**PART-A**

**PART-B**

**\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\***

**PA1: Write a program to convert numbers into words using Enumerations with constructors, methods and instance variables. (INPUT RANGE-0 TO 99999) EX: 36 THIRTY-SIX**

**NAME: NISHANTH ROLL NO:22536**

**CLASS: III BCA**

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

package numbertoword;

import java.IOException;

import java.util.Scanner;

import java.io.\*;

public class NumberToWord {

enum Units{ ZERO(""),ONE("ONE"),TWO("TWO"),THREE("THREE"),FOUR("FOUR"),FIVE("FIVE"),

SIX("SIX"),SEVEN("SEVEN"),EIGHT("EIGHT"),NINE("NINE");

private final String word;

Units(String word){

this.word=word;

}

public String getWord(){

return word;

}

}

enum Tens{

TEN("TEN"),ELEVEN(" ELEVEN"),TWElVE(" TWElVE"),

THIRTEEN(" THIRTEEN"),FOURTEEN(" FOURTEEN"),FIFTEEN(" FIFTEEN"),

SIXTEEN(" SIXTEEN"),SEVENTEEN(" SEVENTEEN"),EIGHTEEN(" EIGHTEEN"),

NINETEEN(" NINETEEN"),TWENTY(" TWENTY"),THIRTY(" THIRTY"),

FOURTY(" FOURTY"),FIFTY(" FIFTY"),SIXTY(" SIXTY"),SEVENTY(" SEVENTY"),

EIGHTY(" EIGHTY"),NINTY(" NINTY");

private final String word;

Tens(String word){

this.word=word;

}

public String getWord(){

return word;

}

}

enum Thousands{

THOUSAND("THOUSAND");

private final String word;

Thousands(String word){

this.word=word;

}

public String getWord(){

return word;

}

}

public static String covertToWords(int number){

if(number==0){

return "ZERO";

} String words=" ";

if(number/1000>0){

words+=convertThreeDigitsToWords(number/1000)+" "+Thousands.THOUSAND.getWord()+" ";

number%=1000;

}

words+=convertThreeDigitsToWords(number);

return words.trim();

}

private static String convertThreeDigitsToWords(int number){

String words=" ";

if(number/100>0)

{

words+=Units.values()[number/100].getWord()+" HUNDRED";

number%=100;

}

if(number>=20)

{

words+=Tens.values()[(number/10)+8].getWord()+ " ";

number%=10;

}

else if(number>=10)

{

words+=Tens.values()[number%10].getWord()+ " ";

number=0;

}

if(number>0)

{

words+=Units.values()[number].getWord()+" ";

number%=100;

}

return words;

}

public static void main(String[] args) {

// TODO code application logic here

Scanner Scanner=new Scanner(System.in);

System.out.println("Enter a number between o and 999999");

int number=Scanner.nextInt();

if (number<0||number>99999){

System.out.println("Please enter input between 0 to 99999");

}

else{

String result=covertToWords(number);

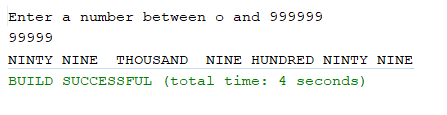
System.out.println(result);

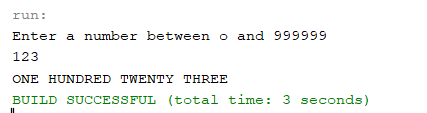
}

}

}

**OUTPUT:**





**\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\***

**PA2: Find the second maximum and second minimum in a set of numbers using auto boxing and unboxing.**

**NAME: NISHANTH ROLL NO:22536**

**CLASS: III BCA**

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

package autounboxing;

import java.util.Scanner;

import java.io.\*;

public static void main(String[] args)throws IOException {

// TODO code application logic here

BufferedReader br=new BufferedReader(new InputStreamReader (System.in));

int size;

System.out.println("Enter the size of the array:");

size=Integer.parseInt(br.readLine());

Integer[] arr=new Integer[size];

System.out.println("Enter a value:");

for(int i=0;i<size;i++){

int a;

a=Integer.parseInt(br.readLine());

arr[i]=a;

}

System.out.println("Array elements:");

for (int j=0;j<size;j++){

System.out.println(" "+arr[j]);

}

Integer i,j;

int n=arr.length;

int temp=0;

for(i=0;i<n;i++){

for(j=1;j<(n-i);j++){

if(arr[j-1]>arr[j]){

temp=arr[j-1];

arr[j-1]=arr[j];

arr[j]=temp;

}

}

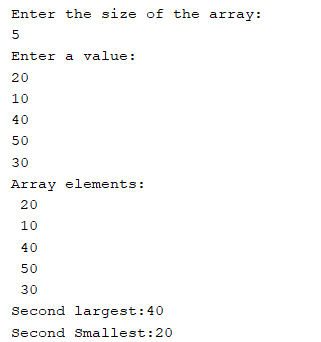
}

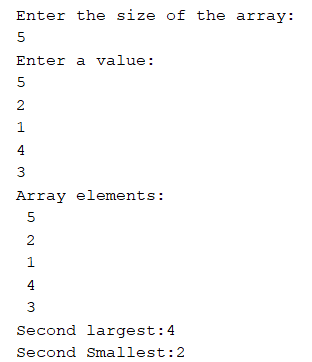
System.out.println("Second largest:" +arr[n-2]);

System.out.println("Second Smallest:" +arr[1]);

}

}

**OUTPUT:**



**\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\***

**PA3: Write a menu driven program to create an Arraylist and perform the following operations**

**i)Adding elements**

**ii) Sorting elements**

**iii) Replace an element with another**

**iv) Removing an element**

**v) Displaying all the elements**

**vi) Adding an element between two elements**

**NAME: NISHANTH ROLL NO:22536**

**CLASS: III BCA**

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

package arraylistpgm3;

import java.util.\*;

import java.io.\*;

class ArrayListPGM3 {

public static void main(String[] args)throws IOException {

BufferedReader br=new BufferedReader(new InputStreamReader (System.in));

ArrayList<Integer>arrayList=new ArrayList<>();

while(true){

System.out.println("\n Arralist operation menu:");

System.out.println("1.Add elements");

System.out.println("2.sort element");

System.out.println("3.Replace an element with another");

System.out.println("4.Remove element");

System.out.println("5.Display all elements");

System.out.println("6.Add an elements between two elements");

System.out.println("7.Exit");

System.out.println("Enter your choice:");

int choice=Integer.parseInt(br.readLine());

switch(choice){

case 1:

System.out.println("Enter an element to add:");

int value=Integer.parseInt(br.readLine());

arrayList.add(value);

System.out.println("Element added successfully");

break;

case 2:

Collections.sort(arrayList);

System.out.println("Elements sorted successfully");

break;

case 3:

System.out.println("Enter the element to be replaced:");

int existvalue=Integer.parseInt(br.readLine());

if(arrayList.contains(existvalue)){

System.out.println("Enter the new element:");

int newvalue=Integer.parseInt(br.readLine());

Collections.replaceAll(arrayList, existvalue, newvalue);

System.out.println("Element replaced successfully");

}

else

{

System.out.println("Elements does not exists");

}

break;

case 4:

System.out.println("Enter the element to be removed");

int element=Integer.parseInt(br.readLine());

if(arrayList.contains(element)){

arrayList.remove(element);

System.out.println("Element removed");

}

else{

System.out.println("Element not found");

}

break;

case 5:

System.out.println("Element in the arrayList:");

for(int num:arrayList){

System.out.println(num+" ");

}

System.out.println();

break;

case 6:

System.out.println("Enter the index position to add the element:");

int pos=Integer.parseInt(br.readLine());

if(pos<arrayList.size()){

System.out.println("Enter the element to add:");

int ele=Integer.parseInt(br.readLine());

arrayList.add(pos,ele);

System.out.println("Element added successfully between two elements");

}

else

{

System.out.println("Position out of bound");

}

break;

case 7:

System.out.println("Existing program....Thank You!!!!");

return;

default:

System.out.println("Invalid choice!!!");

}

}

}

}

**OUTPUT:**

**i)Adding elements**

Arralist operation menu:

1.Add elements

2.sort element

3.Replace an element with another

4.Remove element

5.Display all elements

6.Add an elements between two elements

7.Exit

Enter your choice:

1

Enter an element to add:

2

Element added successfully

Enter your choice:

1

Enter an element to add:

1

Element added successfully

Element in the arrayList:

2

1

4

3

**ii) Sorting elements**

Arralist operation menu:

1.Add elements

2.sort element

3.Replace an element with another

4.Remove element

5.Display all elements

6.Add an elements between two elements

7.Exit

Enter your choice:

2

Elements sorted successfully

Element in the arrayList:

1

2

3

4

**iii) Replace an element with another**

Arralist operation menu:

1.Add elements

2.sort element

3.Replace an element with another

4.Remove element

5.Display all elements

6.Add an elements between two elements

7.Exit

Enter your choice:

3

Enter the element to be replaced:

3

Enter the new element:

5

Element replaced successfully

**iv) Removing an element**

Arralist operation menu:

1.Add elements

2.sort element

3.Replace an element with another

4.Remove element

5.Display all elements

6.Add an elements between two elements

7.Exit

Enter your choice:

4

Enter the element to be removed:

3

Element removed successfully

**v) Displaying all the elements**

Arralist operation menu:

1.Add elements

2.sort element

3.Replace an element with another

4.Remove element

5.Display all elements

6.Add an elements between two elements

7.Exit

Enter your choice:

5

Element in the arrayList:

1

6

2

5

4

**vi) Adding an element between two elements**

Arralist operation menu:

1.Add elements

2.sort element

3.Replace an element with another

4.Remove element

5.Display all elements

6.Add an elements between two elements

7.Exit

Enter your choice:

6

Enter the index position to add the element:

1

Enter the element to add:

6

Element added successfully between two elements

**\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\***

**PA4: Write a java program to find words with even number of characters in a string, then swap the pair of characters in those words and also toggle the characters in a given string**

**EX:**

**Good Morning everyone**

**Output:**

**oGdo vereoyen**

**gOOD mORNING EVERYONE**

**NAME: NISHANTH ROLL NO:22536**

**CLASS: III BCA**

**\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\***

package pa4;

import java.io.\*;

import java.util.Scanner;

public class PA4 {

public static String SwapCharacters(String word){

char[] chars=word.toCharArray();

for(int i=0;i<chars.length-1;i+=2){

char temp=chars[i];

chars[i]=chars[i+1];

chars[i+1]=temp;

}

return new String(chars);

}

public static void main(String[] args)throws IOException {

BufferedReader br=new BufferedReader(new InputStreamReader (System.in));

System.out.println("Enter a String:");

String input=br.readLine();

String[]words=input.split(" ");

StringBuilder result=new StringBuilder();

StringBuilder output=new StringBuilder();

for(String word:words){

if( word.length()%2==0){

result.append(SwapCharacters(word)).append(" ");

}

}

for(char c:input.toCharArray()){

if(Character.isUpperCase(c)){

output.append(Character.toLowerCase(c));

}

else if(Character.isLowerCase(c)){

output.append(Character.toUpperCase(c));

}else{

output.append(c);

}

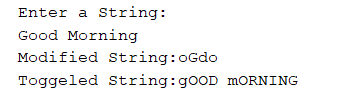
}

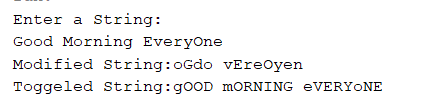
System.out.println("Modified String:"+result.toString().trim());

System.out.println("Toggeled String:"+output.toString().trim());

}

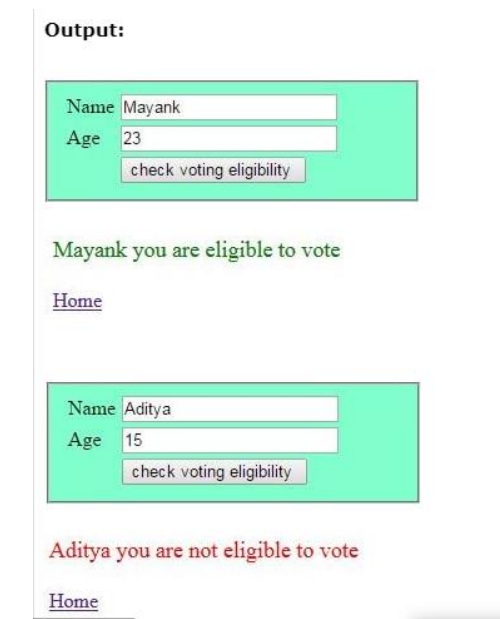
}

**OUTPUT:**

****

**\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\***

**PA5: Write a Servlet program that accepts the age and name and displays if the user is eligible for voting or not**



**NAME: NISHANTH ROLL NO:22536**

**CLASS: III BCA**

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

**index.html**

<!DOCTYPE html>

<html>

<head>

<title>Voting Eligibility Test</title>

<meta charset="UTF-8">

<meta name="viewport" content="width=device-width, initial-scale=1.0">

<style>

table{

background-color: aqua;

width:200px;

margin-top:100x;

margin-left: auto;

margin-right: auto;

border: solid 2px;

}

td{

padding:5px;

}

</style>

</head>

<body>

<form method="POST" action="CheckVoter">

<table>

<tr>

<td>Name</td>

<td><input type="text" name="uname"></td>

</tr>

<tr>

<td>Age</td>

<td><input type="text" name="age"></td>

</tr>

<tr>

<td></td>

<td><input type="submit" name="uname" value="check voting eligibility"></td>

</tr>

</table>

</form>

</body>

</html>

**CheckVotes.java**

package com;

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 = {"/CheckVoter"})

public class CheckVoter extends HttpServlet {

protected void processRequest(HttpServletRequest request, HttpServletResponse response)

throws ServletException, IOException {

response.setContentType("text/html;charset=UTF-8");

try (PrintWriter out = response.getWriter()) {

out.println("<!DOCTYPE html>");

out.println("<html>");

out.println("<head>");

out.println("<title>Servlet CheckVoter</title>");

out.println("</head>");

out.println("<body>");

out.println("<h1>Servlet CheckVoter at " + request.getContextPath() + "</h1>");

String name=request.getParameter("uname");

int age=Integer.parseInt(request.getParameter("age"));

if(age>18)

{

out.println("<h4 style=\"color:green\">" +name + " you are eligible to vote</h4>");

}else{

out.println("<h4 style=\"color:brown\">" +name + " you are not eligible tovote</h4>");

}

out.println("<a href=\"index.html\">Home</a>");

out.println("</body>");

out.println("</html>");

}

}

@Override

protected void doGet(HttpServletRequest request, HttpServletResponse response)

throws ServletException, IOException {

processRequest(request, response);

}

@Override

protected void doPost(HttpServletRequest request, HttpServletResponse response)

throws ServletException, IOException {

processRequest(request, response);

}

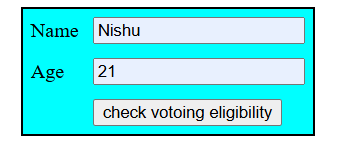
@Override

public String getServletInfo() {

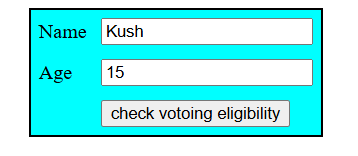
return "Short description";

}// </editor-fold>

}

**OUTPUT:  
**

****

****

****

**\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\***

**PA6: Write a JSP program to print first 10 Fibonacci and 10 prime numbers.**

**NAME: NISHANTH ROLL NO:22536**

**CLASS: III BCA**

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

<!DOCTYPE html>

<html>

<head>

<meta http-equiv="Content-Type" content="text/html; charset=UTF-8">

<title>Fibo And PRime</title>

</head>

<body>

<h4>Fibbonaci series</h4>

<%

int f1=0,f2=1,f3,i;

out.println(f1+"&nbsp;&nbsp;" +f2+"&nbsp;&nbsp;");

for(i=1;i<=10;i++)

{

f3=f1+f2;

out.println(f3+"&nbsp;&nbsp;");

f1=f2;

f2=f3;

}

%>

<h4>Prime Numbers</h4>

<%

int pn=2,count=1;

boolean isprime;

while(count<=10)

{

isprime=true;

for(i=2;i<=pn/2;i++)

{

if(pn%i==0)

{

isprime=false;

break;

}

}

if(isprime){

out.println(pn+"&nbsp;&nbsp;");

count++;

}

pn++;

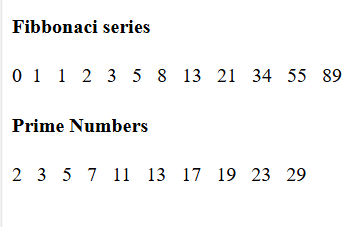
}

%>

</body>

</html>

**OUTPUT:**

****

**\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\***

**PA7: Write a JSP Program to design a shopping cart to add items, remove item and to display items from the cart using Sessions**

**NAME: NISHANTH ROLL NO:22536**

**CLASS: III BCA**

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

**Cart.jsp**

<%@page contentType="text/html" pageEncoding="UTF-8"%>

<!DOCTYPE html>

</head>

<body>

<body align="left">

<h1 >

Online Shopping

</h1>

<br>

<table width="300" height="300" border="0" >

<tr>

<td>

<form action="AddItem.jsp" method="post">

<img src="image/Pepsi.jpg" heigth="100px" width="100px"/>

<br>

<label for="Itemname" >Pepsi</label>

<br>

<input type="hidden" name="item" value="Pepsi" >

<label for="Price" name="Price" value="35">Rs. 35</label>

<br>

<input type="submit" value="Add to Cart">

</form>

</td>

<td>

<form action="AddItem.jsp" method="post">

<img src="image\coco.jpeg" heigth="100px" width="100px"/>

<br><!-- comment -->

<label for="Itemname" >Cola</label>

<br>

<input type="hidden" name="item" value="Cola" >

<label for="Price" name="Price" value="20">Rs. 20</label>

<br>

<input type="submit" value="Add to Cart">

</form>

</td>

<td>

<form action="AddItem.jsp" method="post">

<img src="image\Sprite.jpeg" heigth="100px" width="100px"/>

<br>

<label for="Itemname" >Sprite</label>

<br>

<input type="hidden" name="item" value="Sprite" >

<label for="Price" name="Price" value="10">Rs. 10</label>

<br>

<input type="submit" value="Add to Cart">

</form>

</td>

</table>

<a href="ViewCart.jsp">View Cart</a>

</body>

</html>

**AddItem.jsp**

<%@page contentType="text/html" pageEncoding="UTF-8"%>

<!DOCTYPE html>

<html>

<head>

<meta http-equiv="Content-Type" content="text/html; charset=UTF-8">

<title>Add Item</title>

</head>

<body>

<%@ page language="java" import="java.util.List, java.util.ArrayList" session="true" %>

<%

String item = request.getParameter("item");

if (item != null && !item.trim().isEmpty()) {

HttpSession hs = request.getSession();

List<String> Cart = (List<String>) hs.getAttribute("Cart");

if (Cart == null ) {

Cart = new ArrayList<>();

}

Cart.add(item);

hs.setAttribute("Cart", Cart);

}

response.sendRedirect("Cart.jsp");

%>

</body>

</html>

**ViewCart.jsp**

<%@page contentType="text/html" pageEncoding="UTF-8"%>

<!DOCTYPE html>

<html>

<head>

<meta http-equiv="Content-Type" content="text/html; charset=UTF-8">

<title>JSP Page</title>

</head>

<body>

<%@ page language="java" import="java.util.List, java.util.ArrayList" session="true" %>

<%

HttpSession hs = request.getSession();

List<String> Cart = (List<String>) hs.getAttribute("Cart");

if (Cart == null || Cart.isEmpty()) {

out.println("<p>Your cart is empty.</p>");

} else {

out.println("<ul>");

int j=1;

out.println("<table Border=1>"

+"<th>Item No </th>"

+"<th>Item Name</th>"

+"<th>Edit </th>");

for (String i : Cart)

{

out.println("<tr>"

+"<td>"+ j+" </td>"

+"<td>"+ i + "</td>"

+"<td><a href='Remove.jsp?item=" + i + "'>Remove</a>"

+"</tr>");

j++;

}

out.println("</table>");

out.println("</ul>");

}

%>

<br>

<a href="Cart.jsp">Back to Shopping</a>

</body>

</html>

**Remove.jsp**

<%@page contentType="text/html" pageEncoding="UTF-8"%>

<!DOCTYPE html>

<html>

<head>

<meta http-equiv="Content-Type" content="text/html; charset=UTF-8">

<title>Remove</title>

</head>

<body>

<%@ page language="java" import="java.util.List, java.util.ArrayList" session="true" %>

<%

String item = request.getParameter("item");

if (item != null )

{

HttpSession hs = request.getSession();

List<String> Cart = (List<String>) hs.getAttribute("Cart");

if (Cart != null)

{

Cart.remove(item);

session.setAttribute("Cart", Cart);

}

}

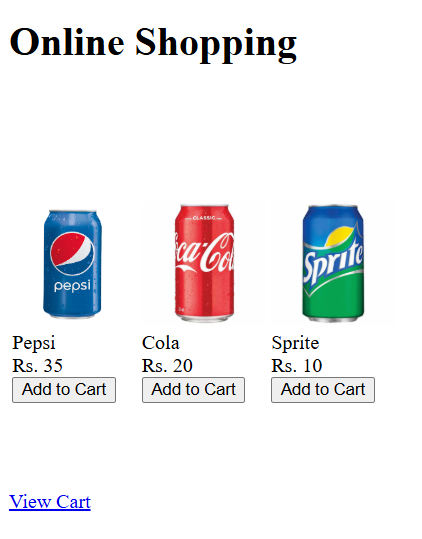
response.sendRedirect("ViewCart.jsp");

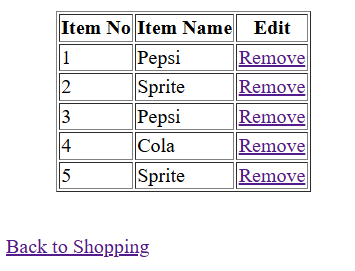
%>

</body>

</html>

**OUTPUT:**

****

****

**\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\***

**PA8: 8. Write a java Servlet program to Download a file and display it on the screen (A link has to be provided in HTML, when the link is clicked corresponding file has to be displayed on screen).**

**NAME: NISHANTH ROLL NO:22536**

**CLASS: III BCA**

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

**index.html**

<!DOCTYPE html>

<html>

<head>

<title>TODO supply a title</title>

<meta charset="UTF-8">

<meta name="viewport" content="width=device-width, initial-scale=1.0">

</head>

<body>

<a href= “FileDownloaders?fname=JAVA.txt">Download File</a>

</body>

</html>

**FileDownloaders.java**

import java.io.FileInputStream;

import java.io.IOException;

import java.io.OutputStream;

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 = {"/FileDownloaders"})

public class FileDownloaders extends HttpServlet {

protected void processRequest(HttpServletRequest request, HttpServletResponse response)

throws ServletException, IOException {

response.setContentType("text/html;charset=UTF-8");

response.setContentType("text/plaintext");

String fname = request.getParameter("fname");

response.setContentType("text/plaintext");

response.setHeader("Content-Disposition", "attachment;filename=\"" + fname + "\"");

try (OutputStream os = response.getOutputStream()) {

FileInputStream file = new FileInputStream("

int i = 0;

while ((i = file.read()) != -1) {

os.write(i);

}

}

file.close();

os.close();

}

// <editor-fold defaultstate="collapsed" desc="HttpServlet methods. Click on the + sign on the left to edit the code.">

@Override

protected void doGet(HttpServletRequest request, HttpServletResponse response)

throws ServletException, IOException {

processRequest(request, response);

}

@Override

protected void doPost(HttpServletRequest request, HttpServletResponse response)

throws ServletException, IOException {

processRequest(request, response);

}

@Override

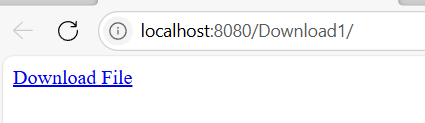
public String getServletInfo() {

return "Short description";

}// </editor-fold>

}

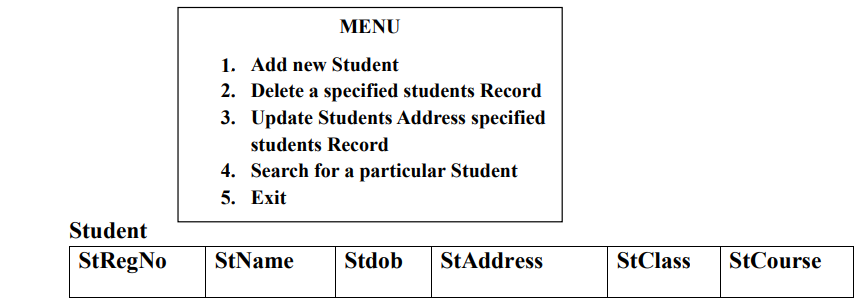
**OUTPUT:**

****

****

**\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\***

**PB1: Write a menu driven JDBC program to perform basic operations with Student Table.**



**NAME: NISHANTH ROLL NO:22536**

**CLASS: III BCA**

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

import java.util.Scanner;

import java.io.\*;

import java.sql.\*;

import java.sql.Date;

import java.util.logging.Level;

import java.util.logging.Logger;

public class PARTB1 {

public static void main(String[] args) {

Scanner sc=new Scanner(System.in);

String regno;

String sname;

String sadd,sclass,scourse;

String dob;

int choice;

try {

Class.forName("org.apache.derby.jdbc.ClientDriver");

Connection con=DriverManager.getConnection("jdbc:derby://localhost:1527/Nishanth","Nishanth","Nishu@123");

Statement stmt=con.createStatement();

ResultSet rs;

do{

System.out.println("MENU");

System.out.println("---------------");

System.out.println("1.Add Student");

System.out.println("2.Delete Student");

System.out.println("3.Update Student");

System.out.println("4.search Student");

System.out.println("5.Exit");

System.out.println("-----------------");

System.out.println("Enter your choice:");

choice=sc.nextInt();

switch(choice){

case 1:

System.out.println("---Enter Student Details---");

System.out.println("Reg NO:");

regno=sc.next();

System.out.println("Name:");

sname=sc.next();

System.out.println("DOB[yyyy-dd-mm]:");

dob=sc.next();

System.out.println("Address:");

sadd=sc.next();

System.out.println("Class:");

sclass=sc.next();

System.out.println("Course:");

scourse=sc.next();

String sql="INSERT INTO NISHANTH.STUDENT (STREGNO, STNAME, STDOB, STADDRESS, STCLASS, STCOURSE)VALUES ('"+regno+"', '"+sname+"', '"+dob.toString()+"', '"+sadd+"', '"+sclass+"', '"+scourse+"')";

int result=stmt.executeUpdate(sql);

if(result==1)

{

System.out.println("Student details are saved");

}

else

{

System.out.println("error while saving student details");

}

break;

case 2:

System.out.print("Enter Student Regno:");

String regno1=sc.next();

sql="SELECT COUNT(\*) FROM NISHANTH.STUDENT WHERE STREGNO='"+regno1+"'";

rs=stmt.executeQuery(sql);

rs.next();

if(rs.getInt(1)==1)

{

sql="DELETE FROM NISHANTH.STUDENT WHERE STREGNO='"+regno1+"'";

int res=stmt.executeUpdate(sql);

if(res==1)

{

System.out.println("Student record is deleted");

}

else

{

System.out.println("record not deleted");

}

}

else

{

System.out.println("Student record not found");

}

break;

case 3:

System.out.println("Enter student regno:");

regno1=sc.next();

sql="SELECT COUNT(\*) FROM NISHANTH.STUDENT WHERE STREGNO='"+regno1+"'";

rs=stmt.executeQuery(sql);

rs.next();

if(rs.getInt(1)==1)

{

String address;

System.out.println("Enter new address:");

address=sc.next();

sql="UPDATE NISHANTH.STUDENT SET STADDRESS='"+ address + "' where STREGNO='"+regno1+"'" ;

int res=stmt.executeUpdate(sql);

if(res==1)

{

System.out.println("Student record is updated");

}

else

{

System.out.println("record notupdated");

}

}

else{

System.out.println("Student record not found");

}

break;

case 4:

System.out.print("Enter Student Regno:");

regno=sc.next();

sql="SELECT COUNT(\*) FROM NISHANTH.STUDENT WHERE STREGNO='"+regno+"'";

rs=stmt.executeQuery(sql);

rs.next();

if(rs.getInt(1)==1)

{

sql="SELECT \* FROM NISHANTH.STUDENT where STREGNO='"+regno+"'" ;

rs=stmt.executeQuery(sql);

rs.next();

System.out.println("Student Details:");

System.out.println("Reg No:"+rs.getString(1));

System.out.println("Name:"+rs.getString(2));

System.out.println("Date of Birth:"+rs.getString(3));

System.out.println("Address:"+rs.getString(4));

System.out.println("Class:"+rs.getString(5));

System.out.println("Course:"+rs.getString(6));

}

else

{

System.out.println("Student record not found");

}

break;

case 5:

stmt.close();;

con.close();

return;

}

}while(true);

// TODO code application logic here

} catch (ClassNotFoundException ex) {

System.out.println(ex.toString());

} catch (SQLException ex) {

System.out.println(ex.toString());

Logger.getLogger(PARTB1.class.getName()).log(Level.SEVERE, null, ex);

}

}

}

**OUTPUT:**

**1.Add New Student**

MENU

---------------

1.Add Student

2.Delete Student

3.Update Student

4.search Student

5.Exit

-----------------

Enter your choice:

1

---Enter Student Details---

Reg NO:

101

Name:

Nishu

DOB[yyyy-dd-mm]:

2004-14-09

Address:

savya

Class:

BCA

Course:

CS

Student details are saved

MENU

---------------

1.Add Student

2.Delete Student

3.Update Student

4.search Student

5.Exit

-----------------

Enter your choice:

1

---Enter Student Details---

Reg NO:

103

Name:

Rakshith

DOB[yyyy-dd-mm]:

2004-11-07

Address:

Naravi

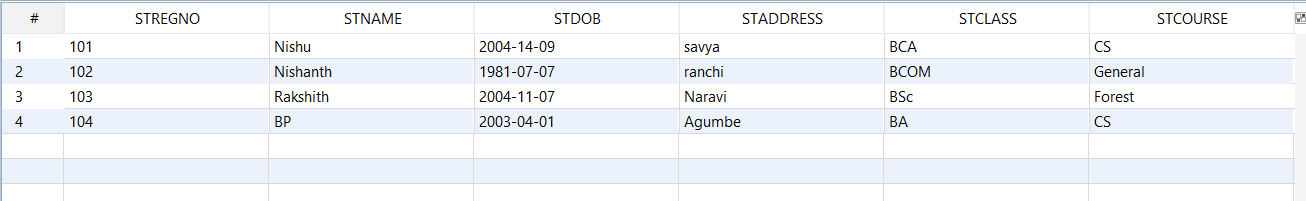
Class:

BSc

Course:

Forest

Student details are saved

****

**2.Delete a specific record**

MENU

---------------

1.Add Student

2.Delete Student

3.Update Student

4.search Student

5.Exit

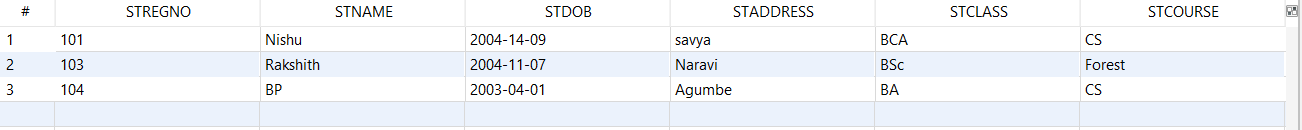
-----------------

Enter your choice:

2

Enter Student Regno:102

Student record is deleted



**3.Update Student Address Specified Student Record**

MENU

---------------

1.Add Student

2.Delete Student

3.Update Student

4.search Student

5.Exit

-----------------

Enter your choice:

3

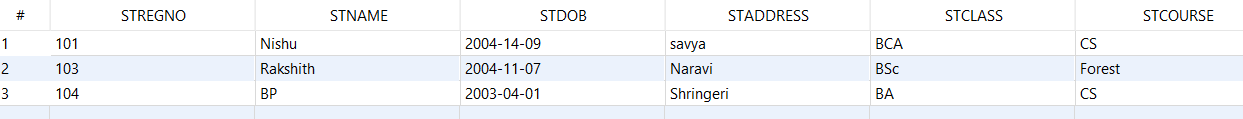
Enter student regno:

104

Enter new address:

Shringeri

Student record is updated

****

**4.Search for a Student**

MENU

---------------

1.Add Student

2.Delete Student

3.Update Student

4.search Student

5.Exit

-----------------

Enter your choice:

4

Enter Student Regno:101

Student Details:

Reg No:101

Name:Nishu

Date of Birth:2004-14-09

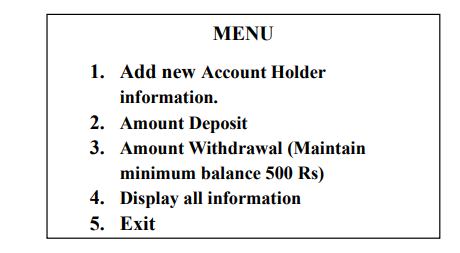
Address:savya

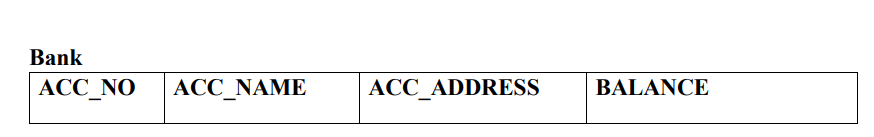
Class:BCA

**Course:CS**

**\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\***

**PB2: Write a menu driven JDBC program to perform basic operations with Bank Table.**





**NAME: NISHANTH ROLL NO:22536**

**CLASS: III BCA**

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

package bank;

import java.io.\*;

import java.sql.\*;

import java.util.\*;

import java.util.Scanner;

import java.util.logging.Level;

import java.util.logging.Logger;

public class Bank {

public static void main(String[] args)throws SQLException {

Scanner in=new Scanner(System.in);

Connection con;

int acc\_no;

String name;

Statement stmt;

ResultSet result;

String sql="";

try {

Class.forName("org.apache.derby.jdbc.ClientDriver");

con=DriverManager.getConnection("jdbc:derby://localhost:1527/Nishanth","Nishanth","Nishu@123");

stmt=con.createStatement();

while(true){

System.out.println("\n\*\*transaction Menu\*\*");

System.out.println("\nDeposite");

System.out.println("\n1.Add Account");

System.out.println("\n2.Deposite");

System.out.println("\n3.Withdraw");

System.out.println("\n4.Display");

System.out.println("\n5.Exit");

System.out.println("Enter your choice:");

int ch=in.nextInt();

switch(ch){

case 1:

System.out.println("Enter the Account holder name:");

name=in.next();

System.out.println("Enter the Account number:");

acc\_no=in.nextInt();

System.out.println("Enter the adress:");

String address=in.next();

System.out.println("Enter the Account Balance:");

float bal=in.nextFloat();

sql="INSERT INTO NISHANTH.BANK (ACCNO, ACC\_NAME, ACC\_ADDRESS, BALANCE)VALUES ('"+acc\_no+"','"+name+"','"+address+"',"+bal+")";

stmt.executeUpdate(sql);

break;

case 2:

System.out.println("Enter the account number:");

acc\_no=in.nextInt();

System.out.print("\n Enter the amount to be deposited:");

float d=in.nextFloat();

if(d<=0)

System.out.print("\nEnter proper amount.");

else

sql="UPDATE BANK SET BALANCE=BALANCE + "+d + " Where ACCNO='"+acc\_no+"'";

stmt.executeUpdate(sql);

break;

case 3:

System.out.println("Enter the account number:");

acc\_no=in.nextInt();

System.out.print("\n Enter the amount to be withdrawn:");

float w=in.nextFloat();

if(w<=0)

System.out.print("\n Enter proper amount.");

else{

sql="UPDATE BANK SET BALANCE=BALANCE - "+w + " Where ACCNO='"+acc\_no+"' and BALANCE-"+w+" >500";

int r=stmt.executeUpdate(sql);

if(r==1)

System.out.println("Updated successfully!!");

else

System.out.println("Cannot withdraw the amount-LOW BALANCE!!");

}

break;

case 4:

result=stmt.executeQuery("SELECT \* FROM BANK");

System.out.println("\n\n Acc\_no \t\t Name \t\t Address \t\t balance ");

System.out.println("----------------------------------------------------------------");

while(result.next())

{

System.out.print(result.getString(1)+"\t\t");

System.out.print(result.getString(2)+"\t\t");

System.out.print(result.getString(3)+"\t\t");

System.out.print((int)result.getInt(4)+"\t\t");

System.out.print("\n");

}

break;

case 5:

System.out.print("\n Quiting ....Thank You!!");

System.exit(0);

break;

default:

System.out.print("\n Enter proper choice..");

}

System.out.print("\n -------------------------------------------------");

}

} catch (ClassNotFoundException se) {

System.out.println("Exception occured while getting connection!!!");

Logger.getLogger(Bank.class.getName()).log(Level.SEVERE, null, se);

}

}

}

**OUTPUT:**

**1.Add New Account Holder information**

\*\*transaction Menu\*\*

1.Add Account

2.Deposite

3.Withdraw

4.Display

5.Exit

Enter your choice:

1

Enter the Account holder name:

Nishanth

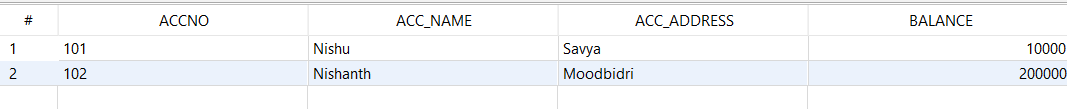
Enter the Account number:

102

Enter the adress:

Moodbidri

Enter the Account Balance:

200000

**2.Amount Deposite**

\*\*transaction Menu\*\*

1.Add Account

2.Deposite

3.Withdraw

4.Display

5.Exit

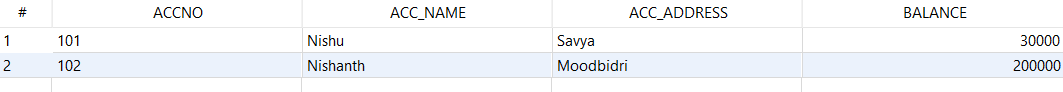
Enter your choice:

2

Enter the account number:

101

Enter the amount to be deposited:20000



**3.Amount Withdrawal**

\*\*transaction Menu\*\*

1.Add Account

2.Deposite

3.Withdraw

4.Display

5.Exit

Enter your choice:

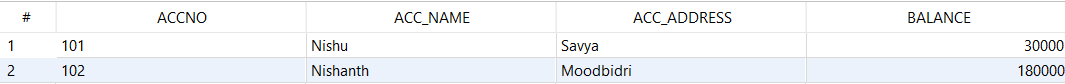
3

Enter the account number:

102

Enter the amount to be withdrawn:20000

Updated successfully!!



**4.Display All Information**

\*\*transaction Menu\*\*

1.Add Account

2.Deposite

3.Withdraw

4.Display

5.Exit

Enter your choice:

4

Acc\_no Name Address Balance

------------------------------------------------------------------------------

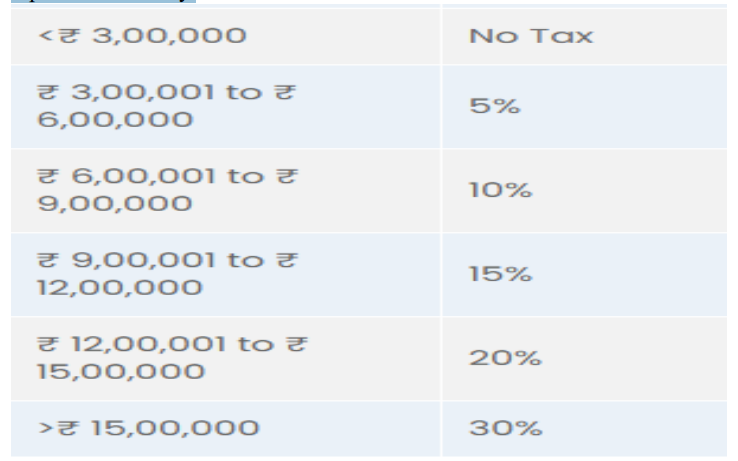
101 Nishu Savya 30000

102 Nishanth Moodbidri 180000

----------------------------------------------------------------------------

**\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\***

**PB3: Write a Java class called Tax with methods for calculating Income Tax. Have this class as a servant and create a server program and register in the rmiregistry. Write a client program to invoke these remote methods of the servant and do the calculations. Accept inputs interactively.**

****

**NAME: NISHANTH ROLL NO:22536**

**CLASS: III BCA**

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

**Tax.java**

package incometax;

import java.rmi.\*;

public interface Tax extends Remote {

double calTax(double a) throws RemoteException;

}

**TaxImpl.java**

package incometax;

import java.rmi.server.\*;

import java.rmi.\*;

public class TaxImpl extends UnicastRemoteObject implements Tax {

public TaxImpl() throws RemoteException {

}

public double calTax(double income) throws RemoteException {

double t;

if (income <= 300000) {

t = 0.0;

} else if (income > 300001 && income <= 600000) {

t = 0.05f \* (income - 300000);

} else if (income > 600001 && income <= 900000) {

t = 0.10f \* (income - 600000);

} else if (income > 900001 && income <= 1200000) {

t = 0.15f \* (income - 900000);

} else if (income > 1200001 && income <= 1500000) {

t = 0.20f \* (income - 1200000);

} else {

t = 0.30f \* (income - 1500000);

}

return t;

}

}

**TaxServer.java**

package incometax;

import java.rmi.\*;

import java.rmi.registry.\*;

import java.util.logging.Level;

import java.util.logging.Logger;

public class TaxServer {

public static void main(String args[]) {

try {

TaxImpl timpl = new TaxImpl();

Registry reg = LocateRegistry.createRegistry(18888);

reg.rebind("TaxServer", timpl);

System.out.println("Server is running.......");

} catch (RemoteException e) {

System.out.println("Exception in server!! .....");

}

}

}

**TaxClient.java**

package incometax;

import java.rmi.AccessException;

import java.rmi.NotBoundException;

import java.rmi.RemoteException;

import java.util.\*;

import java.rmi.registry.\*;

import java.util.logging.Level;

import java.util.logging.Logger;

public class TaxClient {

public static void main(String args[]) {

try {

double d;

Scanner sc = new Scanner(System.in);

Registry reg = LocateRegistry.getRegistry(18888);

Tax t = (Tax) reg.lookup("TaxServer");

System.out.println("Enter the Income:");

d = sc.nextDouble();

System.out.println("The calculated tax amount is=" + t.calTax(d));

} catch (RemoteException e) {

System.out.println("Exception in Client.....");

} catch (NotBoundException ex) {

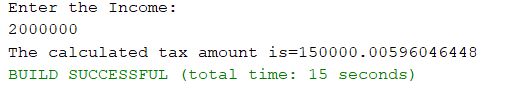
Logger.getLogger(TaxClient.class.getName()).log(Level.SEVERE, null, ex);

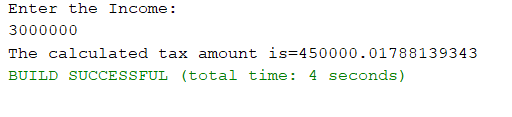
}

}

}

**OUTPUT:**

****

****

**\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\***

**PB4: Write a Java class called SimpleInterest with methods for calculating simple interest. Have this class as a servant and create a server program and register in the rmiregistry. Write a client program to invoke these remote methods of the servant and do the calculations. Accept inputs at command prompt**

**NAME: NISHANTH ROLL NO:22536**

**CLASS: III BCA**

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

**SimpleInterest.java(interface)**

package simpleinterest;

import java.rmi.\*;

public interface SimpleInterest extends Remote {

double computeSI(double p, double t, double r) throws RemoteException;

}

**SimpleInterestImp1.java**

package simpleinterest;

import java.rmi.\*;

import java.rmi.server.\*;

public class SimpleInterestImp1 extends UnicastRemoteObject implements SimpleInterest {

public SimpleInterestImp1() throws RemoteException {

super();

}

@Override

public double computeSI(double p, double t, double r) throws RemoteException {

return (p \* t \* r / 100);

}

}

**SimpleInterestServer.java**

package simpleinterest;

import java.rmi.\*;

import java.rmi.registry.\*;

public class SimpleInterestServer {

public static void main(String[] args) throws RemoteException, AlreadyBoundException {

SimpleInterestImp1 si = new SimpleInterestImp1();

Registry reg = LocateRegistry.createRegistry(18888);

reg.bind("SI", si);

System.out.println("Server is started.....");

}

}

**SimpleInterestClient.java**

package simpleinterest;

import java.rmi.registry.\*;

import java.rmi.\*;

import java.util.Scanner;

public class SimpleInterestClient {

public static void main(String[] args) throws RemoteException, NotBoundException {

Registry reg = LocateRegistry.getRegistry(18888);

SimpleInterest si = (SimpleInterest) reg.lookup("SI");

Scanner sc = new Scanner(System.in);

double p, t, r;

String ans = "n";

do {

System.out.println("Simple Interest Calculation");

System.out.print("Principal: ");

p = sc.nextDouble();

System.out.print(" Time: ");

t = sc.nextDouble();

System.out.print("Rate: ");

r = sc.nextDouble();

System.out.println("Simple Interest is " + si.computeSI(p, t, r));

System.out.println("Do you want to continue[y/n]?");

sc.nextLine();

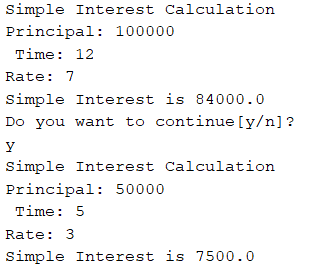
ans = sc.nextLine();

} while (ans.toLowerCase().charAt(0) == 'y');

}

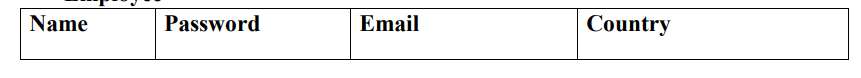
}

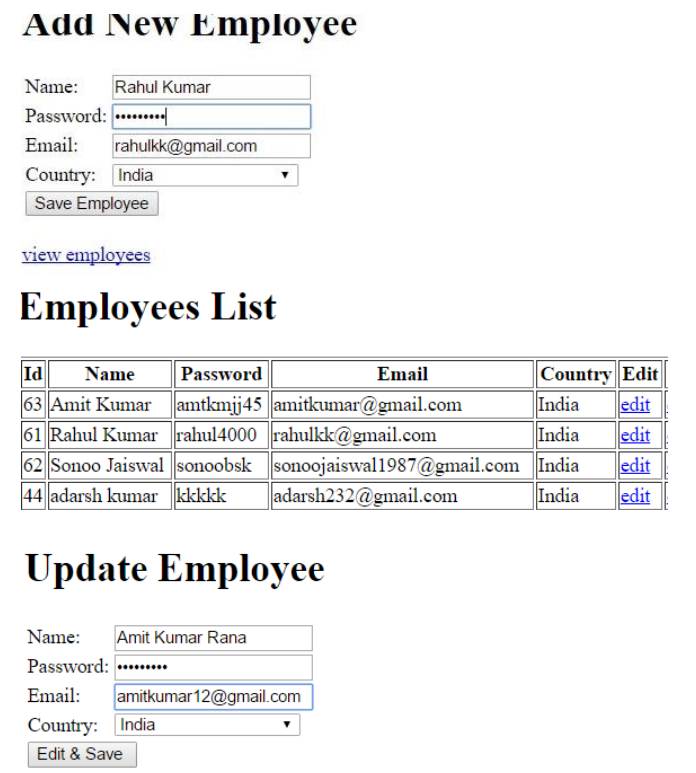
**OUTPUT:**



**\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\***

**PB5: Write a Servlet Program to perform Insert, update and View operations on Employee Table**

****

****

**NAME: NISHANTH ROLL NO:22536**

**CLASS: III BCA**

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

**index.html**

<!DOCTYPE html>

<html>

<body>

<h1>Add Employee</h1>

<form method="POST" action="AddEmployee">

<table>

<tr>

<td>Name:</td>

<td><input type="text" name="ename"></td>

</tr>

<tr>

<td>Password:</td>

<td><input type="password" name="password"></td>

</tr>

<tr>

<td>Email:</td>

<td><input type="email" name="email"></td>

</tr>

<tr>

<td>Country:</td>

<td><select name="country">

<option>India</option>

<option>Nepal</option>

<option>China</option>

<option>Sri Lanka</option>

</select></td>

</tr>

<tr>

<td></td>

<td colspan="2"><input type="submit" name="submit" value="Save Employee"></td>

</tr>

</table>

</form>

<a href="ViewEmployee">View Employee</a>;

</body>

</html>

**AddEmployee.java**

package com;

import java.io.IOException;

import java.io.PrintWriter;

import java.sql.Connection;

import java.sql.DriverManager;

import java.sql.SQLException;

import java.sql.Statement;

import java.util.logging.Level;

import java.util.logging.Logger;

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 = {"/AddEmployee"})

public class AddEmployee extends HttpServlet {

protected void processRequest(HttpServletRequest request, HttpServletResponse response)

throws ServletException, IOException, ClassNotFoundException, SQLException {

response.setContentType("text/html;charset=UTF-8");

try (PrintWriter out = response.getWriter()) {

String ename = request.getParameter("ename");

String password = request.getParameter("password");

String email = request.getParameter("email");

String country = request.getParameter("country");

Class.forName("org.apache.derby.jdbc.ClientDriver");

Connection con = DriverManager.getConnection("jdbc:derby://localhost:1527/Nishanth", "nishu", "1234");

Statement stmt = con.createStatement();

String sql = "INSERT INTO NISHU.EMPLOYEE(ENAME,PASSWORD,EMAIL,COUNTRY)VALUES('" + ename + "','" + password + "','" + email + "','" + country + "') ";

int rcount = stmt.executeUpdate(sql);

stmt.close();

con.close();

out.println("<!DOCTYPE html>");

out.println("<html>");

out.println("<head>");

out.println("<title>Servlet AddEmployee</title>");

out.println("</head>");

out.println("<body>");

if (rcount == 1) {

out.println("<h1>Record Saved</h1>");

} else {

out.println("<h1>There was an error</h1>");

}

out.println("<a href=\"index.html\">Home</a>");

out.println("<h1>Servlet AddEmployee at " + request.getContextPath() + "</h1>");

out.println("</body>");

out.println("</html>");

}

}

@Override

protected void doGet(HttpServletRequest request, HttpServletResponse response)

throws ServletException, IOException {

try {

processRequest(request, response);

} catch (ClassNotFoundException | SQLException ex) {

Logger.getLogger(AddEmployee.class.getName()).log(Level.SEVERE, null, ex);

}

}

@Override

protected void doPost(HttpServletRequest request, HttpServletResponse response)

throws ServletException, IOException {

try {

processRequest(request, response);

} catch (ClassNotFoundException | SQLException ex) {

Logger.getLogger(AddEmployee.class.getName()).log(Level.SEVERE, null, ex);

}

}

@Override

public String getServletInfo() {

return "Short description";

}// </editor-fold>

}

**ViewEmployee.java**

package com;

import java.io.IOException;

import java.io.PrintWriter;

import java.sql.Connection;

import java.sql.DriverManager;

import java.sql.ResultSet;

import java.sql.SQLException;

import java.sql.Statement;

import java.util.logging.Level;

import java.util.logging.Logger;

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 = {"/ViewEmployee"})

public class ViewEmployee extends HttpServlet {

/\*\*

\* Processes requests for both HTTP <code>GET</code> and <code>POST</code>

protected void processRequest(HttpServletRequest request, HttpServletResponse response)

throws ServletException, IOException, ClassNotFoundException, SQLException {

response.setContentType("text/html;charset=UTF-8");

try (PrintWriter out = response.getWriter()) {

Class.forName("org.apache.derby.jdbc.ClientDriver");

Connection con = DriverManager.getConnection("jdbc:derby://localhost:1527/Nishanth", "nishu", "1234");

Statement stmt = con.createStatement();

String sql = "SELECT \* FROM NISHU.EMPLOYEE";

ResultSet rs = stmt.executeQuery(sql);

/\* TODO output your page here. You may use following sample code. \*/

out.println("<!DOCTYPE html>");

out.println("<html>");

out.println("<head>");

out.println("<title>Servlet ViewEmployee</title>");

out.println("</head>");

out.println("<body>");

out.println(" <h1>Employee List</h1>");

out.println(" <table border=\"2\">");

out.println(" <tr>");

out.println(" <th>id</th>");

out.println(" <th>Name</th>");

out.println(" <th>Password</th>");

out.println(" <th>Country</th>");

out.println(" </tr>");

while (rs.next()) {

out.println(" <tr>");

out.println(" <td>" + rs.getString("ID") + "</td>");

out.println(" <td>" + rs.getString("ENAME") + "</td>");

out.println(" <td>" + rs.getString("PASSWORD") + "</td>");

out.println(" <td>" + rs.getString("EMAIL") + "</td>");

out.println(" <td>" + rs.getString("COUNTRY") + "</td>");

out.println(" <td> <a href=\"UpdateEmployee?id=" + rs.getString("ID") + "\">Edit</a></td>");

out.println(" </tr>");

}

out.println(" </table>");

out.println(" <a href=\"index.html\"> Add new record</a>\n" + "");

out.println("<h1>Servlet ViewEmployee at " + request.getContextPath() + "</h1>");

out.println("</body>");

out.println("</html>");

}

}

@Override

protected void doGet(HttpServletRequest request, HttpServletResponse response)

throws ServletException, IOException {

try {

processRequest(request, response);

} catch (ClassNotFoundException | SQLException ex) {

Logger.getLogger(ViewEmployee.class.getName()).log(Level.SEVERE, null, ex);

}

}

@Override

protected void doPost(HttpServletRequest request, HttpServletResponse response)

throws ServletException, IOException {

try {

processRequest(request, response);

} catch (ClassNotFoundException | SQLException ex) {

Logger.getLogger(ViewEmployee.class.getName()).log(Level.SEVERE, null, ex);

}

}

@Override

public String getServletInfo() {

return "Short description";

}// </editor-fold>

}

**UpdateEmployee.java**

package com;

import java.io.IOException;

import java.io.PrintWriter;

import java.sql.Connection;

import java.sql.DriverManager;

import java.sql.ResultSet;

import java.sql.SQLException;

import java.sql.Statement;

import java.util.logging.Level;

import java.util.logging.Logger;

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(name = "UpdateEmployee", urlPatterns = {"/UpdateEmployee"})

public class UpdateEmployee extends HttpServlet {

protected void processRequest(HttpServletRequest request, HttpServletResponse response)

throws ServletException, IOException, ClassNotFoundException, SQLException {

response.setContentType("text/html;charset=UTF-8");

try (PrintWriter out = response.getWriter()) {

int id = Integer.parseInt(request.getParameter("id"));

Class.forName("org.apache.derby.jdbc.ClientDriver");

Connection con = DriverManager.getConnection("jdbc:derby://localhost:1527/Nishanth", "nishu", "1234");

Statement stmt = con.createStatement();

String sql = "SELECT \* FROM NISHU.EMPLOYEE WHERE ID=" + id;

ResultSet rs = stmt.executeQuery(sql);

rs.next();

/\* TODO output your page here. You may use following sample code. \*/

out.println("<!DOCTYPE html>");

out.println("<html>");

out.println("<head>");

out.println("<title>Servlet UpdateEmployee</title>");

out.println("</head>");

out.println("<body>");

out.println("<h1>Servlet UpdateEmployee at " + request.getContextPath() + "</h1>");

out.println("<H1>Update Employee</H1>");

out.println(" <form method=\"POST\" action=\"SaveEmployee\">");

out.println("<input type=\"hidden\" name=\"id\" value=\"" + id + "\">");

out.println(" <table>");

out.println(" <tr>");

out.println(" <td>Name:</td>");

out.println(" <td><input type=\"text\" name=\"ename\" value="+ rs.getString("ENAME") + "></td>");

out.println(" </tr>");

out.println(" <tr>");

out.println(" <td>Password:</td>");

out.println(" <td><input type=\"password\" name=\"password\" value="+ rs.getString("PASSWORD") + "></td>");

out.println(" </tr>");

out.println(" <tr>");

out.println(" <td>Email:</td>");

out.println(" <td><input type=\"email\" name=\"email\" value="+ rs.getString("EMAIL") + "></td>");

out.println(" </tr>");

out.println(" <tr>");

out.println(" <td>Country:</td>");

out.println(" <td><select name=\"country\" >");

String sel = rs.getString("COUNTRY").equals("India") ? "Selected" : "";

out.println(" <option " + sel + ">India</option>");

sel = rs.getString("COUNTRY").equals("Nepal") ? "Selected" : "";

out.println(" <option " + sel + ">Nepal</option>");

sel = rs.getString("COUNTRY").equals("China") ? "Selected" : "";

out.println(" <option " + sel + ">China</option>");

sel = rs.getString("COUNTRY").equals("Sri Lanka") ? "Selected" : "";

out.println(" <option " + sel + ">Sri Lanka</option>");

out.println(" </select></td>");

out.println(" </tr>");

out.println(" <tr>");

out.println(" <td></td>");

out.println(" <td colspan=\"2\"><input type=\"submit\" name=\"submit\" value=\"Edit Emloyee\"></td>");

out.println(" </tr>");

out.println(" </table>");

out.println(" ");

out.println(" </form>");

out.println(" <a href=\"ViewEmployee\">View Employees</a>;");

out.println("</body>");

out.println("</html>");

out.println("</body>");

out.println("</html>");

}

}

// <editor-fold defaultstate="collapsed" desc="HttpServlet methods. Click on the + sign on the left to edit the code.">

@Override

protected void doGet(HttpServletRequest request, HttpServletResponse response)

throws ServletException, IOException {

try {

processRequest(request, response);

} catch (ClassNotFoundException | SQLException ex) {

Logger.getLogger(UpdateEmployee.class.getName()).log(Level.SEVERE, null, ex);

}

}

@Override

protected void doPost(HttpServletRequest request, HttpServletResponse response)

throws ServletException, IOException {

try {

processRequest(request, response);

} catch (ClassNotFoundException | SQLException ex) {

Logger.getLogger(UpdateEmployee.class.getName()).log(Level.SEVERE, null, ex);

}

}

@Override

public String getServletInfo() {

return "Short description";

}// </editor-fold>

}

**SaveEmployee.java**

package com;

import java.io.IOException;

import java.io.PrintWriter;

import java.sql.Connection;

import java.sql.DriverManager;

import java.sql.SQLException;

import java.sql.Statement;

import java.util.logging.Level;

import java.util.logging.Logger;

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(name = "SaveEmployee", urlPatterns = {"/SaveEmployee"})

public class SaveEmployee extends HttpServlet {

protected void processRequest(HttpServletRequest request, HttpServletResponse response)

throws ServletException, IOException, ClassNotFoundException, SQLException {

response.setContentType("text/html;charset=UTF-8");

try (PrintWriter out = response.getWriter()) {

String id = request.getParameter("id");

String ename = request.getParameter("ename");

String password = request.getParameter("password");

String email = request.getParameter("email");

String country = request.getParameter("country");

Class.forName("org.apache.derby.jdbc.ClientDriver");

Connection con = DriverManager.getConnection("jdbc:derby://localhost:1527/Nishanth", "nishu", "1234");

Statement stmt = con.createStatement();

String sql = "UPDATE NISHU.EMPLOYEE SET ENAME='" + ename + "',PASSWORD='" + password + "',EMAIL='" + email + "',COUNTRY='" + country + "' WHERE ID=" + id;

int rcount = stmt.executeUpdate(sql);

stmt.close();

con.close();

/\* TODO output your page here. You may use following sample code. \*/

out.println("<!DOCTYPE html>");

out.println("<html>");

out.println("<head>");

out.println("<title>Servlet SaveEmployee</title>");

out.println("</head>");

out.println("<body>");

out.println("<h1>Servlet SaveEmployee at " + request.getContextPath() + "</h1>");

out.println("</body>");

out.println("</html>");

}

}

@Override

protected void doGet(HttpServletRequest request, HttpServletResponse response)

throws ServletException, IOException {

try {

processRequest(request, response);

} catch (ClassNotFoundException | SQLException ex) {

Logger.getLogger(SaveEmployee.class.getName()).log(Level.SEVERE, null, ex);

}

}

@Override

protected void doPost(HttpServletRequest request, HttpServletResponse response)

throws ServletException, IOException {

try {

processRequest(request, response);

} catch (ClassNotFoundException | SQLException ex) {

Logger.getLogger(SaveEmployee.class.getName()).log(Level.SEVERE, null, ex);

}

}

@Override

public String getServletInfo() {

return "Short description";

}// </editor-fold>

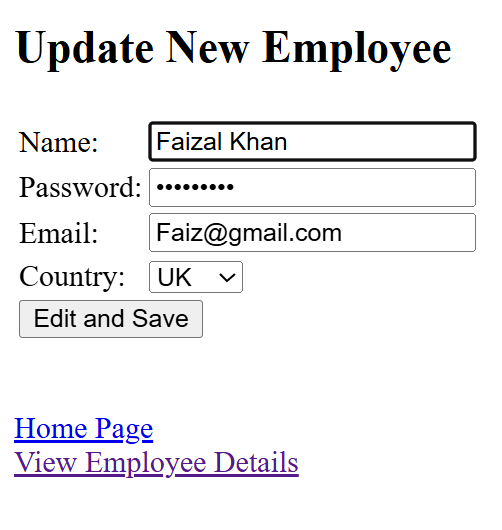
}

**OUTPUT:**



****





**\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\***

**PB6: Write a java JSP program to get student information through a HTML and create a JAVA Bean Class, populate Bean and Display the same information through another JSP**

**NAME: NISHANTH ROLL NO:22536**

**CLASS: III BCA**

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

**index.html**

<!DOCTYPE html>

<html>

<head>

<title>Student info</title>

<meta charset="UTF-8">

<meta name="viewport" content="width=device-width, initial-scale=1.0">

</head>

<body>

<h2>Enter student details</h2>

<form method="POST" action="Firstpage.jsp">

<table>

<tr>

<td>Register No:</td>

<td><input type="text" name="regno"></td>

</tr>

<tr>

<td>Name:</td>

<td><input type="text" name="sname"></td>

</tr>

<tr>

<td>Course:</td>

<td><input type="text" name="course"></td>

</tr>

<tr>

<td>Semester:</td>

<td><input type="text" name="sem"></td>

</tr>

<tr>

<td></td>

<td><input type="submit" name="subBtn" value="Register"></td>

</tr>

</table>

</form>

</body>

</html>

**Student.java**

package bean;

import java.io.Serializable;

public class Student implements Serializable {

private String regNo;

private String name;

private String course;

private String sem;

public Student() {

}

public String getRegNo() {

return regNo;

}

public void setRegNo(String regNo) {

this.regNo = regNo;

}

public String getName() {

return name;

}

public void setName(String name) {

this.name = name;

}

public String getCourse() {

return course;

}

public void setCourse(String course) {

this.course = course;

}

public String getSem() {

return sem;

}

public void setSem(String sem) {

this.sem = sem;

}

}

**Firstpage.jsp**

<%@page import="bean.Student" %>

<%@page contentType="text/html" pageEncoding="UTF-8"%>

<!DOCTYPE html>

<html>

<head>

<meta http-equiv="Content-Type" content="text/html; charset=UTF-8">

<title>first JSP Page</title>

</head>

<body>

<h1>Student Details are saved</h1>

<jsp:useBean id="std" scope="session" class="bean.Student" >

<jsp:setProperty name="std" property="regNo" value="${param.regno}"/>

<jsp:setProperty name="std" property="name" value="${param.sname}"/>

<jsp:setProperty name="std" property="course" value="${param.course}"/>

<jsp:setProperty name="std" property="sem" value="${param.sem}"/>

<a href="../src/java/bean/Student.java"></a>

</jsp:useBean>

<h2><a href="Secondpage.jsp">View Student details </a></h2>

</body>

</html>

**Secondpage.jsp**

<%@page contentType="text/html" pageEncoding="UTF-8"%>

<html>

<head>

<meta http-equiv="Content-Type" content="text/html; charset=UTF-8">

<title>Second JSP Page</title>

</head>

<body>

<h1>Student Details are</h1>

<table>

<tr>

<td>Register No:</td>

<td>${std.regNo}</td>

</tr>

<tr>

<td>Name:</td>

<td>${std.name}</td>

</tr>

<tr>

<td>Course:</td>

<td>${std.course}</td>

</tr>

<tr>

<td>Semester:</td>

<td>${std.sem}</td>

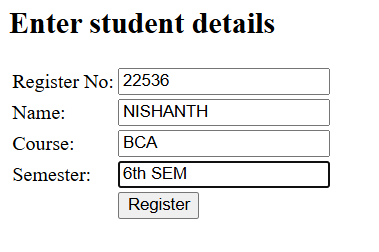
</tr>

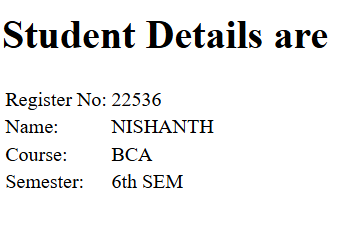
</table>

</body>

</html>

**OUTPUT:**



**\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\***

**PB7: Write a menu driven program to create a linked list and perform the following operations.**

**a. to Insert some Elements at the Specified Position**

**b. swap two elements in a linked list**

**c. to Iterate a LinkedList in Reverse Order**

**d. to Compare Two LinkedList**

**e. to Convert a LinkedList to ArrayList**

**NAME: NISHANTH ROLL NO:22536**

**CLASS: III BCA**

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

package linkedlistpgm;

import java.util.ArrayList;

import java.util.LinkedList;

import java.util.ListIterator;

import java.util.Scanner;

public class LinkedListpgm {

public static void main(String[] args) {

LinkedList<Integer> flist = new LinkedList<Integer>();

LinkedList<Integer> slist = new LinkedList<Integer>();

Scanner sc = new Scanner(System.in);

char choice;

int ele, pos, index1, index2;

do {

System.out.println(" Menu ");

System.out.println("a.Insert Element");

System.out.println("b.Swap elements");

System.out.println("c.Reverse Iterate");

System.out.println("d.Compare List");

System.out.println("e. Convert to array List");

System.out.println("x.Exit");

System.out.println("Enter your choice");

System.out.println("...................................");

choice = sc.next().toLowerCase().charAt(0);

switch (choice) {

case 'a':

System.out.println("Elements in the LinkedList: " + flist);

System.out.print("Enter element to insert: ");

ele = sc.nextInt();

System.out.print("Enter position to insert: ");

pos = sc.nextInt();

if (pos < 0 || pos > flist.size()) {

System.out.println("Invalid position.");

} else {

flist.add(pos, ele);

}

System.out.println("LinkedList: " + flist);

break;

case 'b':

System.out.print("Enter first index to swap: ");

index1 = sc.nextInt();

System.out.print("Enter second index to swap: ");

index2 = sc.nextInt();

if (index1 < 0 || index1 >= flist.size() || index2 < 0 || index2 >= flist.size()) {

System.out.println("Invalid indices.");

} else {

int temp = flist.get(index1);

flist.set(index1, flist.get(index2));

flist.set(index2, temp);

}

System.out.println("LinkedList: " + flist);

break;

case 'c':

System.out.println("LinkedList in Reverse Order: ");

ListIterator<Integer> iterator = flist.listIterator(flist.size());

while (iterator.hasPrevious()) {

System.out.print(iterator.previous() + " ");

}

System.out.println();

break;

case 'd':

System.out.println("Enter elements for the second LinkedList (enter -1 to stop):");

while (true) {

int num = sc.nextInt();

if (num == -1) {

break;

}

slist.add(num);

}

System.out.println("Second list" + slist);

System.out.println("First list" + flist);

if (flist.equals(slist)) {

System.out.println("Both Linkedlist are equal");

} else {

System.out.println("LinkedLists are not equal");

}

break;

case 'e':

ArrayList<Integer> arrayList = new ArrayList<>(flist);

System.out.println("ArrayList: " + arrayList);

break;

case 'x':

System.out.println("Thank you");

return;

default:

System.out.println("Wrong Choice");

}

} while (true);

}

}

**OUTPUT:**

**a. to Insert some Elements at the Specified Position**

Menu

a.Insert Element

b.Swap elements

c.Reverse Iterate

d.Compare List

e. Convert to array List

x.Exit

Enter your choice

...................................

a

Elements in the LinkedList: []

Enter element to insert: 2

Enter position to insert: 0

LinkedList: [2]

Menu

a.Insert Element

b.Swap elements

c.Reverse Iterate

d.Compare List

e. Convert to array List

x.Exit

Enter your choice

...................................

a

Elements in the LinkedList: [2]

Enter element to insert: 3

Enter position to insert: 1

LinkedList: [2, 3]

Menu

a.Insert Element

b.Swap elements

c.Reverse Iterate

d.Compare List

e. Convert to array List

x.Exit

Enter your choice

...................................

a

Elements in the LinkedList: [2, 3]

Enter element to insert: 4

Enter position to insert: 2

LinkedList: [2, 3, 4]

Menu

a.Insert Element

b.Swap elements

c.Reverse Iterate

d.Compare List

e. Convert to array List

x.Exit

Enter your choice

...................................

a

Elements in the LinkedList: [2, 3, 4]

Enter element to insert: 5

Enter position to insert: 3

LinkedList: [2, 3, 4, 5]

**b. swap two elements in a linked list**

Menu

a.Insert Element

b.Swap elements

c.Reverse Iterate

d.Compare List

e. Convert to array List

x.Exit

Enter your choice

...................................

b

Enter first index to swap: 3

Enter second index to swap: 1

LinkedList: [2, 5, 4, 3]

**c. to Iterate a LinkedList in Reverse Order**

Menu

a.Insert Element

b.Swap elements

c.Reverse Iterate

d.Compare List

e. Convert to array List

x.Exit

Enter your choice

...................................

c

LinkedList in Reverse Order:

3 4 5 2

**d. to Compare Two LinkedList**

Menu

a.Insert Element

b.Swap elements

c.Reverse Iterate

d.Compare List

e. Convert to array List

x.Exit

Enter your choice

...................................

d

Enter elements for the second LinkedList (enter -1 to stop):

2

3

4

5

-1

Second list[2, 3, 4, 5]

First list[2, 5, 4, 3]

LinkedLists are not equal

**e. to Convert a LinkedList to ArrayList**

Menu

a.Insert Element

b.Swap elements

c.Reverse Iterate

d.Compare List

e. Convert to array List

x.Exit

Enter your choice

...................................

e

ArrayList: [2, 5, 4, 3]

Menu

a.Insert Element

b.Swap elements

c.Reverse Iterate

d.Compare List

e. Convert to array List

x.Exit

Enter your choice

...................................

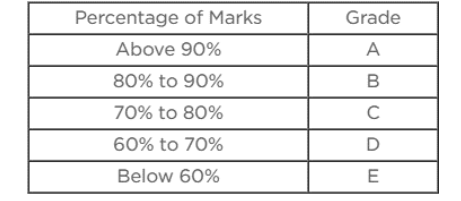
x

Thank you

BUILD SUCCESSFUL (total time: 2 minutes 15 seconds)

**\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\***

**PB8: 8. Implement a java application based on the MVC design pattern. Input student Rolnlo, name, marks in three subjects calculate result and grade and display the result in neat format.**

****

**NAME: NISHANTH ROLL NO:22536**

**CLASS: III BCA**

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

**Main Class:**

**StudentMVC.java**

import java.util.Scanner;

public class StudentMVC {

public static void main(String[] args) {

Studentmodel model = new Studentmodel();

StudentView view = new StudentView();

StudentController controller = new StudentController(model, view);

Scanner sc = new Scanner(System.in);

System.out.println("Enter Roll No:");

String rollno = sc.next();

System.out.println("Enter Name :");

String name = sc.next();

System.out.println("Enter Marks in Subject 1:");

double m1 = sc.nextDouble();

System.out.println("Enter Marks in Subject 2:");

double m2 = sc.nextDouble();

System.out.println("Enter Marks in Subject 3");

double m3 = sc.nextDouble();

controller.setStudentDetails(rollno, name, m1, m2, m3);

controller.calcResult();

controller.printStudentDetails();

}

}

**Class Model:**

**Studentmodel.java**

public class Studentmodel {

private String rollno;

private String Name;

private double marks1;

private double marks2;

private double marks3;

private double percentage;

private char grade;

// defining getter and setter methods

public String getRollno() {

return rollno;

}

public void setRollno(String rollno) {

this.rollno = rollno;

}

public String getName() {

return Name;

}

public void setName(String name) {

this.Name = name;

}

public Double getMarks1() {

return marks1;

}

public void setMarks1(Double marks1) {

this.marks1 = marks1;

}

public Double getMarks2() {

return marks1;

}

public void setMarks2(Double marks2) {

this.marks2 = marks2;

}

public Double getMarks3() {

return marks3;

}

public void setMarks3(Double marks3) {

this.marks3 = marks3;

}

public double getPercentage() {

return percentage;

}

public char getgrade() {

return grade;

}

public void calcResult() {

double total = marks1 + marks2 + marks3;

percentage = (total / 300) \* 100;

if (percentage >= 90) {

grade = 'A';

} else if (percentage >= 80) {

grade = 'B';

} else if (percentage >= 70) {

grade = 'c';

} else if (percentage >= 60) {

grade = 'D';

} else {

grade = 'E';

}

}

}

**Class View**:

**StudentView.java**

public class StudentView {

public void printStudentDetails(Studentmodel Std) {

System.out.println("Student Details: ");

System.out.println("Roll No: " + Std.getRollno());

System.out.println("Name: " + Std.getName());

System.out.println("Marks1: " + Std.getMarks1());

System.out.println("Marks2: " + Std.getMarks2());

System.out.println("Marks3: " + Std.getMarks3());

System.out.println("Marks2: " + Std.getMarks2());

System.out.println("Percentage: " + Std.getPercentage());

System.out.println("Percentage: " + Std.getgrade());

}

}

**Class Controller:**

**StudentController.java**

public class StudentController {

private Studentmodel model;

private StudentView view;

// constructor to initialize

public StudentController(Studentmodel model, StudentView view) {

this.model = model;

this.view = view;

}

public void setStudentDetails(String Rollno, String name, double marks1, double marks2, double marks3) {

model.setRollno(Rollno);

model.setName(name);

model.setMarks1(marks1);

model.setMarks2(marks2);

model.setMarks3(marks3);

}

public void calcResult() {

model.calcResult();

}

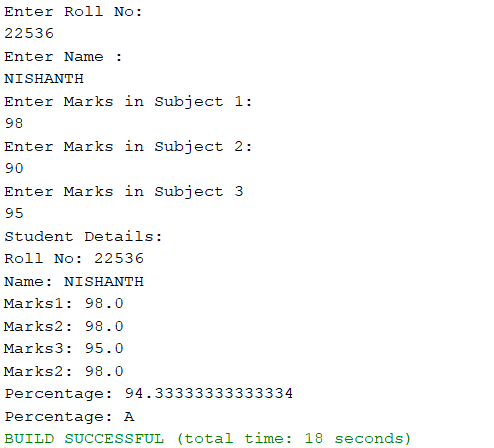
public void printStudentDetails() {

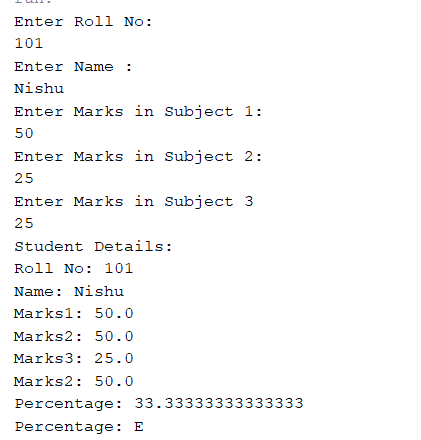
view.printStudentDetails(model);

}

}

**OUTPUT:**

****

****