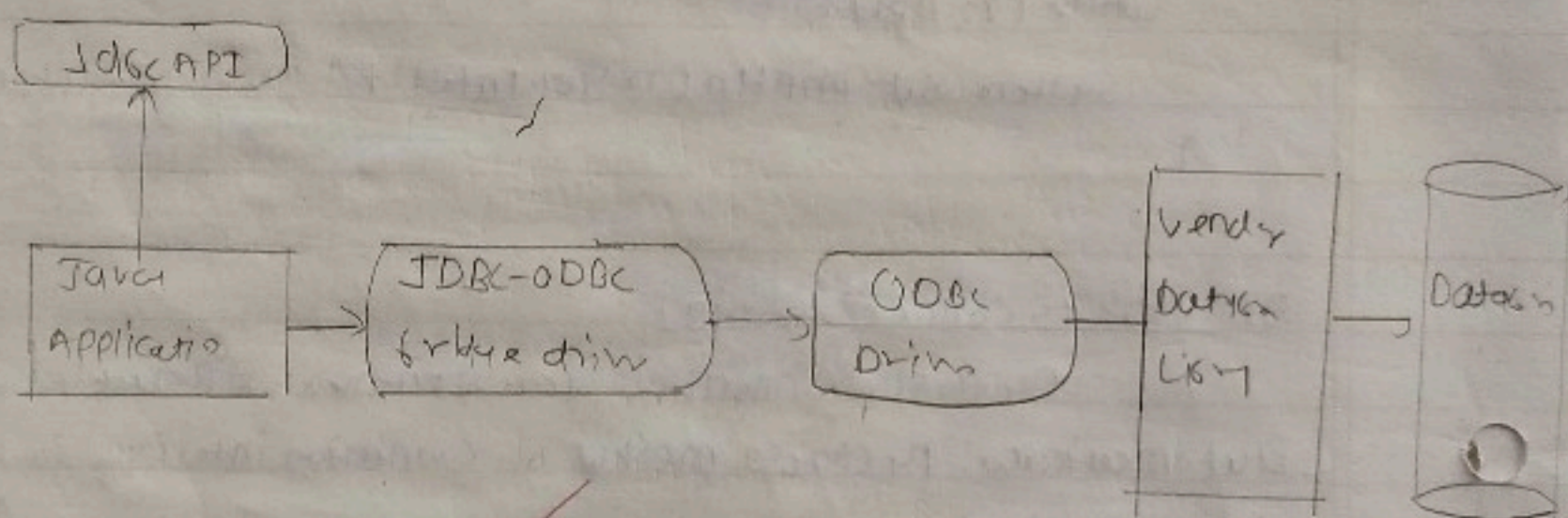


Fig. JDBC-ODBC-Bridge driver

Oracle does not support the JDBC-ODBC-Bridge from Java & Oracle recommends that you use JDBC driver provided by the vendor of your database instead of the JDBC-ODBC bridge.

Adv?

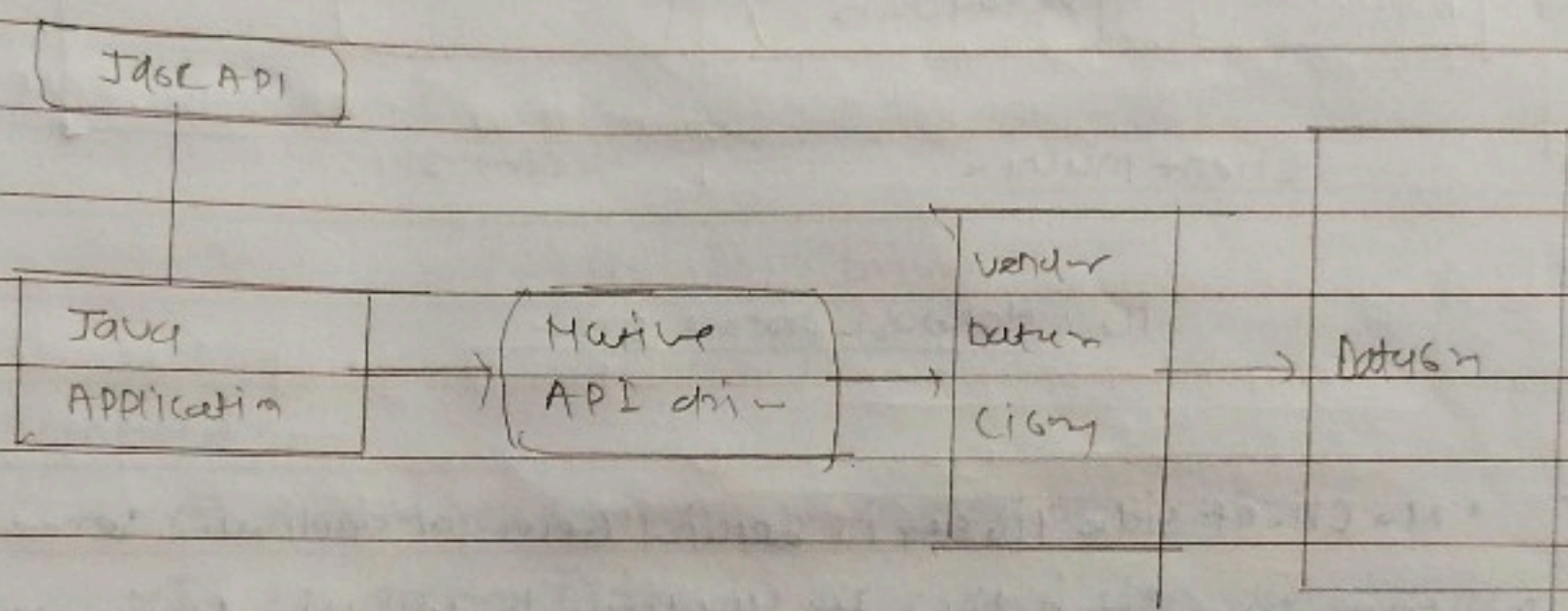
- ① Easy to use
- ② Only connect to any database

Dis? ① It decodes each JDBC method call is converted into ODBC function call.

2) Native-API driver

The Native API driver uses the client-side libraries of the database.
The driver connects JDBC method call into native call of the database API.

It is not written entirely in Java



His Native API Driver

Advantage

- ① Perform better than JDBC-ODBC bridge driver

Disadvantage

- ① The Native driver need to be installed on each client machine
- ② The Vendor client library need to be installed on client machine

3) Network protocol driver

The Network protocol driver use middle that convert JDBC call directly or indirectly into the vendor-specific database protocol. it is fully written in Java.

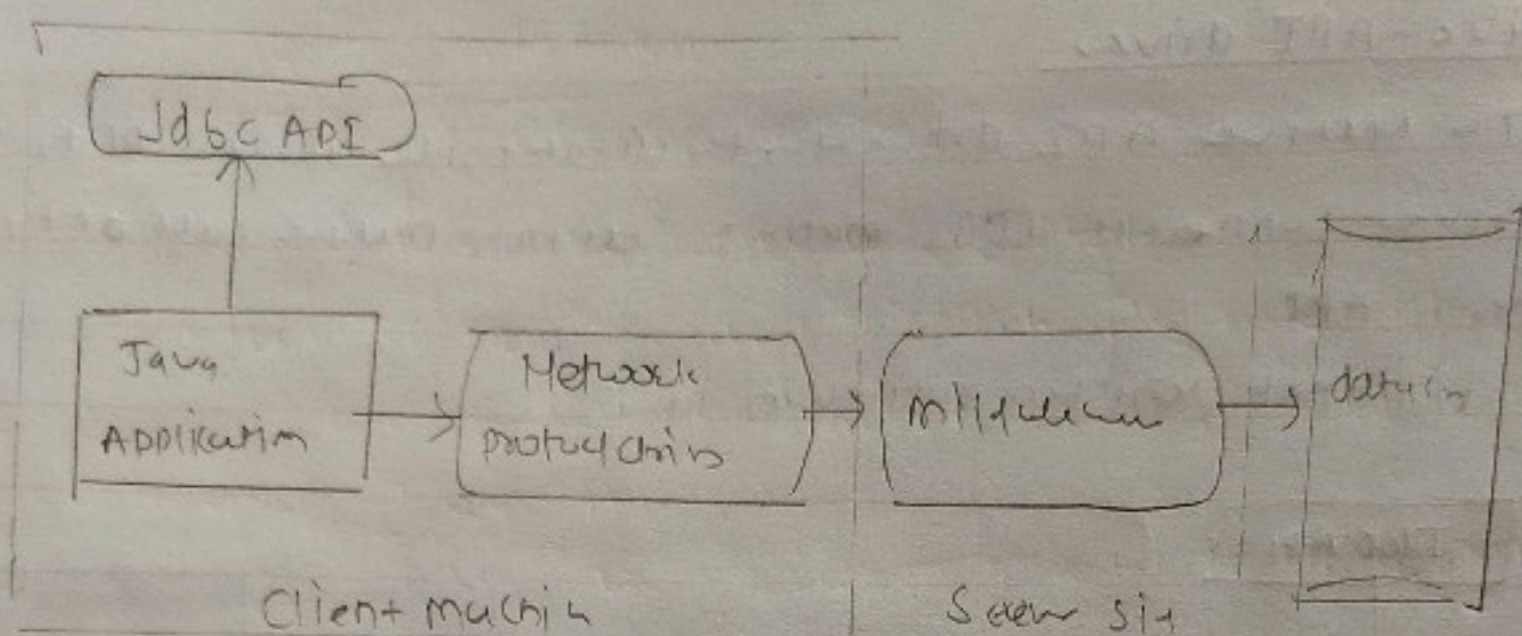


Fig. Network protocol driver

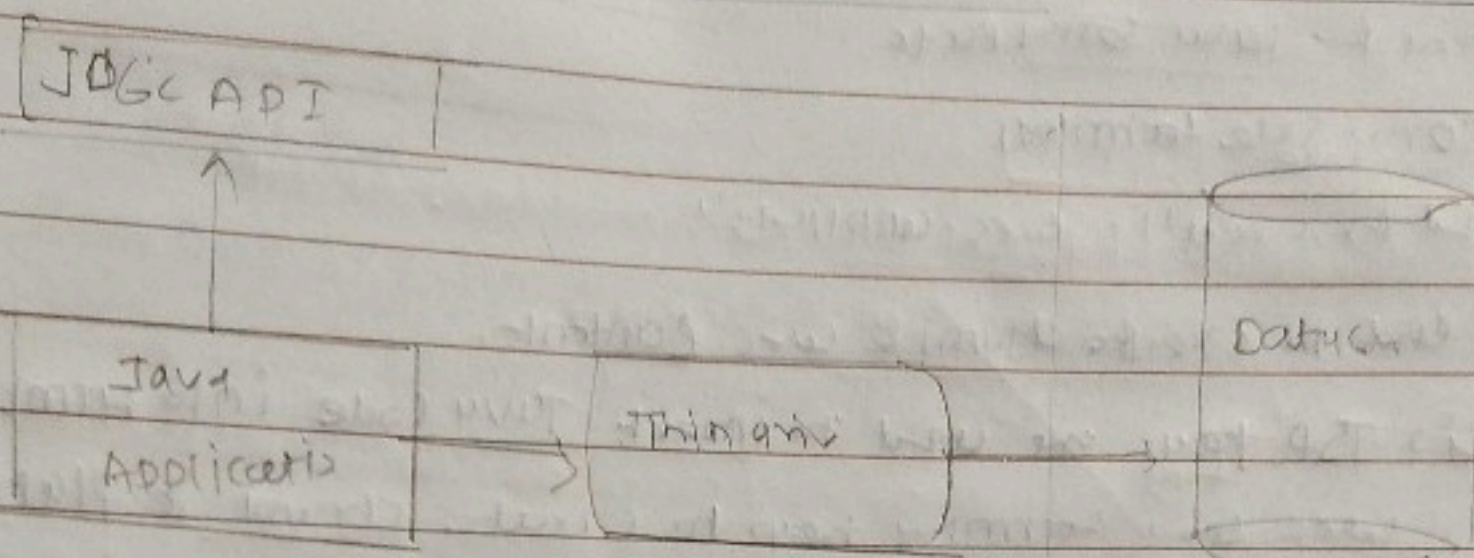
Adv.

- No Client side library is required because of application server that can perform many task like the auditing, load balancing, logging, etc.

Dis.

- Network support is required on client machine.
- Requires database-specific coding to be done in the middle tier.

4) Thin driver :- The thin driver convert JDBC call directly into the vendor-specific database protocol. that is why it known as thin driver. it is fully written in Java language.



Client machine

Thin client

Advantage :-

- ① Better performance than all other drive
- ② No software is required at client side or server side

Disadvantage :-

- ① Drives depend on the Database.

JDBC API

Java

Application

Thin client

Database

Client machine

Thin client

Advantages :-

- ① Better performance than all other driver
- ② No software is required at client side or server side

Disadvantages :-

- ① Drivers depend on the Database.

14) What JSP ? Explain feature of JSP

JSP

- JSP stands for Java Server Page.
- It is Server Side technology.
- It is used for creating web applications.
- It is used to create dynamic web content.
- In this JSP tags are used to insert JAVA code into HTML page.
- It is web based technology helps to create dynamic & platform independent web page.
- In this JAVA code can be inserted in HTML/XML page or code is converted into Servlet by JSP container before proceeding the client request.

• Features :-

- Coding in JSP is easy :- As it is just adding JAVA code to HTML/XML.
- Reduction in the length of code :- In JSP we can use action tags, custom tags, etc.
- Connection to database is easy :- It is easy to connect website to database and allow to read or write data easily to the database.
- Make interactive website :- In this we can create dynamic web page which helps to interact in real time environment.
- Portable, Powerful, Flexible & easy to maintain :- It is dynamic, secure & platform independent so no need for re-compilation.
- Extension to Servlet :- As it has all features of Servlet, implicit objects & custom tags.

Adv

- ① Extension Servlet
- ② easy to maintain
- ③ less code

Advantage of JSP

- it does not require advance knowledge of DP
- it is capable of handling exception
- Easy to the JSP team
- it can trap which an expert can understand

Disadvantage JSP

- Difficult to debug for error
- First time takes time to write code.

Q.5) Life Cycle of JSP page, write with help of example.

JSP

- Translation of JSP page to Servlet
- Compilation of JSP page
- Class loader
- Instantiation
- Initialization
- Request processing
- JSP Cleanup

1) Translation of JSP page to Servlet:

① This is the first step of the JSP life cycle. This translation phase deals with the syntactic correctness of JSP. Here test.jsp file is translated to test.java.

2) Compilation of JSP page:

Here the generated java servlet file (test.java) is compiled to class file (test.class).

3) Classloading:

Servlet class which has been loaded from JSP source is now loaded into the container.

4) Instantiation

Here an instance of the class is generated, the container manages or more instances & providing response to requests.

5) Initialization

JSP init() method is called only once during the life cycle immediately after the generation of servlet instance from JSP.

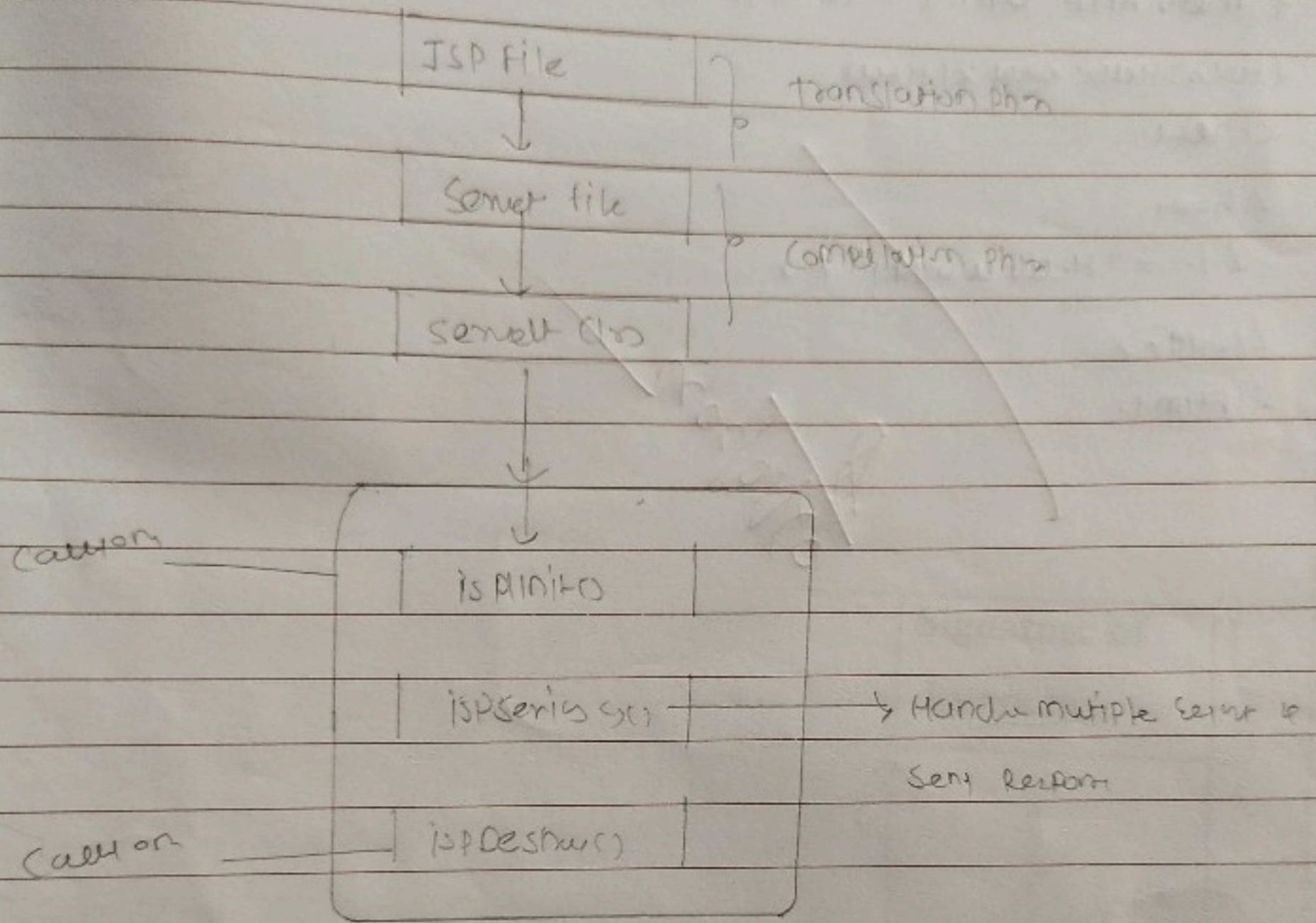
1) Request Processing

- JSP service() method is used to serve the request sent by DP. It takes request and response object as parameter. This method can be overridden.

2) JSP Clean:

- To remove the JSP from the use by the container or destroy the method for Servlet JSP destroy() method can be used.

- It is also used to close open file, releasing DB conn.



JSP life cycle

Ex.

```
<html>
<head>
< meta http-equiv="Content-Type" content="text/html" charset="iso-8859-1">
<title>Hello world</title>
</head>
<body>
<div="Hello world">
</div>
</body>
</html>
```

07/10/2020