



PRACTICAL NOTE BOOK ON ADVANCED JAVA

Department of Computer Science
Assam University, Silchar

MRINAL KANTI PAUL
Roll no.: 03 Paper Code: 605(b)

INDEX

Sr.	Name	Page	Signature
1.	Write a XML Program to Print CD catalog	2	
2.	Write a XML Program to Print Food menu	3	
3.	Write a XML Program to Print Library catalog	4	
4.	Write a XML Program to Print Address Book	5	
5.	Write a java code to connect to mysql using JDBC	6	
6.	Write a java code to create a database	7	
7.	Write a java code to create a table in a database	8	
8.	Write a java code to insert data in a database	9	
9.	Write a java code to select record from a database	10	
10.	Write a demo servlet program	11	
11.	Write a hello world sevlet	12	
12.	Write a servlet program for TCP Client	13	
13.	Write a servlet program for TCP Server	14	

1. Write a XML program to print CD catalog.

Ans.

```
<CATALOG>
<CD>
<TITLE>Empire Burlesque</TITLE>
<ARTIST>Bob Dylan</ARTIST>
<COUNTRY>USA</COUNTRY>
<COMPANY>Columbia</COMPANY>
<PRICE>10.90</PRICE>
<YEAR>1985</YEAR>
</CD>
<CD>
<TITLE>Hide your heart</TITLE>
<ARTIST>Bonnie Tyler</ARTIST>
<COUNTRY>UK</COUNTRY>
<COMPANY>CBS Records</COMPANY>
<PRICE>9.90</PRICE>
<YEAR>1988</YEAR>
</CD>
<CD>
<TITLE>Greatest Hits</TITLE>
<ARTIST>Dolly Parton</ARTIST>
<COUNTRY>USA</COUNTRY>
<COMPANY>RCA</COMPANY>
<PRICE>9.90</PRICE>
<YEAR>1982</YEAR>
</CD>
<CD>
<TITLE>Eros</TITLE>
<ARTIST>Eros Ramazzotti</ARTIST>
<COUNTRY>EU</COUNTRY>
<COMPANY>BMG</COMPANY>
<PRICE>9.90</PRICE>
<YEAR>1997</YEAR>
</CD>
<CD>
<TITLE>One night only</TITLE>
<ARTIST>Bee Gees</ARTIST>
<COUNTRY>UK</COUNTRY>
<COMPANY>Polydor</COMPANY>
<PRICE>10.90</PRICE>
<YEAR>1998</YEAR>
</CD>
</CATALOG>
```

2. Write a XML program to print food menu.

Ans.

```
<breakfast_menu>
<food>
<name>Belgian Waffles</name>
<price>$5.95</price>
<description>Two of our famous Belgian Waffles with plenty of real maple syrup</description>
<calories>650</calories>
</food>
<food>
<name>Strawberry Belgian Waffles</name>
<price>$7.95</price>
<description>Light Belgian waffles covered with strawberries and whipped cream</description>
<calories>900</calories>
</food>
<food>
<name>Berry-Berry Belgian Waffles</name>
<price>$8.95</price>
<description>Light Belgian waffles covered with an assortment of fresh berries and whipped
cream</description>
<calories>900</calories>
</food>
<food>
<name>French Toast</name>
<price>$4.50</price>
<description>Thick slices made from our homemade sourdough bread</description>
<calories>600</calories>
</food>
<food>
<name>Homestyle Breakfast</name>
<price>$6.95</price>
<description>Two eggs, bacon or sausage, toast, and our ever-popular hash browns</description>
<calories>950</calories>
</food>
</breakfast_menu>
```

3. Write a XML program to print library catalog.

Ans.

```
<?xml version="1.0" encoding="ISO-8859-1"?>

<?xml-stylesheet type="text/css" href="bookstore.css"?>

<bookstore category="library">

<bookstore>

<book>

<title>Everyday Italian</title>

<author>Giada De Laurentiis</author>

<year>2005</year>

<price>30.00</price>

</book>

<book>

<title>Harry Potter</title>

<author>J K. Rowling</author>

<year>2005</year>

<price>29.99</price>

</book>

<book>

<title>Learning XML</title>

<author>Erik T. Ray</author>

<year>2003</year>

<price>39.95</price>

</book>

</bookstore>
```

4. Write a XML program to print Address Book.

Ans.

```
<?xml version="1.0" encoding="ISO-8859-1"?>

<?xml-stylesheet type="text/css" href="addressbook.css"?>

<addressbook category="business">

<addressbook>

    <person>

        <name>Nadir Laskar</name>

        <city>Silcahr</city>

        <state>Assam</state>

        <phone>80564322345</phone>

    </person>

    <person>

        <name>Mrinal Paul</name>

        <city>Silchar</city>

        <state>Assam</state>

        <phone>9088765432</phone>

    </person>

    <person>

        <name>Mrinal Roy</name>

        <city>Hilakandi</city>

        <state>Assam</state>

        <phone>8876828762</phone>

    </person>

    <person>
```

<name>Alokesh Dey</name>

<city>Karimganj</city>

<state>Assam</state>

<phone>9984536282</phone>

</person>

<person>

<name>Poonam Pandey</name>

<city>Silchar</city>

<state>Assam</state>

<phone>908765432</phone>

</person>

</addressbook>

5. Write a code to connect with database using JDBC.

Ans.

```
import java.sql.*;
public class JDBCExample {

    public static void main(String[] args) {
        Connection conn = null;
        Statement stmt = null;

        try{
            Class.forName("com.mysql.jdbc.Driver");
            System.out.println("Connecting to database...");
            conn = DriverManager.getConnection("jdbc:mysql://localhost/", "root", "root");
            stmt = conn.createStatement();
            System.out.println("Connection Sucessfull...");
        }
        catch(SQLException se){
            se.printStackTrace();
        }
        catch(Exception e){
            e.printStackTrace();
        }
        finally{
            conn.close();
            stmt.close();
        }
    }
}
```


6. Write a java code to create a database.

Ans.

```
import java.sql.*;
public class JDBCExample {

public static void main(String[] args) {
Connection conn = null;
Statement stmt = null;
try{
    Class.forName("com.mysql.jdbc.Driver");
    System.out.println("Connecting to database...");
    conn = DriverManager.getConnection("jdbc:mysql://localhost/", "root", "root");
    stmt = conn.createStatement();
    stmt.executeUpdate("CREATE DATABASE STUDENTS");
    System.out.println("Database created successfully...");
}
catch(SQLException se){
    se.printStackTrace();
}
catch(Exception e){
    e.printStackTrace();
}
finally{
    conn.close();
    stmt.close();
}
}
}
```

7. Write a java code to create a table in a database.

Ans.

```
import java.sql.*;
public class JDBCExample {

public static void main(String[] args) {
Connection conn = null;
Statement stmt = null;
try{
    Class.forName("com.mysql.jdbc.Driver");
    System.out.println("Connecting to database...");
    conn = DriverManager.getConnection("jdbc:mysql://localhost/students","root","root");
    stmt = conn.createStatement();
    stmt.executeUpdate("""CREATE TABLE REGISTRATION " +
        "(id INTEGER not NULL, " +
        " first VARCHAR(25), " +
        " last VARCHAR(25), " +
        " age INTEGER, " +
        " PRIMARY KEY ( id ))""");
    System.out.println("Table created successfully...");
}
catch(SQLException se){
    se.printStackTrace();
}
catch(Exception e){
    e.printStackTrace();
}
finally{
    conn.close();
    stmt.close();
}
}
}
```

8. Write a Java code to insert record into database table.

Ans.

```
import java.sql.*;
public class JDBCExample {

public static void main(String[] args) {
Connection conn = null;
Statement stmt = null;
try{
    Class.forName("com.mysql.jdbc.Driver");
    System.out.println("Connecting to database...");
    conn = DriverManager.getConnection("jdbc:mysql://localhost/student","root","root");
    stmt = conn.createStatement();
    stmt.executeUpdate("INSERT INTO Registration VALUES (100, 'Mrinal', 'Paul', 22)");
    stmt.executeUpdate("INSERT INTO Registration VALUES (101, 'Mrinal', 'Roy', 21)");
    stmt.executeUpdate("INSERT INTO Registration VALUES (102, 'Nadir', 'Laskar', 22)");
    stmt.executeUpdate("INSERT INTO Registration VALUES (104, 'Biswa', 'Das', 23)");
    stmt.executeUpdate("INSERT INTO Registration VALUES (107, 'Prtap', 'Deb', 19)");
    System.out.println("Data inserted successfully...");
}
catch(SQLException se){
    se.printStackTrace();
}
catch(Exception e){
    e.printStackTrace();
}
finally{
    conn.close();
    stmt.close();
}
}
```

9. Java code to select record from database

Ans.

```
import java.sql.*;
public class JDBCExample {

public static void main(String[] args) {
Connection conn = null;
Statement stmt = null;
try{
    Class.forName("com.mysql.jdbc.Driver");
    System.out.println("Connecting to database...");
    conn = DriverManager.getConnection("jdbc:mysql://localhost/student","root","root");
    stmt = conn.createStatement();
    String sql = "SELECT id, first, last, age FROM Registration";
    ResultSet rs = stmt.executeQuery(sql);
    while(rs.next()){
        int id = rs.getInt("id");
        int age = rs.getInt("age");
        String first = rs.getString("first");
        String last = rs.getString("last");
        System.out.print("ID: " + id);
        System.out.print(", Age: " + age);
        System.out.print(", First: " + first);
        System.out.println(", Last: " + last);
    }
}
catch(SQLException se){
    se.printStackTrace();
}
catch(Exception e){
    e.printStackTrace();
}
finally{
    conn.close();
    stmt.close();
}
}
```

10. Write a demo servlet program.

Ans.

```
import java.io.*;
import javax.servlet.*;
import javax.servlet.annotation.WebServlet;
import javax.servlet.http.*;

@WebServlet(urlPatterns = {"/demo"})
public class demo extends HttpServlet {

    @Override
    public void doGet(HttpServletRequest request, HttpServletResponse response)
        throws ServletException, IOException {
        response.setContentType("text/html;charset=UTF-8");
        try (PrintWriter pw = response.getWriter()) {
            pw.println("<html><head><title>Servlet Demo</title></head><body>");
            pw.println("Welome to servlet Demo");
            pw.println("</body></html>");
        }
    }
}
```

11. Write a Hello world program in servlet.

Ans.

```
import java.io.IOException;
import java.io.PrintWriter;
import javax.servlet.ServletException;
import javax.servlet.annotation.WebServlet;
import javax.servlet.http.HttpServlet;
import javax.servlet.http.HttpServletRequest;
import javax.servlet.http.HttpServletResponse;
```

```
@WebServlet(urlPatterns = {"/hello_world"})
public class hello_world extends HttpServlet {
```

```
    private String message;
    @Override
    public void init() throws ServletException
    {
        message = "Hello World";
    }
```

```
    @Override
    protected void doGet(HttpServletRequest request, HttpServletResponse response)
        throws ServletException, IOException {
        response.setContentType("text/html");
        PrintWriter out = response.getWriter();
        out.println("<h1>" + message + "</h1>");
    }
```

```
    @Override
    public void destroy()
    {

    }
}
```

12. Write a servlet program for TCP Client Side.

Ans.

```
import java.io.BufferedReader;
import java.io.DataOutputStream;
import java.io.InputStreamReader;
import java.net.Socket;

public class tcpclient {
    public static void main(String argv[]) throws Exception{
        String sentence;
        String modifiedSentence;
        BufferedReader inFromUser = new BufferedReader(new InputStreamReader(System.in));

        try (Socket clientSocket = new Socket("localhost", 6789)) {
            DataOutputStream outToServer = new DataOutputStream(clientSocket.getOutputStream());
            BufferedReader inFromServer = new BufferedReader(new
InputStreamReader(clientSocket.getInputStream()));

            sentence = inFromUser.readLine();
            outToServer.writeBytes(sentence + '\n');
            modifiedSentence = inFromServer.readLine();
            System.out.println(modifiedSentence);
            clientSocket.close();
        }
    }
}
```

13. Write a servlet program for TCP Server side.

Ans.

```
import java.io.BufferedReader;
import java.io.DataOutputStream;
import java.io.InputStreamReader;
import java.net.ServerSocket;
import java.net.Socket;

public class tcpserver {
    public static void main(String args[]) throws Exception{
        String clientSentence;
        String capitalizedSentencs;
        ServerSocket welcomeSocket = new ServerSocket(6789);
        while (true) {
            Socket connectionSocket = welcomeSocket.accept();
            BufferedReader inFormClient = new BufferedReader(new
InputStreamReader(connectionSocket.getInputStream()));
            DataOutputStream outToClient = new
DataOutputStream(connectionSocket.getOutputStream());
            clientSentence = inFormClient.readLine();
            capitalizedSentencs = clientSentence.toUpperCase() + '\n';
            outToClient.writeBytes(capitalizedSentencs);
        }
    }
}
```