SVKM'S
Mithibai College of Arts, Chauhan Institute of Science &
Amrutben Jivanlal College of Commerce and Economics (Autonomous)
Academic Year (2022-23)
Class: SYBSC Semester: IV

Program: B.Sc Computer Science **Max. Marks: 75**
Course Name: Advanced Java **Time:**
Course Code: USMACS402 **Duration: 2 hrs 30 minutes**
Date:

REGULAR EXAMINATION

Instructions: Candidates should read carefully the instructions printed on the question paper and on the cover of the Answer Book, which is provided for their use.
1) This question paper contains 2 pages.
2) Answer to each new question to be started on a fresh page.
3) Figures in brackets on the right-hand side indicate full marks.
4) Assume Suitable data if necessary.

Q1 ATTEMPT ANY 3 FROM THE FOLLOWING: **[21]**

A Write a swing snippet for generating the following. **7**

B What is a Driver? Explain any two types of drivers in JDBC. **7**

C Differentiate between AWT and Swing. **7**

D Describe any 4 methods and any 3 fields used for navigation through database records using a resultset object. **7**

Q2	**ATTEMPT ANY 3 FROM THE FOLLOWING:**	**[21]**
A	Define a servlet? Explain its life cycle methods.	**7**
B	Explain forward and include actions in JSP with an example.	**7**
C	Describe servletconfig interface in detail.	**7**
D	Write a servlet code to show the use of request dispatcher, also write its web.xml.	**7**

Q3	**ATTEMPT ANY 3 FROM THE FOLLOWING:**	**[21]**
A	What is JSON? Differentiate between JSON and XML.	**7**
B	What are interceptors in Struts2? What is the execution flow with respect to interceptor?	**7**
C	Explain the components of struts framework.	**7**
D	Write a program to depict MVC using struts2.(Just print your name with a welcome message).	**7**

Q4	**ATTEMPT ANY 3 FROM THE FOLLOWING:**	**[12]**
A	What is the use of BLOB and CLOB?	**4**
B	State the three directive elements available in JSP.	**4**
C	What is action entity in Struts2 framework?	**4**
D	Write a code snippet to encode JSON object.	**4**

****

N

SVKM'S
Mithibai College of Arts, Chauhan Institute of Science &
Amrutben Jivanlal College of Commerce and Economics (Autonomous)
Academic Year (2022-23)
Class: SYBSC Semester: IV

Program: B.Sc Computer Science **Max. Marks: 75**
Course Name: Advanced Java **Time:**
Course Code: USMACS402 **Duration: 2 hrs 30 minutes**
Date:

SOLUTION SET

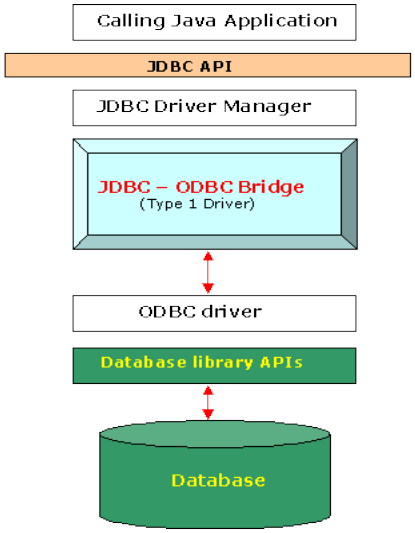
Q1	ATTEMPT ANY 3 FROM THE FOLLOWING:	[21]
A	Write a swing snippet for generating the following. 	7

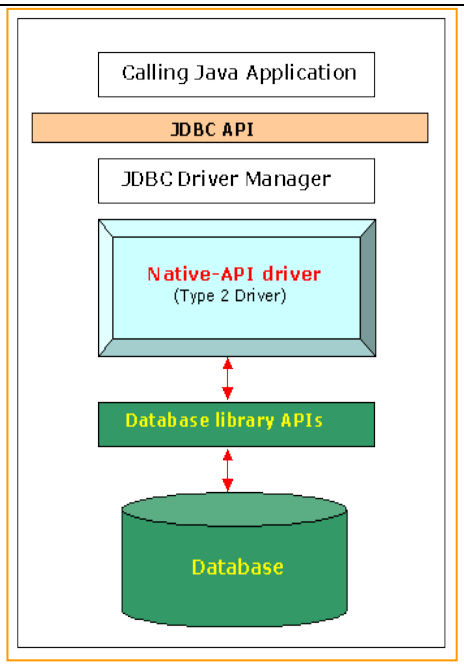
Components texts, labels 5m + buttons 2m

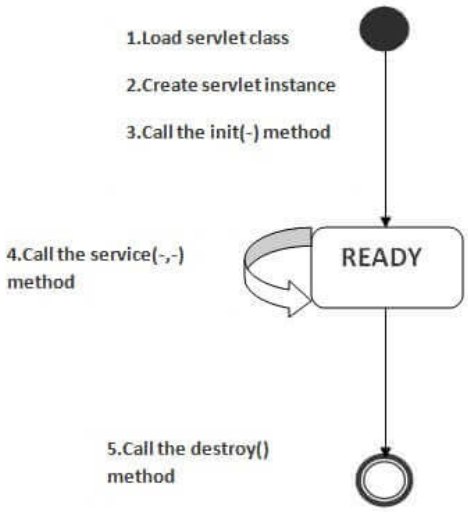
```
public class Registration extends JFrame
{
   JLabel l1, l2, l3, l4, l5, l6, l7, l8;
   JTextField tf1, tf2, tf5, tf6, tf7;
   JButton btn1, btn2;
   JPasswordField p1, p2;
   Registration()
   {
      setVisible(true);
      setSize(700, 700);
      setLayout(null);
      setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
      setTitle("Registration Form in Java");
      l1 = new JLabel("Registration Form in Windows Form:");
      l1.setForeground(Color.blue);
```

```
l1.setFont(new Font("Serif", Font.BOLD, 20));
l2 = new JLabel("Name:");
l3 = new JLabel("Email-ID:");
l4 = new JLabel("Create Passowrd:");
l5 = new JLabel("Confirm Password:");
l6 = new JLabel("Country:");
l7 = new JLabel("State:");
l8 = new JLabel("Phone No:");
tf1 = new JTextField();
tf2 = new JTextField();
p1 = new JPasswordField();
p2 = new JPasswordField();
tf5 = new JTextField();
tf6 = new JTextField();
tf7 = new JTextField();
btn1 = new JButton("Submit");
btn2 = new JButton("Clear");
btn1.addActionListener(this);
btn2.addActionListener(this);
l1.setBounds(100, 30, 400, 30);
l2.setBounds(80, 70, 200, 30);
l3.setBounds(80, 110, 200, 30);
l4.setBounds(80, 150, 200, 30);
l5.setBounds(80, 190, 200, 30);
l6.setBounds(80, 230, 200, 30);
l7.setBounds(80, 270, 200, 30);
l8.setBounds(80, 310, 200, 30);
tf1.setBounds(300, 70, 200, 30);
tf2.setBounds(300, 110, 200, 30);
p1.setBounds(300, 150, 200, 30);
p2.setBounds(300, 190, 200, 30);
tf5.setBounds(300, 230, 200, 30);
tf6.setBounds(300, 270, 200, 30);
tf7.setBounds(300, 310, 200, 30);
btn1.setBounds(50, 350, 100, 30);
btn2.setBounds(170, 350, 100, 30);
add(l1);
add(l2);
add(tf1);
add(l3);
add(tf2);
add(l4);
add(p1);
```

	add(l5);
	add(p2);
	add(l6);
	add(tf5);
	add(l7);
	add(tf6);
	add(l8);
	add(tf7);
	add(btn1);
	add(btn2);
	}	

B	What is a Driver? Explain any two types of drivers in JDBC. Driver 1m+ 6m for 2 drivers A JDBC driver (Java Database Connectivity driver) is a small piece of software that allows JDBC to connect to different databases. Essentially, a JDBC driver makes it possible to do three things: Establish a connection with a data source. Send queries and update statements to the data source. Process the results. • Type I: "Bridge" - • Type II: "Native" - • Type III: "Middleware" - • Type IV: "Pure" Type I Drivers  Type II Drivers	7

<table>
<tr><td></td><td> 
</td><td></td></tr>
<tr><td>C</td><td>Differentiate between AWT and Swing.
7 points 7m
<table><tr><th>S.NO</th><th>AWT</th><th>Swing</th></tr><tr><td>1.</td><td>Java AWT is an API to develop GUI applications in Java</td><td>Swing is a part of Java Foundation Classes and is used to create various applications.</td></tr><tr><td>2.</td><td>The components of Java AWT are heavy weighted.</td><td>The components of Java Swing are light weighted.</td></tr><tr><td>3.</td><td>Java AWT has comparatively less functionality as compared to Swing.</td><td>Java Swing has more functionality as compared to AWT.</td></tr><tr><td>4.</td><td>The execution time of AWT is more than Swing.</td><td>The execution time of Swing is less than AWT.</td></tr><tr><td>5.</td><td>The components of Java AWT are platform dependent.</td><td>The components of Java Swing are platform independent.</td></tr><tr><td>6.</td><td>MVC pattern is not supported by AWT.</td><td>MVC pattern is supported by Swing.</td></tr><tr><td>7.</td><td>AWT provides comparatively less powerful components.</td><td>Swing provides more powerful components.</td></tr></table></td><td>7</td></tr>
<tr><td>D</td><td>Describe any 4 methods and any 3 fields used for navigation through database records using a ResultSet object.
4 methods 4m+ 3 fields 3m
1. boolean first()
2. boolean isFirst()
3. boolean beforeFirst()
4. boolean isbeforeFirst()
1. TYPE_FORWARD_ONLY</td><td>7</td></tr>
</table>

	2. TYPE_SCROLL_SENSITIVE 3. TYPE_SCROLL_INSENSITIVE	
Q2	ATTEMPT ANY 3 FROM THE FOLLOWING:	[21]
A	Define a servlet? Explain its life cycle methods. Servlet 2m +5m for life cycle Servlet technology is used to create a web application (resides at server side and generates a dynamic web page). Servlet technology is robust and scalable because of java language. Before Servlet, CGI (Common Gateway Interface) scripting language was common as a server-side programming language.  	7
B	Explain forward and include actions in JSP with an example. Forward 3.5m + Include 3.5m jsp:forward action tag The jsp:forward action tag is used to forward the request to another resource it may be jsp, html or another resource. Syntax of jsp:forward action tag without parameter <jsp:forward page="relativeURL | <%= expression %>" /> Syntax of jsp:forward action tag with parameter <jsp:forward page="relativeURL | <%= expression %>"> <jsp:param name="parametername" value="parametervalue | <%=expression%>" /> </jsp:forward> The jsp:include action tag is used to include the content of another resource it may be jsp, html or servlet.	7

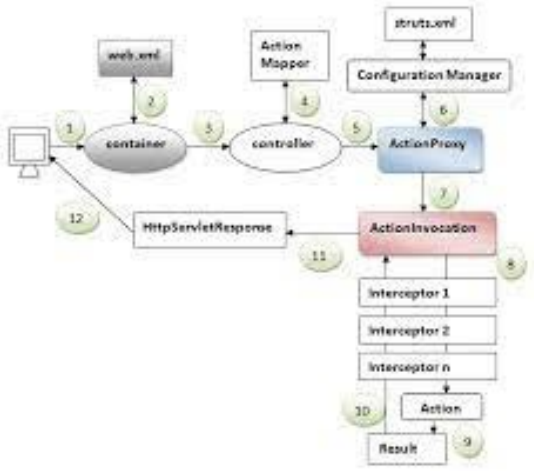
	The jsp include action tag includes the resource at request time so it is better for dynamic pages because there might be changes in future. The jsp:include tag can be used to include static as well as dynamic pages. Advantage of jsp:include action tag Code reusability : We can use a page many times such as including header and footer pages in all pages. So it saves a lot of time. Syntax of jsp:include action tag without parameter <jsp:include page="relativeURL | <%= expression %>" /> Syntax of jsp:include action tag with parameter <jsp:include page="relativeURL | <%= expression %>"> <jsp:param name="parametername" value="parametervalue | <%=expression%>" /> </jsp:include>	
C	Describe servletconfig interface in detail. If the configuration information is modified from the web.xml file, we don't need to change the servlet. So it is easier to manage the web application if any specific content is modified from time to time. Advantage of ServletConfig The core advantage of ServletConfig is that you don't need to edit the servlet file if information is modified from the web.xml file. If the configuration information is modified from the web.xml file, we don't need to change the servlet. So it is easier to manage the web application if any specific content is modified from time to time. Advantage of ServletConfig The core advantage of ServletConfig is that you don't need to edit the servlet file if information is modified from the web.xml file. Methods of ServletConfig interface public String getInitParameter(String name):Returns the parameter value for the specified parameter name. public Enumeration getInitParameterNames():Returns an enumeration of all the initialization parameter names. public String getServletName():Returns the name of the servlet. public ServletContext getServletContext():Returns an object of ServletContext.	7
D	Write a servlet code to show the use of request dispatcher, also write web.xml. String n=request.getParameter("userName"); String p=request.getParameter("userPass"); if(p.equals("servlet"){	7

```
    RequestDispatcher rd=request.getRequestDispatcher("servlet2");
    rd.forward(request, response);
  }
  else{
    out.print("Sorry UserName or Password Error!");
    RequestDispatcher rd=request.getRequestDispatcher("/index.html");
    rd.include(request, response);
                                <servlet-mapping>
                        <servlet-name>Login</servlet-name>
                         <url-pattern>/servlet1</url-pattern>
                                </servlet-mapping>
                                <servlet-mapping>
                      <servlet-name>WelcomeServlet</servlet-name>
                         <url-pattern>/servlet2</url-pattern>
                                </servlet-mapping>
```

Q3	ATTEMPT ANY 3 FROM THE FOLLOWING:	[21]
A	What is JSON? Differentiate between JSON and XML. JSON 2m+ json vs xml 5m JSON stands for JavaScript Object Notation. JSON is a lightweight data-interchange format. JSON is plain text written in JavaScript object notation. JSON is used to send data between computers. JSON is language independent	7

It is JavaScript Object Notation	It is Extensible markup language
It is based on JavaScript language.	It is derived from SGML.
It is a way of representing objects.	It is a markup language and uses tag structure to represent data items.
It does not provides any support for namespaces.	It supports namespaces.
It supports array.	It doesn't supports array.
Its files are very easy to read as compared to XML.	Its documents are comparatively difficult to read and interpret.
It doesn't use end tag.	It has start and end tags.

B	What are interceptors in Struts2? What is the execution flow with respect to interceptor? Interceptors 2m+ the execution flow 5m Interceptor is an object that is invoked at the preprocessing and postprocessing of a request. In Struts 2, interceptor is used to perform operations such as validation, exception handling, internationalization, displaying intermediate result etc.	7

	Advantage of interceptors Pluggable If we need to remove any concern such as validation, exception handling, logging etc. from the application, we don't need to redeploy the application. We only need to remove the entry from the struts.xml file. 	
C	Explain the components of struts framework. Any 5 for 7m • Filter Dispatcher. • Action. • Result. • Configuration file. • Interceptors. • Deployment descriptor. • Tag Library.	7
D	Write a program to depict MVC using struts2. Main snippet with other assumptions 7m 1. <?xml version="1.0" encoding="UTF-8" ?> 2. <!DOCTYPE struts PUBLIC "-//Apache Software Foundation//DTD Struts 3. Configuration 2.1//EN" "http://struts.apache.org/dtds/struts-2.1.dtd"> 4. <struts> 5. <package name="default" extends="struts-default"> 6. 7. <action name="product" class="com.Product"> 8. <result name="success">welcome.jsp</result> 9. </action> 10. 11. </package> 12. </struts>	7

Q4	ATTEMPT ANY 3 FROM THE FOLLOWING:	[12]
A	What is the use of BLOB and CLOB? 2m each one A BLOB is binary large object that can hold a variable amount of data with a maximum length of 65535 characters. These are used to store large amounts of binary data, such as images or other types of files. Fields defined as TEXT also hold large amounts of data. CLOB stands for Character Large Object in general, an SQL Clob is a built-in datatype and is used to store large amount of textual data. Using this datatype, you can store data up to 2,147,483,647 characters. The java.sql.Clob interface of the JDBC API represents the CLOB datatype. Since the Clob object in JDBC is implemented using an SQL locator, it holds a logical pointer to the SQL CLOB (not the data). MYSQL database provides support for this datatype using four variables. • TINYTEXT: A CLOB type with a maximum of 28-1 (255) characters. • TEXT: A CLOB type with a maximum of 216-1 (65535) characters. • MEDIUMTEXT: A CLOB type with a maximum of 224-1 (16777215) characters. • LONGTEXT: A CLOB type with a maximum of 232-1 (4294967295) characters.	4
B	State the three directive elements available in JSP. 3 elements for 4m Page Include Taglib	4
C	What is action entity in Struts2 framework? Struts 2 Action Struts 2 Action Action Interface ActionSupport class In struts 2, action class is POJO (Plain Old Java Object). POJO means you are not forced to implement any interface or extend any class. Generally, execute method should be specified that represents the business logic. The simple action class may look like: Welcome.java package com.mithi;	4

	public class Welcome {
	public String execute(){
	return "success";
	}
	}	

D	Write a code snippet to implement JSON.
	4m for program to encode or decode
	import org.json.simple.JSONObject;
	public class JsonExample1{
	public static void main(String args[]){
	JSONObject obj=new JSONObject();
	obj.put("name","sonuj");
	obj.put("age",new Integer(27));
	obj.put("salary",new Double(600000));
	System.out.print(obj);
	}}
		4

SVKM'S
Mithibai College of Arts, Chauhan Institute of Science &
Amrutben Jivanlal College of Commerce and Economics (Autonomous)
Academic Year (2022-23)
Class: SYBSC Semester: IV

Program: B.Sc Computer Science **Max. Marks: 75**
Course Name: Advanced Java **Time: 10:30 a.m to 1:00 p.m**
Course Code: USMACS402 **Duration: 2 hrs 30 minutes**
Date:

REGULAR EXAMINATION

Instructions: Candidates should read carefully the instructions printed on the question paper and on the cover of the Answer Book, which is provided for their use.
1) This question paper contains 2 pages.
2) Answer to each new question to be started on a fresh page.
3) Figures in brackets on the right-hand side indicate full marks.
4) Assume Suitable data if necessary.

Q1	**ATTEMPT ANY 3 FROM THE FOLLOWING:**	**[21]**
A	Explain any 7 Swing components.	**7**
B	Write a JDBC program to search records from Student table. Student (sap_id, name,date_of_birth,city)	**7**
C	What are Panes in Swings? Why they are required?	**7**
D	What is Statement in JDBC? How it is different from Callable Statement.	**7**

Q2	**ATTEMPT ANY 3 FROM THE FOLLOWING:**	**[21]**
A	What is Deployment descriptor in servlet? Explain its structure.	**7**
B	Write a program in JSP to create a cookie that keeps the count of number of times the page is visited.	**7**
C	What is a JSP Implicit Object? Describe any 5 implicit JSP objects.	**7**
D	What is session Management in servlet? Discuss any one method to manage session in servlets.	**7**

Q3	**ATTEMPT ANY 3 FROM THE FOLLOWING:**	**[21]**
A	Write a code snippet to encode and decode JSON.	**7**

B	What is MVC? Explain the core components of struts2 framework.	**7**
C	Describe struts.xml. Configuration file.	**7**
D	State the properties of JSON. Also compare JSON and XML.	**7**

Q4	**ATTEMPT ANY 3 FROM THE FOLLOWING:**	**[12]**
A	Enlist the 4 types of drivers available in JDBC.	**4**
B	State any 4 functions of web container.	**4**
C	Write a short note on JDBC architecture.	**4**
D	How a struts 2 is different from struts 1?	**4**

****

SVKM'S
Mithibai College of Arts, Chauhan Institute of Science &
Amrutben Jivanlal College of Commerce and Economics (Autonomous)
Academic Year (2022-23)
Class: SYBSC Semester: IV

Program: B.Sc Computer Science Max. Marks: 75
Course Name: Advanced Java Time:
Course Code: USMACS402 Duration: 2 hrs 30 minutes
Date:

SOLUTION SET

Q1	ATTEMPT ANY 3 FROM THE FOLLOWING:	[21]
A	Explain any 7 Swing components. 7 components 7m with explanation and syntax o JButton class o JRadioButton class o JTextArea class o JComboBox class o JTable class o JColorChooser class o JProgressBar class	7

B	Write a JDBC program to search records from Student table.
	Student (sap_id, name,date_of_birth,city)
	7m jdbc program
	// Java Program retrieving contents of
	// Table Using JDBC connection
	
	// Java code producing output which is based
	// on values stored inside the "student" table in DB
	
	// Importing SQL libraries to create database
	import java.sql.*;
		7

```
public class GFG {

    // Step1: Main driver method
    public static void main(String[] args)
    {
        // Step 2: Making connection using
        // Connection type and inbuilt function on
        Connection con = null;
        PreparedStatement p = null;
        ResultSet rs = null;

        con = connection.connectDB();

        // Try block to catch exception/s
        try {

            // SQL command data stored in String datatype
            String sql = "select * from student";
            p = con.prepareStatement(sql);
            rs = p.executeQuery();

            // Printing ID, name, email of customers
            // of the SQL command above
            System.out.println("id\t\tname\t\temail");

            // Condition check
            while (rs.next()) {

                int id = rs.getInt("id");
                String name = rs.getString("name");
                String email = rs.getString("email");
                System.out.println(id + "\t\t" + name
```

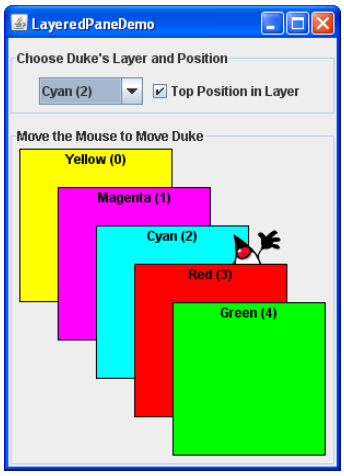
```
                                                    + "\t\t" + email);
                        }
                    }

                    // Catch block to handle exception
                    catch (SQLException e) {

                        // Print exception pop-up on screen
                        System.out.println(e);
                    }
                }
            }
```

C What are Panes in Swings? Why they are required? **7**

Panes 3m + reasons 4m

A layered pane is a Swing container that provides a third dimension for positioning components: *depth*, also known as *Z order*. When adding a component to a layered pane, you specify its depth as an integer. The higher the number, closer the component is to the "top" position within the container. If components overlap, the "closer" components are drawn on top of components at a lower depth. The relationship between components at the same depth is determined by their positions within the depth.

Every Swing container that has a root pane — such as `JFrame`, `JApplet`, `JDialog`, or `JInternalFrame` — automatically has a layered pane. Most programs do not explicitly use the root pane's layered pane, so this section will not discuss it. You can find information about it in The Root Pane, which provides an overview, and The Layered Pane, which has further details. This section tells you how to create your own layered pane and use it anywhere you can use a regular Swing container.

	Swing provides two layered pane classes. The first, `JLayeredPane`, is the class that root panes use and is the class used by the example in this section. The second, `JDesktopPane`, is a `JLayeredPane` subclass that is specialized for the task of holding internal frames. For examples of using `JDesktopPane`, see How to Use Internal Frames. Here is a picture of an application that creates a layered pane and places overlapping, colored labels at different depths: 	
D	What is Statement in JDBC? How it is different from Callable Statement. Statement 2m + differentiate 5 points 5m Statement interface The Statement interface provides methods to execute queries with the database. The statement interface is a factory of ResultSet i.e. it provides factory method to get the object of ResultSet. Commonly used methods of Statement interface: The important methods of Statement interface are as follows: 1) public ResultSet executeQuery(String sql): is used to execute SELECT query. It returns the object of ResultSet. 2) public int executeUpdate(String sql): is used to execute specified query, it may be create, drop, insert, update, delete etc.	**7**

3) public boolean execute(String sql): is used to execute queries that may return multiple results.

4) public int[] executeBatch(): is used to execute batch of commands.

CallableStatement	PreparedStatement
It is used when the stored procedures are to be executed.	It is used when SQL query is to be executed multiple times.
You can pass 3 types of parameter IN, OUT, INOUT.	You can pass any type of parameters at runtime.
Used to execute functions.	Used for the queries which are to be executed multiple times.
Performance is very high.	Performance is better than Statement.
Used to call the stored procedures.	Used to execute dynamic SQL queries.
It extends PreparedStatement interface.	It extends Statement Interface.
No protocol is used for communication.	Protocol is used for communication.

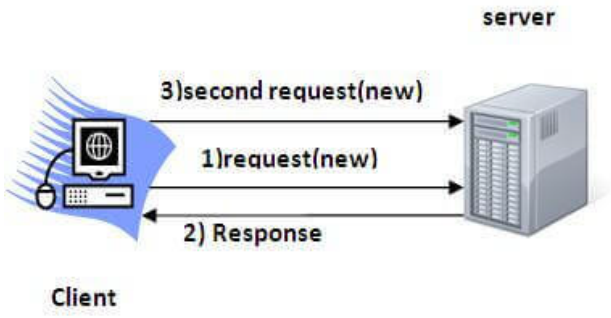
Q2	**ATTEMPT ANY 3 FROM THE FOLLOWING:**	**[21]**
A	What is Deployment descriptor in servlet? Explain its structure. Deployment descriptor 2m +5m structure The **deployment descriptor** is an xml file, from which Web Container gets the information about the servet to be invoked. The web container uses the Parser to get the information from the web.xml file. There are many xml parsers such as SAX, DOM and Pull. There are many elements in the web.xml file. Here is given some necessary elements to run the simple servlet program. <web-app> represents the whole application. <servlet> is sub element of <web-app> and represents the servlet. <servlet-name> is sub element of <servlet> represents the name of the servlet. <servlet-class> is sub element of <servlet> represents the class of the servlet. <servlet-mapping> is sub element of <web-app>. It is used to map the servlet.	**7**

	<url-pattern> is sub element of <servlet-mapping>. This pattern is used at client side to invoke the servlet.	

B	Write a program in JSP to create a cookie to count the number of times the page is visited.
	7m for all steps
	<form>
	<fieldset style="width:20%; background-color:#e6ffe6;">
	<legend>Count visitor</legend>
	<%
	Integer hitsCount =
	(Integer)application.getAttribute("hitCounter");
	if(hitsCount ==null || hitsCount == 0)
	{
	/* First visit */
	out.println("Welcome to my website!!");
	hitsCount = 1;
	}
	else
	{
	/* return visit */
	out.println("Welcome to my website!!");
	hitsCount += 1;
	}
	application.setAttribute("hitCounter", hitsCount);
	%>
	 You are visitor number: <%= hitsCount%>
	</fieldset>
	</form>
	</body>
	</html>
		7

C	What is a JSP Implicit Object? Describe any 5 implicit JSP objects. 2m implicit object+ 5 object 5m syntax	**7**

	These objects are *created by the web container* that are available to all the jsp pages. ObjectType out JspWriter request HttpServletRequest response HttpServletResponse config ServletConfig application ServletContext session HttpSession pageContext PageContext page Object exception Throwable	
D	What is session Management in servlet? Discuss any one method to manage session in servlets. Session management 3m 1 method 4m **Session** simply means a particular interval of time. **Session Tracking** is a way to maintain state (data) of an user. It is also known as **session management** in servlet. Http protocol is a stateless so we need to maintain state using session tracking techniques. Each time user requests to the server, server treats the request as the new request. So we need to maintain the state of an user to recognize to particular user. HTTP is stateless that means each request is considered as the new request. It is shown in the figure given below:	**7**

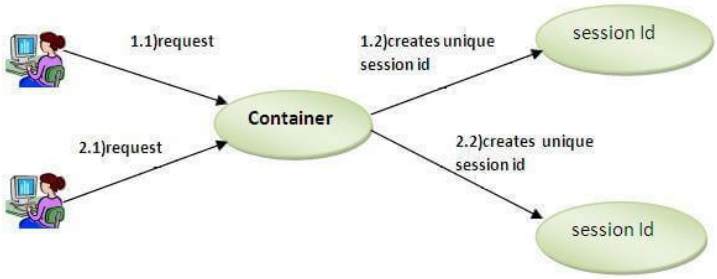


Why use Session Tracking?

To recognize the user It is used to recognize the particular user.

In such case, container creates a session id for each user.The container uses this id to identify the particular user.An object of HttpSession can be used to perform two tasks:

1. bind objects
2. view and manipulate information about a session, such as the session identifier, creation time, and last accessed time.



How to get the HttpSession object ?

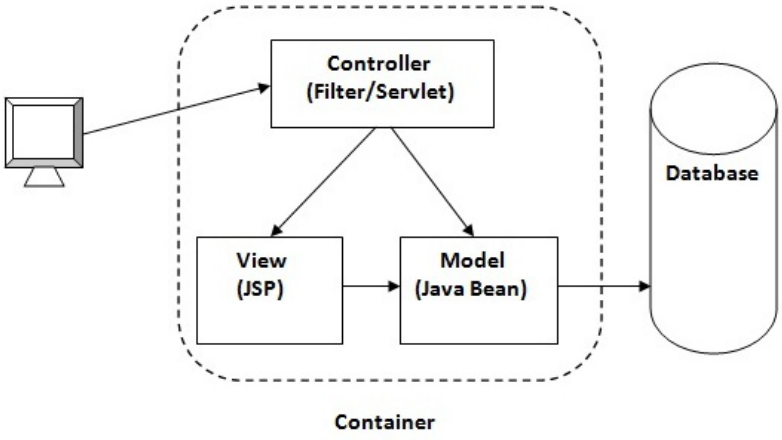
The HttpServletRequest interface provides two methods to get the object of HttpSession:

1. **public HttpSession getSession():**Returns the current session associated with this request, or if the request does not have a session, creates one.

	2. **public HttpSession getSession(boolean create):**Returns the current HttpSession associated with this request or, if there is no current session and create is true, returns a new session.	
Q3	**ATTEMPT ANY 3 FROM THE FOLLOWING:**	**[21]**

A	Write a code snippet to encode and decode JSON.
	Encode 3.5 +decode3.5m
	1. **import** org.json.simple.JSONObject;
	2. **public class** JsonExample1{
	3. **public static void** main(String args[]){
	4. JSONObject obj=**new** JSONObject();
	5. obj.put("name","rohit");
	6. obj.put("age",**new** Integer(27));
	7. obj.put("salary",**new** Double(600000));
	8. System.out.print(obj);
	9. }}
	
	1. **import** java.util.HashMap;
	2. **import** java.util.Map;
	3. **import** org.json.simple.JSONValue;
	4. **public class** JsonExample2{
	5. **public static void** main(String args[]){
	6. Map obj=**new** HashMap();
	7. obj.put("name","rohit");
	8. obj.put("age",**new** Integer(27));
	9. obj.put("salary",**new** Double(600000));
	10. String jsonText = JSONValue.toJSONString(obj);
	11. System.out.print(jsonText);
	12. }}
		7

B	What is MVC? Explain the core components of struts2 framework. MVC 2m+ components of struts2 5m	**7**

	MVC (Model View Controller) design pattern. The MVC design pattern consists of three modules model, view and controller. **Model** The model represents the state (data) and business logic of the application. **View** The view module is responsible to display data i.e. it represents the presentation. **Controller** The controller module acts as an interface between view and model. It intercepts all the requests i.e. receives input and commands to Model / View to change accordingly.  from a high level, Struts2 is a pull-MVC (or MVC2) framework. The Model-ViewController pattern in Struts2 is implemented with the following five core components – • Actions • Interceptors • Value Stack / OGNL • Results / Result types • View technologies	
C	Describe struts.xml. Configuration file. Explanation + structure 7m # Struts 2 Configuration File 1. Struts 2 Configuration File	**7**

2. Elements of struts.xml file

The struts application contains two main configuration files **struts.xml** file and **struts.properties** file.

The struts.properties file is used to override the default values of **default.xml** file provided by struts framework. So it is not mandatory. Mostly, you will not use struts.properties file. We will learn about it later.

Here, we are going to learn all about struts.xml file. First of all let us see the simple example of struts.xml file

struts.xml

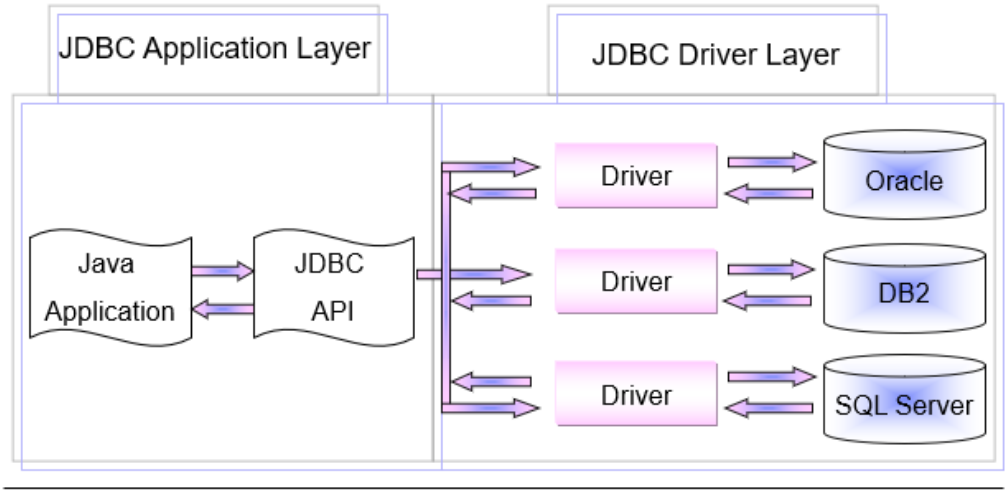
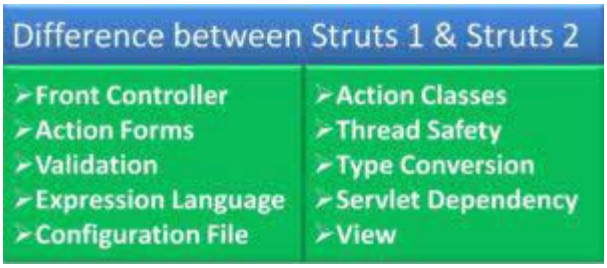
```
1. <?xml version="1.0" encoding="UTF-8" ?>
2. <!DOCTYPE struts PUBLIC "-
   //Apache Software Foundation//DTD Struts
3. Configuration 2.1//EN" "http://struts.apache.org/dtds/struts-2.1.dtd">
4. <struts>
5. <package name="default" extends="struts-default">
6.
7. <action name="product" class="com.javatpoint.Product">
8. <result name="success">welcome.jsp</result>
9. </action>
10.
11. </package>
12. </struts>
```

D State the properties of JSON. Also compare JSON and XML. **7**

Features of JSON

- Simplicity
- Openness
- Self-Describing
- Internationalization
- Extensibility
- Interoperability

	<table><tr><th>JSON</th><th>XML</th></tr><tr><td>JSON stands for javascript object notation.</td><td>XML stands for an extensible markup language.</td></tr><tr><td>The extension of json file is .json.</td><td>The extension of xml file is .xml.</td></tr><tr><td>The internet media type is application/json.</td><td>The internet media type is application/xml or text/xml.</td></tr><tr><td>The type of format in JSON is data interchange.</td><td>The type of format in XML is a markup language.</td></tr><tr><td>It is extended from javascript.</td><td>It is extended from SGML.</td></tr><tr><td>It is open source means that we do not have to pay anything to use JSON.</td><td>It is also open source.</td></tr><tr><td>The object created in JSON has some type.</td><td>XML data does not have any type.</td></tr></table>	
Q4	**ATTEMPT ANY 3 FROM THE FOLLOWING:**	**[12]**
A	Enlist 4 types of drivers available in JDBC. 4 drivers 4m JDBC-ODBC Bridge Driver Native-API Partly-Java Driver JDBC-Net Pure-Java Driver Native Protocol Pure-Java Driver	**4**
B	State any 4 functions of web container. Any 4 functions 4m • Communication Support • Lifecycle Management • Multi-threading support • Security • JSP Support	**4**
C	Write a short note on JDBC architecture. 4m	**4**

	▪ It can be categorized into into two layers:  	
D	How a struts 2 is different from struts 1? 4 points 4m  	**4**

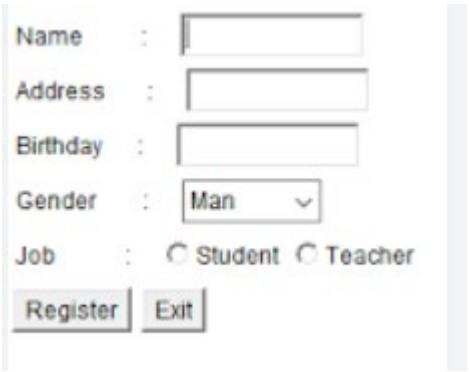
SVKM'S
Mithibai College of Arts, Chauhan Institute of Science &
Amrutben Jivanlal College of Commerce and Economics (Autonomous)
Academic Year (2022-23)
Class: SYBSC Semester: IV

Program: B.Sc Computer Science **Max. Marks: 75**
Course Name: Advanced Java **Time:**
Course Code: USMACS402 **Duration: 2 hrs 30 minutes**
Date:

REGULAR EXAMINATION

Instructions: Candidates should read carefully the instructions printed on the question paper and on the cover of the Answer Book, which is provided for their use.
1) This question paper contains 2 pages.
2) Answer to each new question to be started on a fresh page.
3) Figures in brackets on the right-hand side indicate full marks.
4) Assume Suitable data if necessary.

Q1	**ATTEMPT ANY 3 FROM THE FOLLOWING:**	**[21]**
A	What is Swing? Discuss the features of Swing in detail.	**7**
B	Write a Swing Code to design the following:	**7**



C	Write a note on BLOB and CLOB with an example.	**7**
D	State and explain JDBC steps for connecting a java program to a database.	**7**

Q2	**ATTEMPT ANY 3 FROM THE FOLLOWING:**	**[21]**
A	Develop servlet application of basic calculator (+, -,*, /) using HttpServletRequest and HttpServletResponse.	**7**
B	Describe Servletcontext interface with an example.	**7**

C	Describe JSP Life cycle.	**7**
D	What are JSP Actions? How they are different from directives?	**7**

Q3 ATTEMPT ANY 3 FROM THE FOLLOWING: [21]

A	Explain why to use JSON and its applications. Also write a program in JSON to encode an employee class.	**7**
B	Explain Struts 2 Architecture in detail.	**7**
C	Discuss the different data types in JSON with an example.	**7**
D	What is Value Stack in Struts2? What is the execution flow with respect to Value Stack?	**7**

Q4 ATTEMPT ANY 3 FROM THE FOLLOWING: [12]

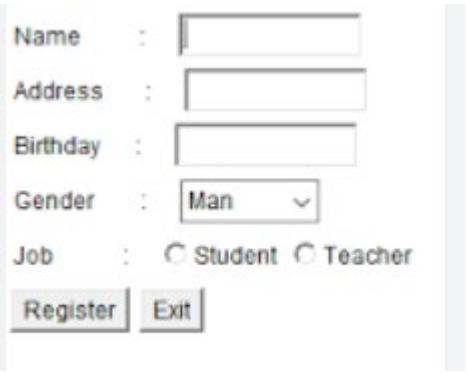
A	MVC is the important feature of SWING over AWT. Justify.	**4**
B	State the 4 session management schemes available in servlets.	**4**
C	What are Scrollable and non-Scrollable ResultSets?	**4**
D	Write a note on OGNL in Struts2.	**4**

****

SVKM'S
Mithibai College of Arts, Chauhan Institute of Science &
Amrutben Jivanlal College of Commerce and Economics (Autonomous)
Academic Year (2022-23)
Class: SYBSC Semester: IV

Program: B.Sc Computer Science **Max. Marks: 75**
Course Name: Advanced Java **Time:**
Course Code: USMACS402 **Duration: 2 hrs 30 minutes**
Date:

SOLUTION SET

Q1	ATTEMPT ANY 3 FROM THE FOLLOWING:	[21]
A	What is Swing? Discuss the features of Swing in detail. Swing 2m +5 exploring features of Swing 5m Java Swing provides platform-independent and lightweight components. The javax.swing package provides classes for java swing API such as JButton, JTextField, JTextArea, JRadioButton, JCheckbox, JMenu, JColorChooser etc. Java swing components are platform-independent. Swing components are lightweight. Swing supports pluggable look and feel. Swing follows MVC	7
B	Write a Swing Code to design the following:  public void init() {	7

```
        setLayout(new FlowLayout(FlowLayout.LEFT));

        add(new Label("Name         :"));
        add(new TextField(10));

        add(new Label("Address      :"));
        add(new TextField(10));

        add(new Label("Birthday     :"));
        add(new TextField(10));


        add(new Label("Gender       :"));
        Choice gender = new Choice();
        gender.addItem("Man");
        gender.addItem("Woman");
        Component add = add(gender);

        add(new Label("Job          :"));
        CheckboxGroup job = new CheckboxGroup();
        add(new Checkbox("Student", job, false));
        add(new Checkbox("Teacher", job, false));

        add(new Button("Register"));
        add(new Button("Exit"));
        }
```

C | Write a note on BLOB and CLOB. | 7

BLOB 3.5m and CLOB 3.5m.

A BLOB is binary large object that can hold a variable amount of data with a maximum length of 65535 characters.

These are used to store large amounts of binary data, such as images or other types of files. Fields defined as TEXT also hold large amounts of data. The difference between the two is that the sorts and comparisons on the stored data are case sensitive on BLOBs and are not case sensitive in TEXT fields. You do not specify a length with BLOB or TEXT.

Storing blob in to database

To store Blob datatype to database, using JDBC program follow the steps given below

Step 1: Connect to the database

You can connect to a database using the getConnection() method of the DriverManager class.

Connect to the MySQL database by passing the MySQL URL which is jdbc:mysql://localhost/sampleDB (where sampleDB is the database name), username and password as parameters to the getConnection() method.

```
String mysqlUrl = "jdbc:mysql://localhost/sampleDB";
Connection con = DriverManager.getConnection(mysqlUrl, "root", "password");
```

Step 2: Create a Prepared statement

Create a PreparedStatement object using the prepareStatement() method of the Connection interface. To this method pass the insert query (with place holders) as a parameter.

```
PreparedStatement pstmt = con.prepareStatement("INSERT INTO MyTableVALUES(?, ?)");
```

Step 3: Set values to the place holders

Set the values to the place holders using the setter methods of the PreparedStatement interface. Chose the methods according to the datatype of the column. For Example if the column is of VARCHAR type use setString() method and if it is of INT type you can use setInt() method.

And if it is of Blob type you can set value to it using the setBinaryStream() or setBlob() methods. To these methods pass an integer variable representing the parameter index and an object of InputStream class as parameters.

```
pstmt.setString(1, "sample image");
//Inserting Blob type
InputStream in = new FileInputStream("E:\images\cat.jpg");
pstmt.setBlob(2, in);
```

Step 4: Execute the statement

Execute the above created PreparedStatement object using the execute() method of the PreparedStatement interface.

Retrieving blob from database

The getBlob() method of the ResultSet interface accepts an integer representing the index of the column (or, a String value representing the name of the column) and retrieves the value at the specified column and returns it in the form of a Blob object.

```
while(rs.next()) {
   rs.getString("Name");
   rs.getString("Type");
   Blob blob = rs.getBlob("Logo");
```

}

The getBytes() method of the Blob Interface retrieves the contents of the current Blob object and returns as a byte array.

Using the getBlob() method you can get the contents of the blob in to a byte array and create an image using the write() method of the FileOutputStream object.

```
byte byteArray[] = blob.getBytes(1,(int)blob.length());
FileOutputStream outPutStream = new FileOutputStream("path");
outPutStream.write(byteArray);
```

CLOB stands for Character Large Object in general, an SQL Clob is a built-in datatype and is used to store large amount of textual data. Using this datatype, you can store data up to 2,147,483,647 characters.

The java.sql.Clob interface of the JDBC API represents the CLOB datatype. Since the Clob object in JDBC is implemented using an SQL locator, it holds a logical pointer to the SQL CLOB (not the data).

MYSQL database provides support for this datatype using four variables.

TINYTEXT: A CLOB type with a maximum of 28-1 (255) characters.

TEXT: A CLOB type with a maximum of 216-1 (65535) characters.

MEDIUMTEXT: A CLOB type with a maximum of 224-1 (16777215) characters.

LONGTEXT: A CLOB type with a maximum of 232-1 (4294967295) characters.

Storing Clob datatype in to table in a database
To store Clob datatype to database, using JDBC program follow the steps given below

Step 1: Connect to the database
You can connect to a database using the getConnection() method of the DriverManager class.

Connect to the MySQL database by passing the MySQL URL which is jdbc:mysql://localhost/sampleDB (where sampleDB is the database name), username and password as parameters to the getConnection() method.

```
String mysqlUrl = "jdbc:mysql://localhost/sampleDB";
Connection con = DriverManager.getConnection(mysqlUrl, "root", "password");
```

Step 2: Create a Prepared statement

Create a PreparedStatement object using the prepareStatement() method of the Connection interface. To this method pass the insert query (with place holders) as a parameter.

```
PreparedStatement pstmt = con.prepareStatement("INSERT INTO Technologies(Name,
Type, Article ) VALUES (?, ?, ?)");
```

Step 3: Set values to the place holders

Set the values to the place holders using the setter methods of the PreparedStatement interface. Chose the methods according to the datatype of the column. For Example if the column is of VARCHAR type use setString() method and if it is of INT type you can use setInt() method.

And if it is of Clob type you can set value to it using the setCharacterStream() or setClob() methods. To these methods pass an integer variable representing the parameter index and an object of the Reader class as parameters.

```
pstmt.setString(1, "JavaFX");
pstmt.setString(2, "Java Library");
FileReader reader = new FileReader("E:\images\javafx.txt");
pstmt.setClob(3, reader);
pstmt.execute();
```

Step 4: Execute the statement

Execute the above created PreparedStatement object using the execute() method of the PreparedStatement interface.

Retrieving blob from database

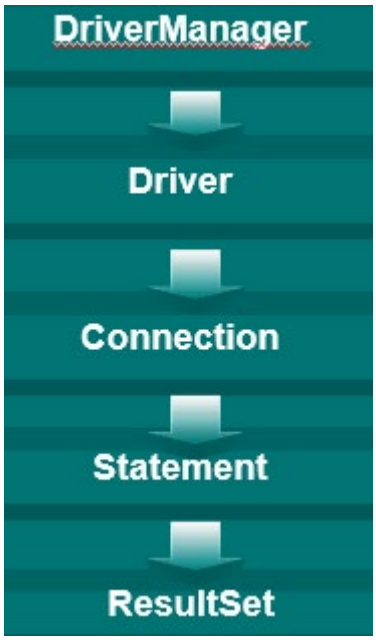
The getClob() method of the ResultSet interface accepts an integer representing the index of the column (or, a String value representing the name of the column) and retrieves the value at the specified column and returns it in the form of a Clob object.

```
while(rs.next()) {
  System.out.println(rs.getString("Name"));
  System.out.println(rs.getString("Type"));
  Clob clob = rs.getClob("Article");
}
```

The getCharacterStream() method of the Clob Interface retrieves the contents of the current Clob object and returns as a Reader object.

Using the getClob() method you can get the contents of the Clob as a Reader object and create text file with the retrieved contents, using the write() method of the FileOutputStream object.

	Reader r = clob.getCharacterStream();
	char cbuf[] = new char[r.read()];
	r.read(cbuf);
	FileOutputStream outPutStream = new
	FileOutputStream("E:\images\clob_output"+i+".txt");
	outPutStream.write(cbuf.toString().getBytes());	

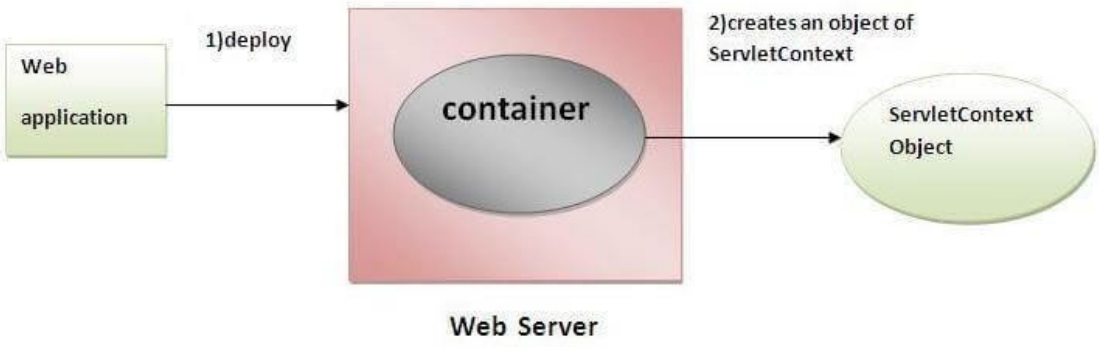
D	State and explain JDBC steps for connecting a java program to a database. JDBC steps with all classes and interface 7M  	7
Q2	ATTEMPT ANY 3 FROM THE FOLLOWING:	[21]

A	Develop servlet application of basic calculator (+,-, *, /) using HttpServletRequest and HttpServletResponse.
	public void doGet(HttpServletRequest request, HttpServletResponse response)
	throws ServletException, IOException {
	response.setContentType("text/html;charset=UTF-8);
	PrintWriter out = response.getWriter();
	out.println("<html><head><title>Servlet CalculatorServlet</title></head><body>"); double n1 = Double.parseDouble(request.getParameter("txtN1"));
		7

```
double n2 = Double.parseDouble(request.getParameter("txtN2"));
double result =0;
String opr=request.getParameter("opr");
if(opr.equals("+")) result=n1+n2;
if(opr.equals("-")) result=n1-n2;
if(opr.equals("*")) result=n1*n2;
if(opr.equals("/")) result=n1/n2;

out.println("<h1> Result = "+result);
 out.println("</body></html>");
}

}
```

B	Describe Servletcontext interface with an example. Servletcontext 7M (explanation,methids,syntax and example) An object of ServletContext is created by the web container at time of deploying the project. This object can be used to get configuration information from web.xml file. There is only one ServletContext object per web application. If any information is shared to many servlet, it is better to provide it from the web.xml file using the <context-param> element. Advantage of ServletContext Easy to maintain if any information is shared to all the servlet, it is better to make it available for all the servlet. We provide this information from the web.xml file, so if the information is changed, we don't need to modify the servlet. Thus it removes maintenance problem. Usage of ServletContext Interface There can be a lot of usage of ServletContext object. Some of them are as follows: 1. The object of ServletContext provides an interface between the container and servlet. 2. The ServletContext object can be used to get configuration information from the web.xml file.	7

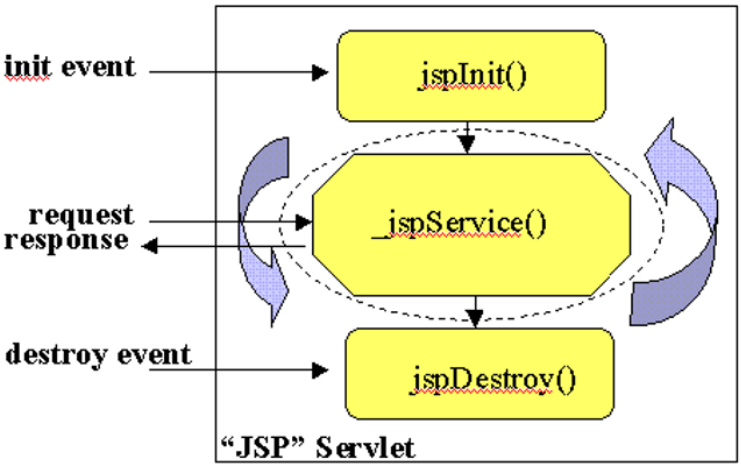
3. The ServletContext object can be used to set, get or remove attribute from the web.xml file.
4. The ServletContext object can be used to provide inter-application communication.



Commonly used methods of ServletContext interface

There is given some commonly used methods of ServletContext interface.

1. public String getInitParameter(String name):Returns the parameter value parameter name.
2. public Enumeration getInitParameterNames():Returns the names of the cont parameters.
3. public void setAttribute(String name,Object object):sets the given object in the
4. public Object getAttribute(String name):Returns the attribute for the specified
5. public Enumeration getInitParameterNames():Returns the names of the cont parameters as an Enumeration of String objects.
6. public void removeAttribute(String name):Removes the attribute with the giv servlet context.
7. public class DemoServlet extends HttpServlet{
8. public void doGet(HttpServletRequest req,HttpServletResponse res)
9. throws ServletException,IOException
10. {
11. res.setContentType("text/html");
12. PrintWriter pw=res.getWriter();
13.
14. //creating ServletContext object

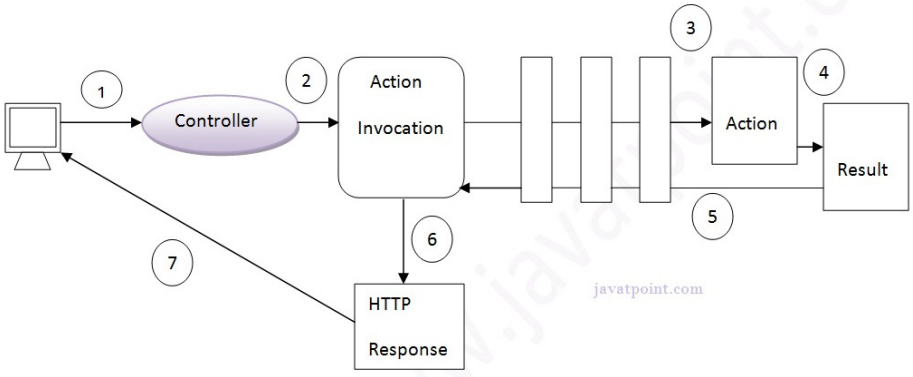
	15. ServletContext context=getServletContext(); 16. 17. //Getting the value of the initialization parameter and printing it 18. String driverName=context.getInitParameter("dname"); 19. pw.println("driver name is="+driverName); 20. 21. pw.close(); 22. 23. }}	
C	Describe JSP Life cycle. 3 methods+diagram 7m  	7
D	What are JSP Actions? How they are different from directives? Action explanation + enlisting actions 3m +difference 4 point 4m The action tags are used to control the flow between pages and to use Java Bean.	7

JSP Action Tags	Description
jsp:forward	forwards the request and response to another resource.
jsp:include	includes another resource.
jsp:useBean	creates or locates bean object.
jsp:setProperty	sets the value of property in bean object.
jsp:getProperty	prints the value of property of the bean.
jsp:plugin	embeds another components such as applet.
jsp:param	sets the parameter value. It is used in forward and include mostly.
jsp:fallback	can be used to print the message if plugin is working. It is used in jsp:plugin.

Include Directive	Include Action
Translation time	Run time
Copies the included file	References to the included file
For static content	For dynamic content
Cannot pass parameters	Can pass parameters

Q3	ATTEMPT ANY 3 FROM THE FOLLOWING:	[21]
A	Explain why to use JSON and its applications. Also write a program in JSON to encode a class. use JSON and its applications 3M + program 4M JSON or JavaScript Object Notation is a lightweight, text-based, data-interchange format that follows JavaScript object syntax. JSON is used for data transportation and restoration in places of XML structures. The JSON format is often used when data is sent from a server to a web page or a browser for rendering. The most desired and popular applications of JSON are listed below:	7

	1. It can be used with various modern programming languages such as Python, Ruby, Java, etc.
	2. It is used for data transmission between a server and web applications
	3. JSON is used in JS-based applications, for example- web browser extensions and websites.
	4. APIs and web services use the JSON format for providing public data.
	5. JSON data format simplifies complex data by converting the data extraction process into a meaningful and predictable JSON file format.
	6. import org.json.simple.JSONObject;
	7. public class JsonExample1{
	8. public static void main(String args[]){
	9. JSONObject obj=new JSONObject();
	10. obj.put("name","aron");
	11. obj.put("age",new Integer(27));
	12. obj.put("salary",new Double(600000));
	13. System.out.print(obj);
	14. }}	

B	Explain Struts 2 Architecture in detail. Diagram 3M+ explanation 4M 	7
C	Discuss the different data types in JSON with an example. 5 data types 5m+example 2m o JSONValue o JSONObject o JSONArray o JsonString o JsonNumber	7
D	What is Value Stack in Struts2? What is the execution flow with respect to Value Stack?	7

A valueStack is simply a stack that contains application specific objects such as action objects and other model object.

At the execution time, action is placed on the top of the stack.

We can put objects in the valuestack, query it and delete it.

he value stack can be accessed via the tags provided for JSP, Velocity or Freemarker. There are various tags which we will study in separate chapters, are used to get and set struts 2.0 value stack. You can get valueStack object inside your action as follows −

ActionContext.getContext().getValueStack()

Once you have a ValueStack object, you can use the following methods to manipulate that object −

Sr.No	ValueStack Methods & Description
1	Object findValue(String expr) Find a value by evaluating the given expression against the stack in the default search order.
2	CompoundRoot getRoot() Get the CompoundRoot which holds the objects pushed onto the stack.
3	Object peek() Get the object on the top of the stack without changing the stack.
4	Object pop()

	Get the object on the top of the stack and remove it from the stack. 5 void push(Object o) Put this object onto the top of the stack.	
Q4	ATTEMPT ANY 3 FROM THE FOLLOWING:	[12]
A	MVC is the important feature of SWING over AWT. Justify. Swing uses fewer system resources, adds lots more sophisticated components, helps to tailor the look & feel of the programs. swing development has its root in MVC(Model – View- Controller) architecture. MVC allows swing components to be replaced with different data models & views. Plug gable look & feel is result of MVC architecture. Since java is platform independent & runs on client machine , the look & feel of any platform has to be known Lightweight components:- in swing most of the components have their own view supported by java look & feel classes (it can't rely on native system classes) Pluggable look & feel:- supports cross platform look & feel also called java look & feel that remains same across all platforms wherever the program runs.	4
B	State the 4 session management schemes available in servlets. 1. Cookies 2. HttpSession 3. URL Rewriting 4. Hidden form field	4
C	What is Scrollable and non-Scrollable ResultSet? A scrollable updatable result set maintains a cursor which can both scroll and update rows. Derby only supports scrollable insensitive result sets. To create a scrollable insensitive result set which is updatable, the statement has to be created with concurrency mode ResultSet.	4

	Non-scrollable resultset can only move in forward direction from first to last element and also they cannot move directly to any row in the database	
D	Write a note on OGNL in Struts2. The Object Graph Navigation Language (OGNL) is an expression language. It simplifies the accessibility of data stored in the ActionContext. OGNL is based on a context and Struts builds an ActionContext map for use with OGNL. The ActionContext map consists of the following − • Application − Application scoped variables • Session − Session scoped variables • Root / value stack − All your action variables are stored here • Request − Request scoped variables • Parameters − Request parameters • Atributes − The attributes stored in page, request, session and application scope	4

****