

**DEPARTMENT OF ARTIFICIAL INTELLIGENCE AND DATA SCIENCE**

**PRACTIAL RECORD BOOK**

**22ADA582: Object Oriented Programming**

NAME USN YEAR/SEM SECTION

BRANCH

:

:

:

:

:

**DEPARTMENT OF ARTIFICIAL INTELLIGENCE AND DATA SCIENCE**

A blue and white logo

Description automatically generated

**CERTIFICATE**

This is to certify that Mr./Mrs. studying in semester Branch has satisfactorily completed the course of experiments in Laboratory for the academic session 2023-2024 as prescribed by the Visvesvaraya Technological University.

Register Number of the Candidate:

Signature of the Staff Incharge Signature of HOD

Date: Date:

| **Sl. No** | **Program / Exercise** |
| --- | --- |
| 1. | Basic programs |
| 2. | Create a Java class called Student with the following details as variables within it.  USN, Name, Branch, Phone  Write a Java program to create n Student objects and print the USN, Name, Branch, and Phone of these objects with suitable headings. |
| 3. | Design a super class called Staff with details as StaffId, Name, Phone, Salary. Extend this class by writing three subclasses namely Teaching (domain, publications), Technical (skills), and Contract (period). Write a Java program to read and display at least 3 staff objects of all three categories. |
| 4. | Write a java program demonstrating Method overloading and Constructor overloading. |
| 5. | Develop a java application to implement currency converter (Dollar to INR, EURO to INR, Yen to INR and vice versa), distance converter (meter to KM, miles to KM and vice versa), time converter (hours to minutes, seconds and vice versa) using packages. |
| 6. | Write a program to generate the resume. Create 2 Java classes Teacher (data: personal information, qualification, experience, achievements) and Student (data: personal information, result, discipline) which implements the java interface Resume with the method biodata(). |
| 7. | Write a Java program that implements a multi-thread application that has three threads. First thread generates a random integer for every 1 second; the second thread computes the square of the number and prints; the third thread will print the value of the cube of the number. |
| 8. | Write a program to perform string operations using ArrayList. Write functions for the following a. Append - add at end b. Insert – add at particular index c. Search d. List all strings starting with a given letter. |
| 9. | Write a Java program to read two integers a and b. Compute a/b and print, when b is not zero. Raise an exception when b is equal to zero. |
| 10 | Write a java program that reads a file name from the user, displays information about whether the file exists, whether the file is readable, or writable, the type of file and the length of the file in bytes |

**INDEX SHEET**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **SL.NO** | **Programs** | **Page no** | **Date of Execution** | **Staff Signature** |
| 1. | Basic programs |  | 02/11/2023 |  |
| 2 | Write a Java program to create n Student objects and print the USN, Name, Branch, and Phone of these objects with suitable headings. |  | 07/11/2023 |  |
| 3. | Design a super class called Staff with details as StaffId, Name, Phone, Salary. Extend this class by writing three subclasses namely Teaching (domain, publications), Technical (skills), and Contract (period). Write a Java program to read and display at least 3 staff objects of all three categories. |  | 21/11/2023 |  |
| 4. | Write a java program demonstrating Method overloading and Constructor overloading. |  | 28/11/2023 |  |
| 5. | Develop a java application to implement currency converter (Dollar to INR, EURO to INR, Yen to INR and vice versa), distance converter (meter to KM, miles to KM and vice versa), time converter (hours to minutes, seconds and vice versa) using packages. |  | 04/12/2023 |  |
| 6. | Write a program to generate the resume. Create 2 Java classes Teacher (data: personal information, qualification, experience, achievements) and Student (data: personal information, result, discipline) which implements the java interface Resume with the method biodata(). |  | 12/12/2023 |  |
| 7. | Write a Java program that implements a multi-thread application that has three threads. First thread generates a random integer for every 1 second; the second thread computes the square of the number and prints; the third thread will print the value of the cube of the number. |  | 19/12/2023 |  |
| 8. | Write a program to perform string operations using ArrayList. Write functions for the following a. Append - add at end b. Insert – add at particular index c. Search d. List all strings starting with a given letter. |  | 26/12/2023 |  |
| 9. | Write a Java program to read two integers a and b. Compute a/b and print, when b is not zero. Raise an exception when b is equal to zero. |  | 16/01/2024 |  |
| 10. | Write a java program that reads a file name from the user, displays information about whether the file exists, whether the file is readable, or writable, the type of file and the length of the file in bytes |  | 16/01/2024 |  |

## Program 1

**Basic Programs in Java**

1. **Addition of two numbers.**
2. **Check whether the given number is even or odd.**
3. **Sum of arrays.**
4. **Reversing of string.**
5. Addition of two numbers

import java.util.Scanner;

public class Demo

{

public static void main(String[] args)

{

int x, y, sum;

Scanner myobj = new Scanner(System.*in*);

System.*out*.println("enter first number:");

x = myobj.nextInt();

System.*out*.println("enter second number:");

y = myobj.nextInt();

sum = x + y;

System.*out*.println("Sum is: " + sum);

}

}

## /\*\* OUTPUT\*\*/

1. Check whether the given number is even or odd.

import java.util.Scanner;

public class Even {

public static void main(String[] args) {

int num;

Scanner x = new Scanner(System.*in*);

System.*out*.println("enter the number");

num = x.nextInt();

if(num % 2 == 0)

{

System.*out*.println(num + " num is even ");

}

else

{

System.*out*.println(num + " num is odd ");

}

}

}

## /\*\*OUTPUT\*\*/

1. Sum of Array

public class Main {

public static void main(String[] args) {

int[] myArray = {1, 5, 10, 25};

int sum = 0;

int i;

for (i = 0; i < myArray.length; i++)

{

sum += myArray[i];

}

System.out.println("The sum is: " + sum);

}

}

## /\*\*OUTPUT\*\*/

1. Reversing String

public class Reverse

{

public static void main(String[] args)

{

String str = "Hello";

String rstr = "";

System.out.println("Original string: " + str);

for (int i = 0; i < str.length(); i++)

{

rstr = str.charAt(i) + rstr;

}

System.out.println("Reversed string: "+ rstr);

}

}

**/\*\*OUTPUT\*\*/**

## Program 2

**Create a Java class called Student with the following details as variables within it.**

**USN**

**Name**

**Branch**

**Phone number**

**Write a Java program to create n Student objects and print the USN, Name, Branch, and Phone number of these objects with suitable headings.**

import java.util.Scanner;

public class Student

{

String usn,name,branch;

long phone;

void insertStudent(String reg,String nm, String br,long ph)

{

usn=reg;

name=nm;

branch=br;

phone=ph;

}

void displayStudent()

{

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

System.out.println("USN= "+usn);

System.out.println("NAME= "+name);

System.out.println("BRANCH= "+branch);

System.out.println("PHONE NUMBER= "+phone);

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

}

public static void main(String args[])

{

Student st[]=new Student [100];

Scanner ip=new Scanner(System.in);

System.out.println("Enter the number of students");

int n=ip.nextInt();

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

st[i]=new Student();

for(int j=0;j<n;j++)

{

System.out.println("Enter the Usn,Name,Branch,Phone Number");

String usn=ip.next();

String name=ip.next();

String branch=ip.next();

long phone=ip.nextLong();

st[j].insertStudent(usn,name,branch,phone);

}

for( int m=0;m<n;m++)

{

System.out.format("Student %d details are\n",m+1);

st[m].displayStudent();

}

}

}

## /\*\*OUTPUT\*\*/

## Program 3

**Design a super class called Staff with details as StaffId, Name, Phone, Salary. Extend this class by writing three subclasses namely Teaching (domain, publications), Technical (skills), and Contract (period). Write a Java program to read and display at least 3 staff objects of all three categories.**

import java.util.Scanner;

class Staff

{

String staffId;

String name;

long phone;

float salary;

public void accept()

{

Scanner scanner = new Scanner(System.in);

System.out.print("Enter Staff Id: ");

staffId = scanner.next();

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

name = scanner.next();

System.out.print("Enter Phone: ");

phone = scanner.nextLong();

System.out.print("Enter Salary: ");

salary = scanner.nextFloat();

}

public void display()

{

System.out.println("Staff Id: " + staffId);

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

System.out.println("Phone: " + phone);

System.out.println("Salary: " + salary);

}

}

class Teaching extends Staff

{

String domain;

int n;

public void accept()

{

super.accept();

Scanner scanner = new Scanner(System.in);

System.out.print("Enter Domain: ");

domain = scanner.next();

System.out.print("Enter Number of Publications: ");

n = scanner.nextInt();

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

}

public void display()

{

super.display();

System.out.println("Domain: " + domain);

System.out.println("Publications:"+n);

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

}

}

class Technical extends Staff

{

String skill;

public void accept()

{

super.accept();

Scanner scanner = new Scanner(System.in);

System.out.print("Enter technical Skills: ");

skill = scanner.nextLine();

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

}

public void display()

{

super.display();

System.out.println("Technical Skills: " + skill);

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

}

}

class Contract extends Staff

{

int period;

public void accept()

{

super.accept();

Scanner scanner = new Scanner(System.in);

System.out.print("Enter Period: ");

period = scanner.nextInt();

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

}

public void display()

{

super.display();

System.out.println("Contract Period: " + period);

}

}

class Four

{

public static void main(String[] args)

{

Teaching teaching = new Teaching();

System.out.println("Enter the details of Teaching Staff");

teaching.accept();

Technical technical = new Technical();

System.out.println("Enter the details of Technical Staff");

technical.accept();

Contract contract = new Contract();

System.out.println("Enter the details of Contract Staff");

contract.accept();

System.out.println("The details of Teaching Staff");

teaching.display();

System.out.println("The details of Technical Staff");

technical.display();

System.out.println("The details of Contract Staff");

contract.display();

}

}

## /\*\*OUTPUT\*\*/

## Program 4

## Write a java program demonstrating Method overloading and Constructor overloading.

## Method overloading

## class MOverloading

## {

## int add(int a, int b)

## {

## int sum = a+b;

## return sum;

## }

## int add(int a, int b, int c)

## {

## int sum = a+b+c;

## return sum;

## }

## float add(float a, float b)

## {

## float sum = a+b;

## return sum;

## }

## }

## class MOverload

## {

## public static void main(String args[])

## {

## MOverloading obj = new MOverloading();

## int s1=obj.add(10, 20);

## int s2=obj.add(10, 20, 30);

## float s3=obj.add(2.2f,2.2f);

## System.out.println("Method Overload Sum1="+s1);

## System.out.println("Method Overload Sum2="+s2);

## System.out.println("Method Overload Sum3="+s3);

## }

## /\*\*OUTPUT\*\*/

## Constructor overloading

public class Constructor

{

int id;

String name;

Constructor()

{

System.out.println("This is Default constructor");

System.out.println("Student Id : "+id + "\nStudent Name : "+name);

}

Constructor(int i, String n)

{

System.out.println("This is Parameterized Constructor:");

id = i;

name = n;

System.out.println("Student Id : "+id + "\nStudent Name : "+name);

}

public static void main(String[] args)

{

Constructor s = new Constructor();

Constructor student = new Constructor(10, "David");

}

}

## }

**/\*\*OUTPUT\*\*/**

**Program 5**

**Develop a java application to implement currency converter (Dollar to INR, EURO to INR, Yen to INR and vice versa), distance converter (meter to KM, miles to KM and vice versa), time converter (hours to minutes, seconds and vice versa) using packages.**

**CurrencyC.java**

package cc;

import java.util.\*;

public class CurrencyC

{

double inr,usd;

double euro,yen;

Scanner in=new Scanner(System.*in*);

public void dollartorupee()

{

System.*out*.println("Enter dollars to convert into Rupees:");

usd=in.nextInt();

inr=usd\*83.36;

System.*out*.println("Dollar ="+usd+" equal to INR="+inr);

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

}

public void rupeetodollar()

{

System.*out*.println("Enter Rupee to convert into Dollars:");

inr=in.nextInt();

usd=inr/83.36;

System.*out*.println("Rupee ="+inr+"equal to Dollars="+usd);

}

public void eurotorupee()

{

System.*out*.println("Enter Euro to convert into Rupees:");

euro=in.nextInt();

inr=euro\*89.76;

System.*out*.println("Euro ="+euro+" equal to INR="+inr);

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

}

public void rupeetoeuro()

{

System.*out*.println("Enter Rupees to convert into Euro:");

inr=in.nextInt();

euro=(inr/89.76);

System.*out*.println("Rupee ="+inr +"equal to Euro="+euro);

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

}

public void yentoruppe()

{

System.*out*.println("Enter Yen to convert into Rupees:");

yen=in.nextInt();

inr=yen\*0.57;

System.*out*.println("Yen ="+yen+" equal to INR="+inr);

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

}

public void ruppetoyen()

{

System.*out*.println("Enter Rupees to convert into Yen:");

inr=in.nextInt();

yen=(inr/0.57);

System.*out*.println("INR="+inr +"equal to YEN"+yen);

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

}

}

**DistanceC.java**

package dc;

import java.util.\*;

public class DistanceC

{

double km,m,miles;

Scanner in=new Scanner(System.in);

public void mtokm()

{

System.out.println("Enter the distance in meter");

m=in.nextDouble();

km=(m/1000);

System.out.println(m+"m" +" is equal to "+km+"km");

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

}

public void kmtom()

{

System.out.println("Enter the distance in Kilometer");

km=in.nextDouble();

m=km\*1000;

System.out.println(km+"km" +" is equal to "+m+"m");

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

}

public void milestokm()

{

System.out.println("Enter the distance in miles");

miles=in.nextDouble();

km=(miles\*1.60934);

System.out.println(miles+"miles" +" is equal to "+km+"km");

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

}

public void kmtomiles()

{

System.out.println("Enter the distance in km");

km=in.nextDouble();

miles=(km\*0.621371);

System.out.println(km+"km" +" is equal to "+miles+"miles");

}

}

**TimeC.java**

package tc;

import java.util.\*;

public class TimeC

{

int hours,seconds,minutes;

Scanner in = new Scanner(System.in);

public void hourstominutes()

{

System.out.println("Enter the no of Hours to convert into minutes");

hours=in.nextInt();

minutes=(hours\*60);

System.out.println("Minutes: " + minutes);

}

public void minutestohours()

{

System.out.println("Enter the no of Minutes to convert into Hours");

minutes=in.nextInt();

hours=minutes/60;

System.out.println("Hours: " + hours);

}

public void hourstoseconds()

{

System.out.println("Enter the no of Hours to convert into Seconds");

hours=in.nextInt();

seconds=(hours\*3600);

System.out.println("Seconds: " + seconds);

}

public void secondstohours()

{

System.out.println("Enter the no of Seconds to convert into Hours");

seconds=in.nextInt();

hours=seconds/3600;

System.out.println(seconds+"seconds"+ " is equal to "+hours+"hour");

}

}

**MainC.java**

package mc;

import cc.\*;

import dc.\*;

import tc.\*;

public class Main

{

public static void main(String args[])

{

CurrencyC obj=new CurrencyC();

DistaceC obj1=new DistanceC();

TimeC obj2=new TimeC();

obj.dollartorupee();

obj.rupeetodollar();

obj.eurotorupee();

obj.rupeetoeuro();

obj.yentoruppe();

obj.ruppetoyen();

obj1.mtokm();

obj1.kmtom();

obj1.milestokm();

obj1.kmtomiles();

obj2.hourstominutes();

obj2.minutestohours();

obj2.hourstoseconds();

obj2.secondstohours();

}

}

**/\*\*OUTPUT\*\*/**

**Program 6**

**Write a program to generate the resume. Create 2 Java classes Teacher (data: personal information, qualification, experience, achievements) and Student (data: personal information, result, discipline) which implements the java interface Resume with the method biodata().**

interface Resume

{

void biodata();

}

class Teacher implements Resume

{

String name,qualification,achievements;

float experience;

public void biodata()

{

name="Ram Sharma";

qualification="M.Tech";

achievements="P1 publication";

experience=19.8f;

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

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

System.out.println("Qualification :"+qualification);

System.out.println("Achievements : "+achievements);

System.out.println("Experience : "+experience);

}

}

class Student implements Resume

{

String name,discipline;

float result;

public void biodata()

{

name="Rohini V ";

result=7.8f;

discipline="AI&DS";

System.out.println("");

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

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

System.out.println("Result : "+result+" cgpa");

System.out.println("Discipline : "+discipline);

}

}

public class InterfaceP

{

public static void main(String[] args)

{

Teacher obj1=new Teacher();

obj1.biodata();

Student obj2=new Student();

obj2.biodata();

}

}

**/\*\*OUTPUT\*\*/**

**Program 7**

**Write a Java program that implements a multi-thread application that has three threads. First thread generates a random integer for every 1 second; second thread computes the square of the number and prints; third thread will print the value of cube of the number.**

import java.util.Random;

class Square extends Thread

{

int x;

Square(int n)

{

x = n;

}

public void run()

{

int sqr = x \* x;

System.out.println("Square of " + x + " = " + sqr );

}

}

class Cube extends Thread

{

int x;

Cube(int n)

{

x = n;

}

public void run()

{

int cub = x \* x \* x;

System.out.println("Cube of " + x + " = " + cub );

}

}

class Rnumber extends Thread

{

public void run()

{

Random random = new Random();

for(int i =0; i<5; i++)

{

int randomInteger = random.nextInt(10);

System.out.println("Random Integer generated : " + randomInteger);

Square s = new Square(randomInteger);

s.start();

Cube c = new Cube(randomInteger);

c.start();

try

{

Thread.sleep(1000);

}

catch (InterruptedException ex)

{

System.out.println(ex);

}

}

}

}

public class ThreadP

{

public static void main(String[] args)

{

Rnumber n = new Rnumber();

n.start();

}

}

**/\*\*OUTPUT\*\*/**

**Program 8**

**Write a program to perform string operations using ArrayList. Write functions for the following a. Append - add at end b. Insert – add at particular index c. Search d. List all string starts with given letter**

import java.util.\*;

public class ArrayL

{

ArrayList<String> list=new ArrayList<String>(); //Creating arraylist

public void arraydisplay()

{

list.add("CSE");//Adding object in arraylist

list.add("ISE");

list.add("ME");

System.out.println("ArrayList element are");

System.out.println(list);

System.out.println("");

}

public void appendatend()

{

System.out.println("Enter the element to append at end");

Scanner num=new Scanner(System.in);

String ele=num.next();

list.add(ele);

System.out.println(list);

System.out.println("");

}

public void insertatpos()

{

System.out.println("Enter the position and element to insert");

Scanner num=new Scanner(System.in);

int posind=num.nextInt();

String ele=num.next();

list.add(posind,ele);

System.out.println(list);

System.out.println("");

}

public void searchele()

{

System.out.println("Enter the Array element to search");

Scanner scobj=new Scanner(System.in);

String arele=scobj.next();

int in=list.indexOf(arele);

if(in==-1)

{

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

}

else

{

System.out.println("Element found at "+in);

}

}

void print()

{

Scanner nip=new Scanner(System.in);

System.out.println("Enter the starting character to print strings");

char inputc=nip.next().charAt(0);

String strc=Character.toString(inputc);

System.out.println("String starting with character "+strc);

for(int i=0;i<list.size();i++)

{

if(list.get(i).startsWith(strc))

{

System.out.println(list.get(i));

}

}

}

public static void main(String args[])

{

ArrayL obj=new ArrayL();

obj.arraydisplay();

obj.appendatend();

obj.insertatpos();

obj.searchele();

obj.print();

}

}

**/\*\*OUTPUT\*\*/**

**Program 9**

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

import java.util.\*;

public class TryP

{

int c;

void div(int a,int b)

{

try

{

c=a/b;

System.out.println("Result="+c);

}

catch(ArithmeticException e)

{

System.out.println("Cannot divide by zero");

}

}

public static void main(String args[])

{

TryP obj=new TryP();

Scanner in=new Scanner(System.in);

System.out.println("Enter the values of a and b");

int no1=in.nextInt();

int no2=in.nextInt();

obj.div(no1,no2);

}

}

**/\*\*OUTPUT\*\*/**

**Program 10**

**Write a java program that reads a file name from the user, displays information about whether the file exists, whether the file is readable, or writable, the type of file and the length of the file in bytes.**

import java.io.File;

import java.util.Scanner;

class FileP

{

public static void main(String args[ ])

{

Scanner obj=new Scanner(System.in);

String fname=obj.next();

File f1 = new File(fname);

System.out.println("File Name: " + f1.getName());

f1.setWritable(false);

System.out.println(f1.exists() ? "File exists" : "File does not exist");

System.out.println(f1.canWrite() ? "File is writeable" : "File is not writeable");

System.out.println(f1.canRead() ? "File is readable" : "File is not readable");

String fileName = f1.toString();

int index = fileName.lastIndexOf('.');

if(index > 0)

{

String type = fileName.substring(index + 1);

System.out.println("File type is " + type);

}

else

{

System.out.println("File doesn't have type");

}

System.out.println("File size: " + f1.length() + " Bytes");

}

}

**/\*\*OUTPUT\*\*/**

Internal Evaluation

Date: 23/01/2024

Name of the program obtained:

|  |  |  |  |
| --- | --- | --- | --- |
| **Sl.no** | **Particulars** | **Max Marks** | **Obtained Marks** |
| 1. | Observation Book | 10 |  |
| 2. | Record Book | 10 |  |
| 3. | Compilation of Program | 12 |  |
| 4. | Execution of Program | 13 |  |
| 5. | Viva | 05 |  |
|  | **Total** | 50 |  |

Total marks secured:

Signature of Faculty