MAWLANA BHASHANI SCIENCE AND TECHNOLOGY UNIVERSITY

SANTOSH, TANGAIL-1902

DEPARTMENT OF INFORMATION AND COMMUNICATION TECHNOLOGY

Lab Report

Lab Report No : 01

Lab Report on : Comparing Abstract Classes and Interfaces for Multiple Inheritance in Java

Course Title : Software Engineering and Project Management Lab

Course Code : ICT 3108

Submitted By	Submitted To
Name: Monirojjaman Ayoive ID: IT23013 3rd Year, 1st Semester Session: 2022-2023 Dept. of ICT, MBSTU.	Dr. Ziaur Rahman Professor, Dept. of ICT, MBSTU.

Date of Performance:

Date of Submission:

Lab report-1:

In java, multiple inheritance means a class can get features from more than one parent. An abstract class does not support multiple inheritance. A class can extend only one abstract class. An interface supports multiple inheritance in java. A class can implement more than one interface at the same time. Abstract classes can have both abstract and normal methods. Interfaces contain only abstract methods. We use an abstract class when classes are closely related. We use an interface when multiple inheritance is needed.

Therefore, interfaces are used to solve the multiple inheritance problem in Java.

Java code example:

```
interface A{
        void show();
    }
interface B{
        void print();
    }
class Test implements A, B{
    public void show(){
     system.out.println("Interface A");
     }
    public void print(){
    system.out.println("Interface B");
    }
```

MAWLANA BHASHANI SCIENCE AND TECHNOLOGY UNIVERSITY

SANTOSH, TANGAIL-1902

DEPARTMENT OF INFORMATION AND COMMUNICATION TECHNOLOGY

Lab Report

Lab Report No : 02

Lab Report on : Implementing Encapsulation for Secure Data Handling

Course Title : Software Engineering and Project Management Lab

Course Code : ICT 3108

Submitted By	Submitted To
Name: Monirojjaman Ayoive ID: IT23013 3rd Year, 1st Semester Session: 2022-2023 Dept. of ICT, MBSTU.	Dr. Ziaur Rahman Professor, Dept. of ICT, MBSTU.

Date of Performance:

Date of Submission:

Lab report-2:

Encapsolation is an oop concept that hides data and protect them from direct access. It ensures data sequrity by making variable private inside a class. Outside classes cannot change data directly. Data can be changed only using public methods. These methods check the values before saving them. This prevents wrong or harmful data from entering the system. Encapsolation also keeps data consistent and correct. If invalid data is given, the method rejects it. In bank system, this is very important for safety. Therefore, encapsolation ensures both data and security.

Java code:

```
Class_Account{
  private string accountNumber;
  private double balance;

  public void setAccountNumber(string accNo){
     if(accNo!=null && !accNo.isempty()){
        accountNumbers = accNo;
        }
        else{
        system.out.println("Invalid account number");
     }
  }
  public void setInitialBalance(double amount){
  if (amount>=0){
     balance =amount;
     }
  else{
     system.out.println("Balance cannot be negative");
     }
  }
}
```

MAWLANA BHASHANI SCIENCE AND TECHNOLOGY UNIVERSITY

SANTOSH, TANGAIL-1902

DEPARTMENT OF INFORMATION AND COMMUNICATION TECHNOLOGY

Lab Report

Lab Report No : 04

Lab Report on : Using JDBC to Execute SELECT Queries and Handle Exceptions in Java

Course Title : Software Engineering and Project Management Lab

Course Code : ICT 3108

Submitted By	Submitted To
Name: Monirojjaman Ayoive ID: IT23013 3rd Year, 1st Semester Session: 2022-2023 Dept. of ICT, MBSTU.	Dr. Ziaur Rahman Professor, Dept. of ICT, MBSTU.

Date of Performance:

Date of Submission:

Lab report-4:

JDBC stands for Java Database connectivity. It is used to connect a Java program with a relational database. JDBC works as a bridge between Java application and database. The Java program sends SQL queries using JDBC. JDBC driver receaves the request and talks to the database. The database process the ~~database~~ query and sends to the ~~database~~ result back. JDBC driver receaves/returns the result to the Java program. JDBC allow data to be inserted, updated, deleted and read. It also handles connection and error management. Thus, JDBC manages smooth communication between Java and Database To execute a SELECT query, JDBC follows some fixed steps. First, a connection is created with the database. Then a statement

Prepared statement object is created. After that, a SELECT SQL query is written. The query is executed using executequery() method. This method returns a Resultset. try block is used to handle database operation safely. catch block handles SQL or runtime errors. finally block is used to close connection and resources.

Java code:

```
import java.sql*;
class selectExample{
  public static void main(string[] args){
    collection con=null;
    try{
      con= DriverManager.getConnection(
    "Jdbc:mysql://localhost:3306/testdb", "root", "password");
    statement st= con.createstatement();
    Resultset rs= st.executeQuery("SELECT *
      FROM student");
```

```
while(rs.next()){
 system.out.println(rs.getInt(1)+" "+rs.getString(2));
   }
}
 catch (Exception e) {
   system.out.println("Error occured");
} finally {
 try {
      if(con!=null)
       con.close();
        }
   catch(Exception e) {
     System.out.println("connection not closed");
    }
  }
 }
}
```

MAWLANA BHASHANI SCIENCE AND TECHNOLOGY UNIVERSITY

SANTOSH, TANGAIL-1902

DEPARTMENT OF INFORMATION AND COMMUNICATION TECHNOLOGY

Lab Report

Lab Report No : 05

Lab Report on : Servlet Controller in Java EE: Managing Model-View Flow with JSP

Course Title : Software Engineering and Project Management Lab

Course Code : ICT 3108

Submitted By	Submitted To
Name: Monirojjaman Ayoive ID: IT23013 3rd Year, 1st Semester Session: 2022-2023 Dept. of ICT, MBSTU.	Dr. Ziaur Rahman Professor, Dept. of ICT, MBSTU.

Date of Performance:

Date of Submission:

Lab report - 5:

In a Java EE application, a servlet works as a controller. The controller manages the flow between model and view The model contains buisness data and logic. The servlet gets data from the model. Then it sends this data to the view. JSP is used as the view to show data to the user. The servlet forward the request to JSP using RequestDispatcher. Data is sent using request attributes. JSP reads the data and displays it. Thus, the servlet controller controls the application flow.

Java code:

```
import java.io.*;
import javax.servlet.*;
import javax.http.*;
public class Helloservlet extends Httpservlet{
protected void doGet(HttpservletRequest req,
  HttpservletResponse res)
```

```
throws ServletException, IOException{
    String name = "student";
    req.setAttribute("msg", name);
RequestDispatcher rd = req.getRequestDispatcher(
        "hello.jsp");
    rd.forward(req, res);
  }
}
```

JSP code for views:

```
<html>
<body>
  <h2> Hello, ${msg} </h>
</body>
</html>
```

MAWLANA BHASHANI SCIENCE AND TECHNOLOGY UNIVERSITY

SANTOSH, TANGAIL-1902

DEPARTMENT OF INFORMATION AND COMMUNICATION TECHNOLOGY

Lab Report

Lab Report No : 06

Lab Report on : Secure and Efficient Database Insertion in JDBC Using PreparedStatement

Course Title : Software Engineering and Project Management Lab

Course Code : ICT 3108

Submitted By	Submitted To
Name: Monirojjaman Ayoive ID: IT23013 3rd Year, 1st Semester Session: 2022-2023 Dept. of ICT, MBSTU.	Dr. Ziaur Rahman Professor, Dept. of ICT, MBSTU.

Date of Performance:

Date of Submission:

Lab report-6:

Prepared statement is used to execute SQL queries safely in JDBC. It improves performance because the query is precompiled by the database. The same query can be used many time with different values. Prepared-statement is faster than statement for repeated queries. It also improves security by preventing SQL injection attacks. User input is treated as data, not as SQL code. Statement directly executes SQL and is less secure. Prepared statement uses placeholder(?) for values. These values are set using setter methods. So, prepared-statement is better for the performance and security.

Java code (Insert using preparedstatement):

```
import java.sql.*;
class insertExample{
    public static void main(String[] args){
  try{
   connection con= DriverManager.getConnection(
 "jdbc:mysql://localhost:3306/testdb","root","password");
String sql="Insert Into student(id,name) values(?,?)";
PreparedStatement ps=con.preparedstatement(sql);
   ps.setInt(1,1);
   ps.setString(2,"Rahim");
   ps.executeUpdate();
  System.out.println("Record inserted");
  con.close();
  } catch (Exception e){
    System.out.println("Error occurred");
   }
  }
}
```

MAWLANA BHASHANI SCIENCE AND TECHNOLOGY UNIVERSITY

SANTOSH, TANGAIL-1902

DEPARTMENT OF INFORMATION AND COMMUNICATION TECHNOLOGY

Lab Report

Lab Report No : 07

Lab Report on : ResultSet in JDBC: Fetching Data with next(), getString(), and getInt()

Course Title : Software Engineering and Project Management Lab

Course Code : ICT 3108

Submitted By	Submitted To
Name: Monirojjaman Ayoive ID: IT23013 3rd Year, 1st Semester Session: 2022-2023 Dept. of ICT, MBSTU.	Dr. Ziaur Rahman Professor, Dept. of ICT, MBSTU.

Date of Performance:

Date of Submission:

Lab report-7:

Resultset is an object in JDBC that stores data returned from a database query. It is mainly used with Select queries. Resultset works like a table with rows and columns. The next() method moves the cursor to the next row. It returns true if data is available. The setstring() method is available used to read string type data. Data is read column by column from Resultset. Resultset helps Java programs fetch database records easily. Thus, it is very important for retrieving data in JDBC.

Java code(Resultset usage):

```
import java.sql*;
class ResultSetExample{
 public static void main(string[] args){
  try {
```

```
Connection con=DriverManager.getConnection(
"Jdbc:mysql://Localhost: 3306/testdb", "root", "password");
statement st= con.createstatement();
Resultset rs= st.executeQuery("SELect id, name
                from student");

while(rs.next()){
    int id=rs.getInt("int");
    String name = rs.getString("Name");
    System.out.println(id+" "+name);
    }

con.close();
} catch (Exception e){
    System.out.println("Error occured");
    }
  }

}
```

MAWLANA BHASHANI SCIENCE AND TECHNOLOGY UNIVERSITY

SANTOSH, TANGAIL-1902

DEPARTMENT OF INFORMATION AND COMMUNICATION TECHNOLOGY

Lab Report

Lab Report No : 08

Lab Report on : Development of RESTful Web Services Using Spring Boot

Course Title : Software Engineering and Project Management Lab

Course Code : ICT 3108

Submitted By	Submitted To
Name: Monirojjaman Ayoive ID: IT23013 3rd Year, 1st Semester Session: 2022-2023 Dept. of ICT, MBSTU.	Dr. Ziaur Rahman Professor, Dept. of ICT, MBSTU.

Date of Performance:

Date of Submission:

Lab report – 08:

Describe the difference between the entity managers persist(), merge() and remove() operations. When would you each method in a typical database transaction scenario. Difference table is given below:

Method	Purpose	Return Type	When to use
persist()	Marks a new entity managed and schedules it for insertion	void	When inserting a new record
Merge()	Copies the stated a detached entity into a managed entity	Manage entity	When updating an entity.
remove()	Deletes managed entity from the database	void	When deleting a record

Detailed explaination and used scenarioes:

1. persist (object entity):

. Used for saving new entities to the db.
. Adds the object to the persistence context and inserts it into the database
. Entity must be transient.

2. merge():

. used for updating detached entities
. Return a new managed copy of the entity with its state copied from the input.
. Entity can be detached / transient.

3. remove():

. used for deleting an entity from database.
. Marks the entity for deletion, it is removed at commit.
. Entity must be managed.

MAWLANA BHASHANI SCIENCE AND TECHNOLOGY UNIVERSITY

SANTOSH, TANGAIL-1902

DEPARTMENT OF INFORMATION AND COMMUNICATION TECHNOLOGY

Lab Report

Lab Report No : 09

Lab Report on : Demonstrating a Java Project with GUI and Core Implementation

Course Title : Software Engineering and Project Management Lab

Course Code : ICT 3108

Submitted By	Submitted To
Name: Monirojjaman Ayoive ID: IT23013 3rd Year, 1st Semester Session: 2022-2023 Dept. of ICT, MBSTU.	Dr. Ziaur Rahman Professor, Dept. of ICT, MBSTU.

Date of Performance:

Date of Submission:

Lab repart-09:

I developed a Hotel Booking Management system using Java swing for the graphical user user interface and mysql for database Management.

Graphical user Interface(GUI) Description:

Main window:

Public class hotelBooking extends Jframe.

Input section (Top parsel)

```
idField = new Jtexfield();
nameField = new JtextField();
rigHtsfield = new JtextField();
Room typeBox = new Jcombox<><new string[]
  "single ($500)", "Double($1000)", "Deluxe($2000)"});
```

Button section (Middle Panel):

```
J Button b1 = new JButton("Book");
Jbutton b2 = new JButton("update");
Jbutton b3 = new JButton("cancel");
```

Display section (Botton panel):

```
displayArea = new JTextArea();
```

Important code explanation:

Database connection (JDBC):

```
connection c = DriverManager.getConnection(
"Jdbc:mysql:///localhost:3306/hotel db", "~~ty~~ub@23");
```

Booking update and Delete operation:

```
b1 addAction Listener(e→
    runsql("Insert into booking(name,room-type,
            neights value(?,?,?)");

b2. add action listener(e→ runsql("Update booking
    set name=?, room-type=?, nights=?
    where id=?"));
b3. add action listener (e→ runsql("Delete from
    booking where id=?"));
```

Prepared statement usage:

```
Prepared statement ps = c.preparedstament (sql);
```

Room cost calculation:

```
int p= roomtype.equals ("single") ? 500
    :roomType.equals("Double")? 1000:2000;
```

Total fee = Room Price x Number of Nights.