Course Name: JAVA LABORATORY			
Course code:	22ISL45	**Credits:**	1
L: T:P:	0:0:2	**CIE Marks:**	50
Exam Hours:	03	**SEE Marks:**	50
Total Hours:	03		

Course Objectives:	
1.	To provide an introduction to java and object oriented concepts
2.	Understand the servlet programming.
3.	Analyze and use exception handling in java.
4.	A better understanding of Spring Framework in java.

Course Outcomes: At the end of the course, student will be able to:

CO1 Articulate classes, its members and the relationships among them needed for a specific problem.

CO2 Apply the basic concepts of packages and implementing java programs.

CO3 Create and use, exceptionhandling,Networking and JDBC in Java programs.

CO4 Design and develop web based application usingservlets, JSP and Spring Framework

Mapping of Course outcomes to Program outcomes:														
	PO1	**PO2**	**PO3**	**PO4**	**PO5**	**PO6**	**PO7**	**PO8**	**PO9**	**PO10**	**PO11**	**PO12**	**PSO1**	**PSO2**	**PSO3**
CO1	2	2	3	-	-	-	-	-	-	-	-	1	2	1	-
CO2	3	2	3	-	1	-	-	-	-	-	-	2	3	2	-
CO3	3	2	3	-	2	-	-	1	-	-	-	2	3	1	-
CO4	2	2	3	-	2	-	-	1	-	-	-	2	3	1	1

Expt. No	**Contents of the experiment**	**Hours**	**CO's**

1.	Create a class Vehicle. The class should have two fields - no_of_seats, no_of_wheels and a method showVehicle. Create two objects-Motorcycle and Car for this class. Display the output to show the descriptions for Car and Motorcycle.	2	CO 1
2.	Construct a program to create a package called Balance which has an Account class with display_balance method in it. Import Balance package in another program to access display_Balance method of Account class.	2	CO 2
3.	Design a super class called Employee with details as EmployeeId, name, Phone, Salary. Extend this class by writing three subclasses namely Teaching (domain, publications), Technical (skills), and Contract (period). Write a JAVA program to read and displayat least 3 Employee objects of all three categories.	2	CO1 ,CO 2
4.	Implement a JAVA program to read two integers a and b. Compute a/b and print, when b is not zero. Raise an exception when b is equal to zero. Also demonstrate working of ArrayIndexOutOfBoundException.	2	CO1 ,CO 3
5.	Implement Java Program to Get the Components of any give URL such as Protocol, file, port and host.	2	CO3
6.	Implement client-server communication, where client can send the message and server can receive the message without internet.	2	CO3
7.	Implement JDBC program to insert and retrieve student (student_name,student_usn, student_dept) record from student database.	2	CO3
8.	Create web page authentication using JSP and JDBC connectivity (login authentication) using session.	2	CO3 ,4
9.	Structure the java servlet program to fetch the student details using JDBC.	2	CO4
10.	Develop a Servlet program to demonstrate hit counter.	2	CO4

TEXT BOOKS:

TB No.	Author / Edition/ Publication / Year	Chapters
1.	Herbert Schildt: Java The Complete Reference, 7th Edition, Tata McGraw Hill, 2007	2, 3, 4, 5, 6,7, 8, 9,10, 12,13,15
2.	Craig Walls, "Spring in Action", 5 th Edition,Manning Publications, 2018	1,2

REFERENCE BOOKS:

RB No.	Author / Edition/ Publication / Year	Chapters
1.	Y. Daniel Liang: Introduction to JAVA Programming, 6th Ed, Pearson Education, 2007.	13,14,16

1. Create a class Vehicle. The class should have two fields - no_of_seats, no_of_wheels and a method showVehicle. Create two objects-Motorcycle and Car for this class. Display the output to show the descriptions for Car and Motorcycle.

Abstract Class: Abstraction is the mechanism of exhibiting the necessary things by hiding the unnecessary things.
An abstract class is a class that may have at least one abstract method.(ie without body)

we cannot create an object for abstract class. Abstract class may have reference variables but may not have memory for it.

/* Write a Java program to implement an Vehicle Abstract class.*/

```
abstract class Vehicle
{
public abstract void wheels();
public abstract void seating();
public abstract void brakes();
 }
class Car extends Vehicle
 {
public void wheels()
{
 System.out.println("\nCar Has Four Wheels");
 }
 public void seating()
 {
 System.out.println("Car Has Four Seating Capacity");
}
public void brakes()
{
System.out.println("Car Has Power Brakes\n");
 } }
class Bike extends Vehicle
{
 public void wheels()
{
System.out.println("Bike Has Two Wheels");
 }
public void seating()
 {
System.out.println("Bike Has Two Seating Capacity");
}
public void brakes()
{
 System.out.println("Bike Has Disk Brakes");
 } }
class VehicleDemo
{
public static void main(String args[])
{
Vehicle v=new Car();
Vehicle v1=new Bike();
v.wheels();
v.seating();
v.brakes();
 v1.wheels();
```

```
v1.seating();
v1.brakes();
 } }
```

2. Write a program to make a package Balance in which has Account class with Display_Balance method in it. Import Balance package in another program to access Display_Balance method of Account class.

```
class BankBal
{
        public static void main(String ar[])
{
try
{
        balance.Account a=new balance.Account();
        a.read();
        a.disp();
}
catch(Exception e)
        {       System.out.println(e); }
        }
}
```

```
package balance;
import java.io.*;
public class Account
{
        long acc,bal;
        String name;
public void read()throws Exception
{
        DataInputStream in=new DataInputStream(System.in);
        System.out.println("Enter the name :");
        name=in.readLine();
        System.out.println("Enter the account number :");
        acc=Long.parseLong(in.readLine());
        System.out.println("Enter the account balance :");
        bal=Long.parseLong(in.readLine());
}
public void disp()
{
System.out.println("~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~");
System.out.println("--- Account Details ---");
System.out.println("~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~");
System.out.println("Name :"+name);
System.out.println("Account number :"+acc);
```

```
System.out.println("Balance :"+bal);
}
}
```

3. Design a super class called Employee with details as EmployeeId, name, Phone, Salary. Extend this class by writing three subclasses namely Teaching (domain, publications), Technical (skills), and Contract (period). Write a JAVA program to read and displayat least 3 Employee objects of all three categories.

```
import java.util.Scanner;
class  Employee {
   String empId;
   String empName;
   long empPhone;
   float empSalary;
   public void accept() {
      Scanner scanner = new Scanner(System.in);
      System.out.print("Enter Staff Id: ");
      empId = scanner.next();
      System.out.print("Enter Name: ");
      empName = scanner.next();
      System.out.print("Enter Phone: ");
      empPhone = scanner.nextLong();
      System.out.print("Enter Salary: ");
      empSalary = scanner.nextFloat();
   }
   public void display() {
      System.out.println("Staff Id: " + empId);
      System.out.println("Name: " + empName);
      System.out.println("Phone: " + empPhone);
      System.out.println("Salary: " + empSalary);
   }
}
class Teaching extends Employee {
   String domain;
   int n;
   public void accept() {
      super.accept();
      Scanner scanner = new Scanner(System.in);
      System.out.print("Enter Domain: ");
      domain = scanner.next();
      System.out.print("Enter Number of Publications: ");
      n = scanner.nextInt();
      System.out.println("\n");
   }
   public void display() {
      super.display();
      System.out.println("Domain: " + domain);
```

```
        System.out.println("Publications:" + n);
        System.out.println("\n");
    }
}
class Technical extends Employee {
    String skill;
    public void accept() {
        super.accept();
        Scanner scanner = new Scanner(System.in);
        System.out.print("Enter technical Skills: ");
        skill = scanner.nextLine();
        System.out.println("\n");
    }
    public void display() {
        super.display();
        System.out.println("Technical Skills: " + skill);
        System.out.println("\n");
    }
}
class Contract extends Employee {
    int period;
    public void accept() {
        super.accept();
        Scanner scanner = new Scanner(System.in);
        System.out.print("Enter Period: ");
        period = scanner.nextInt();
        System.out.println("\n");
    }
    public void display() {
        super.display();
        System.out.println("Contract Period: " + period);
    }
}
class EmployeeFour {
    public static void main(String[] args) {
        Teaching teaching = new Teaching();
        System.out.println("Enter the details of Teaching Staff");
        teaching.accept();
        Technical technical = new Technical();
        System.out.println("Enter the details of Technical Staff");
        technical.accept();
        Contract contract = new Contract();
        System.out.println("Enter the details of Contract Staff");
        contract.accept();
        System.out.println("The details of Teaching Staff");
        teaching.display();
        System.out.println("The details of Technical Staff");
        technical.display();
```

```
            System.out.println("The details of Contract Staff");
            contract.display();
        }
    }
```

4. Java program to read two integers a and b. Compute a/b and print, when b is not zero. Raise an exception when b is equal to zero.

```
import java.util.Scanner;
class division
{ public static void main(String[] args)
{
int a,b,result;
Scanner input =new Scanner(System.in);
System.out.println("Input two integers");
a=input.nextInt();
b=input.nextInt(); t
ry {
result=a/b;
System.out.println("Result="+result); }
catch(ArithmeticException e) {
System.out.println("exception caught: Divide by zero error"+e); } }
```

5. Implement Java Program to Get the Components of any give URL such as Protocol, file, port and host.

```
import java.net.URL;

public class URLMain {
  public static void main(String[] args) {
    try {
      URL url = new URL("https://www.example.com/path/to/file.html?key=value#fragment");
      System.out.println("Protocol: " + url.getProtocol());
      System.out.println("Host: " + url.getHost());
      System.out.println("Port: " + url.getPort());
      System.out.println("Path: " + url.getPath());
      System.out.println("Query: " + url.getQuery());
      System.out.println("Fragment: " + url.getRef());
    } catch (Exception e) {
      System.out.println("Error: " + e.getMessage());
    }
  }
}
```

6. Implement client-server communication, where client can send the message and server can receive the message without internet.

SERVERSIDE.JAVA

```
import java.io.*; import java.net.*;
public class ServerSide
 { public static void main(String[]args)
{ try{
ServerSocketss=new ServerSocket(3306);
Socket s=ss.accept();//establishes connection
DataInputStream dis=new DataInputStream(s.getInputStream());
String str=(String)dis.readUTF();
System.out.println("message= "+str); ss.close();
}catch(Exception e){System.out.println(e);}
}
}
```

CLIENTSIDE.JAVA

```
import java.io.*;
 import java.net.*;
public class ClientSide {
public static void main(String[] args) {
try{
Socket s=new Socket("localhost",3306);
DataOutputStreamdout=new DataOutputStream(s.getOutputStream());
dout.writeUTF("Hello Server");
dout.flush()
dout.close();
s.close();
}
catch(Exception e)
{
System.out.println(e);}
}
}
```

7. Write a JDBC program to insert, update, delete, and select queries from database

```
import java.sql.*;

public class InsertDemo {

// JDBC driver name and database URL

static final String JDBC_DRIVER = "com.mysql.jdbc.Driver"; static final String DB_URL =
"jdbc:mysql://localhost:3306/";

// Database credentials

static final String USER = "root";

static final String PASS = "1234"; static final String dbName = "jdbctest";

public static void main(String[] args) { Connection conn = null;
Statement stmt = null; try{
//STEP 2: Register JDBC driver Class.forName("com.mysql.jdbc.Driver");

//STEP 3: Open a connection System.out.println("Connecting to a selected database...");
conn = DriverManager.getConnection(DB_URL+dbName,USER,PASS);
System.out.println("Connected database successfully...");
//STEP 4: Execute a query System.out.println("Inserting records into the table..."); stmt =
conn.createStatement();

String sql = "INSERT INTO employees " +

"VALUES ('5001', '30', 'Surendra', 'K','Coimbatore')"; stmt.executeUpdate(sql);
sql = "INSERT INTO employees " +

"VALUES ('5002', '21', 'Athul', 'K','Wayanad')";

stmt.executeUpdate(sql);

sql = "INSERT INTO employees " +

"VALUES ('5003', '20', 'Yadhu', 'K T','Palakkad')";

stmt.executeUpdate(sql);

sql = "INSERT INTO employees " +

"VALUES ('5004', '23', 'Pravin', 'K P','Thrissur')";

stmt.executeUpdate(sql);


sql = "UPDATE employees " +
```

```
"SET age = 21 WHERE id in (5003, 5004)";

stmt.executeUpdate(sql);

System.out.println("Inserted records into the table...");


String sql2="DELETE FROM employees where id=5004"; stmt.executeUpdate(sql2);
System.out.println("deleted records in the table where id is 5004...");
ResultSet rs= stmt.executeQuery("select * from employees"); while(rs.next()){
System.out.println(rs.getString(1)+ " "+rs.getString(2)+ " "+rs.getString(3)+ " "+rs.getString(4)+ "
"+rs.getString(5)+" ");
}


}catch(SQLException se){

//Handle errors for JDBC se.printStackTrace();
}catch(Exception e){
//Handle errors for Class.forName e.printStackTrace();
}finally{

//finally block used to close resources try{
if(stmt!=null) conn.close();
}catch(SQLException se){

}// do nothing try{
if(conn!=null) conn.close();
}catch(SQLException se){ se.printStackTrace();
}//end finally try
}//end try System.out.println("Goodbye!");
}//end main

}
```

8. Create web page authentication using JSP and JDBC connectivity (Login Application) with session

PROGRAMS:

Login.html

```
<html>
<head>
<title>Login</title>
<meta charset="UTF-8">
<meta name="viewport" content="width=device-width, initial-scale=1.0">
</head>
<body>
<form action="login.jsp" method="post" id="styleform">
<h2>Login Authentication</h2><hr color="black"><br>
Username: <input type="text" name="user"/><br><br>
Password: <input type="password" name="pwd"/><br><br><br>
<input type="submit" value="Submit" id="stylesub"/>
</form>
</body>
</html>
```

Login.jsp

```
<%@page contentType="text/html" pageEncoding="UTF-8"%>
<!DOCTYPE html>
<html>
<head>
<meta http-equiv="Content-Type" content="text/html; charset=UTF-8">
<title>Login</title>
</head>
<body>
<%@ page import ="java.sql.*" %>
<%@ page import ="javax.sql.*" %>
<%String userid = request.getParameter("user");
```

```
 String pwd = request.getParameter("pwd");
Class.forName("com.mysql.jdbc.Driver");
java.sql.Connection con =
DriverManager.getConnection("jdbc:mysql://localhost:3306/emp","users","");
 Statement st= con.createStatement();
ResultSet rs= st.executeQuery("select * from users where user_id="'+userid+'"");
 if(rs.next())
 {if(rs.getString(2).equals(pwd)) {
session.setAttribute("user",rs.getString(3));
 String name=(String)session.getAttribute("user");
out.println("Welcome "+ name);
}else
System.out.println("Invalid password try again");
 }
 %></body>
</html>
```

9. Write a Servlet program to demonstrate hit counter

Java file

```
import java.io.*;
import java.sql.Date;
import java.util.*;
import javax.servlet.*;
import javax.servlet.http.*;
public class PageHitCounter extends HttpServlet {
private int hitCount;
public void init() {
hitCount = 0; }
public void doGet(HttpServletRequest request, HttpServletResponse response)
throws ServletException, IOException {
// Set response content type
response.setContentType("text/html");
hitCount++;
PrintWriter out = response.getWriter();
String title = "Total Number of Hits";
String docType = "<!doctype html public \"-//w3c//dtd html 4.0 " +
"transitional//en\">\n";
out.println(docType +
"<html>\n" +
"<head><title>" + title + "</title></head>\n" +
"<body bgcolor = \"#f0f0f0\">\n" +
"<h1 align = \"center\">" + title + "</h1>\n" +
"<h2 align = \"center\">" + hitCount + "</h2>\n" +
"</body>
</html>"
);}
public void destroy() {
```

```
// This is optional step but if you like you
// can write hitCount value in your database.
}}
```

Xml file

```
<servlet>
 <servlet-name>PageHitCounter</servlet-name>
 <servlet-class>PageHitCounter</servlet-class>
</servlet>
<servlet-mapping>
 <servlet-name>PageHitCounter</servlet-name>
 <url-pattern>/PageHitCounter</url-pattern>
</servlet-mapping>
```