**Experiment – 6**

Q1) Write a program in Java to develop user defined exception for “Divide by Zero” error.

Ans:

**Program:**

// exp 1

package Exp\_6;

import java.util.Scanner;

public class DivideByZero{

    public static void main(String[] args) {

        Scanner sc = new Scanner(System.in);

        System.out.print("Enter first value: ");

        int n = sc.nextInt();

        System.out.print("Enter second value: ");

        int m = sc.nextInt();

        try{

            double ans = n/m;

            System.out.println(ans);

        }

        catch(ArithmeticException e){

            System.out.print("Second number cannot be 0");

        }

        catch(Exception e){

            System.out.println("Something unexpected happened!");

        }

        sc.close();

    }

}

**Output:**

A computer screen with white text

AI-generated content may be incorrect.

Q2) Write a program in Java to demonstrate throw, throws, finally, multiple try block and multiple catch exception

Ans:

**Program:**

// exp 2

package Exp\_6;

import java.util.Scanner;

public class Second {

    public static void checkAge(int age) throws ArithmeticException{

        if(age<18){

            throw new ArithmeticException("You are ineligible");

        }

        else{

            System.out.println("You are eligible");

        }

    }

    public static void main(String[] args){

        Scanner sc = new Scanner(System.in);

        try{

            int age = Integer.parseInt(args[0]);

            if(age<0){

                throw new ArithmeticException("Negative age not possible");

            }

            checkAge(age);

        }

        catch(NumberFormatException e){

            System.out.println("Enter valid age in integers only.");

        }

        catch(ArithmeticException e){

            System.out.println(e.getMessage());

        }

        catch(Exception e){

            System.out.println("Something unexpected happened!");

        }

        finally{

            sc.close();

            System.out.println("Exiting the program....");

        }

    }

}

**Output:**

A computer screen shot of white text

AI-generated content may be incorrect.

Q3) Write a small application in Java to develop Banking Application in which user deposits the amount Rs 1000.00 and then start withdrawing ofRs 400.00, Rs 300.00 and it throws exception "Not Sufficient Fund" when user withdraws Rs 500 thereafter

Ans:

**Program:**

// exp 3

package Exp\_6;

import java.util.Scanner;

class LowBalance extends Exception{

    LowBalance(){super("Insufficient Balance");}

    LowBalance(String message){super(message);}

}

public class BankAccount {

    public static void main(String[] args) {

        Scanner sc = new Scanner(System.in);

        try{

System.out.print("Enter amount to deposit: ");

        int amt = sc.nextInt();

            while(true){

                System.out.println("Current Balance: "+amt);

                System.out.print("Enter amount to withdraw(-1 to exit): ");

                int wd = sc.nextInt();

                if(wd == -1){

                    break;

                }

                if(wd > amt){

                    throw new LowBalance("You have insufficent balance");

                }

                amt -= wd;

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

            }

        }

        catch(LowBalance e){

            System.out.println(e.getMessage());

        }

        catch(NumberFormatException e){

            System.out.println("Enter only integral inputs! ");

        }

        catch(Exception e){

            System.out.println("Something unexpected happened!");

        }

        finally{

            System.out.println("Exiting the system. Thank you for coming!");

            sc.close();

        }

    }

}

**Output:**

A computer screen shot of a black screen

AI-generated content may be incorrect.

Q4) Write an application that contains a method named average () has one argument that is an array of strings. It converts these to double values and returns their average. The method generates a NullPointerException,if an array elements is null or a NumberFormatException, if an element is incorrectly formatted. Include throws statement in method declaration.

Ans:

**Program:**

// exp 4

package Exp\_6;

public class AverageMethod {

    public static double average(String[] a) throws NullPointerException, NumberFormatException{

        if(a.length == 0){

            throw new NullPointerException("No numbers entered!");

        }

        int n = a.length;

        double[] result = new double[n];

        for(int i=0;i<n;i++){

            double d = Double.parseDouble(a[i]);

            result[i] = d;

        }

        double sum = 0;

        for (double i : result){

            sum += i;

        }

        double avg = sum/n;

        return avg;

    }

    public static void main(String[] args) {

        try{

            System.out.println(average(args));

        }

        catch(NullPointerException e){

            System.out.println(e.getMessage());

        }

        catch(NumberFormatException e){

            System.out.println("Enter only double values");

        }

        catch(Exception e){

            System.out.println("Something unexpected occured");

        }

        finally{

            System.out.println("Exiting the program..");

        }

    }

}

**Output:**

A computer screen shot of a program code

AI-generated content may be incorrect.

Q5) Write an application that generates custom exception if first argument from command line argument is 0.

Ans:

**Program:**

// exp 5

package Exp\_6;

class ArgumentZero extends Exception{

    ArgumentZero(){super("First argument is zero!!");}

    ArgumentZero(String message){super(message);}

}

public class FirstArgumentZero {

    public static void main(String[] args) {

        try{

            if(args.length==0){

                throw new StringIndexOutOfBoundsException("Insufficient Input");

            }

            int i = Integer.parseInt(args[0]);

            if(i == 0){

                throw new ArgumentZero("You have entered the first argument as 0");

            }

            else{

                System.out.println("Good job! You entered non zero initial value");

            }

        }

        catch(StringIndexOutOfBoundsException e){

            System.out.println(e.getMessage());

        }

        catch(ArgumentZero e){

            System.out.println(e.getMessage());

        }

        catch(Exception e){

            System.out.println(e.getMessage());

        }

        finally{System.out.println("Exiting the program...");}

    }

}

**Output:**

A computer screen shot of a program

AI-generated content may be incorrect.

Q6) A marklist containing reg.no and marks for a subject is given.if the marks are <0,user-defined IllegalMarkException is thrown out and handled with the message "Illegal Mark". For all valid marks, the candidate will be declared as "PASS" if the marks are equal to or greater than 40, otherwise it will be declared as "FAIL".Write a class called IllegalMarkException.

Ans:

**Program:**

// exp 6

package Exp\_6;

import java.util.Scanner;

class IllegalMarksException extends Exception{

    IllegalMarksException(){super("Illegal Marks");}

    IllegalMarksException(String message){super(message);}

}

public class MarkLists{

    public static void checkMarks(int[] regNo, double[] marks) throws IllegalMarksException{

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

            if(marks[i] <0){

                throw new IllegalMarksException();

            }

        }

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

            if(marks[i] >= 40){

                System.out.println(regNo[i] + " : PASS");

            }

            else{

                System.out.println(regNo[i] + " : FAIL");

            }

        }

    }

    public static void main(String[] args) {

        Scanner sc = new Scanner(System.in);

        System.out.println("Enter number of entries: ");

        int size = sc.nextInt();

        int[] regNo = new int[size];

        double[] marks = new double[size];

        for(int i=0;i<size;i++){

            System.out.print("Enter Registration Number: ");

            regNo[i] = sc.nextInt();

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

            marks[i] = sc.nextDouble();

        }

        try{

            checkMarks(regNo, marks);

        }

        catch(IllegalMarksException e){

            System.out.println(e.getMessage());

        }

        catch(Exception e){

            System.out.println("Something unexpected happened!");

        }

        finally{

            System.out.println("Exiting the program...");

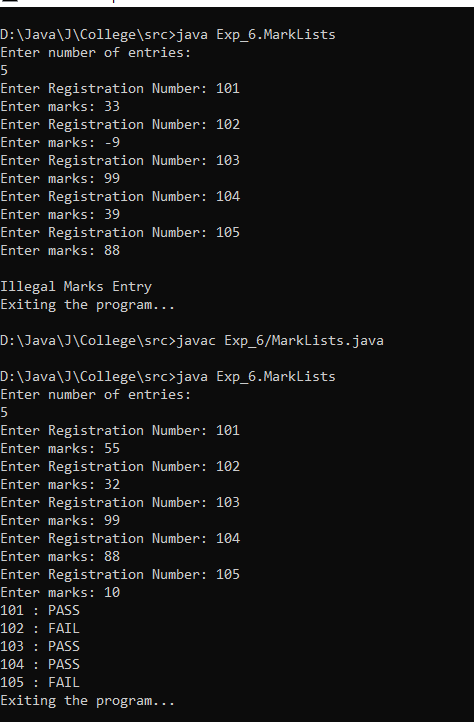
        }

        sc.close();

    }

}

**Output:**



Q7) Assume that there are two packages, student and exam. A student package contains Student class and the exam package contains Result class. Write a program that generates mark sheet for students.

Ans:

**Program:**

// Student class

package Exp\_6.Student;

public class Student{

    int rollNo;

    String name;

    String branch;

    double marks;

    public Student(){}

    public Student(int rollNo, String name, String branch){

        this.rollNo = rollNo;

        this.name = name;

        this.branch=branch;

    }

    public void showMarks(double marks){

        if(marks>100 || marks<0) System.out.println("Invalid Marks!");

        this.marks = marks;

        this.showStudentDetails();

    }

    public void showStudentDetails(){

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

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

        System.out.println("Branch: "+branch);

        System.out.println("Marks: "+marks);

    }

}

// Result class

package Exp\_6.Exam;

import Exp\_6.Student.\*;

public class Result {

    public static void main(String[] args) {

        Student s1 = new Student(7,"Aryan","IT");

        Student s2 = new Student(8,"Jay","CE");

        s1.showMarks(99);

        s2.showMarks(46);

    }

}

**Output:**

A computer screen shot of a black screen

AI-generated content may be incorrect.

Q8) Define a class A in package a pack. In class A, three variables are defined of access modifiers protected, private and public. Define class B in package bpack which extends A and write display method which accesses variables of class A. Define class C in package cpack which has one method display() in that create one object of class A and display its variables. Define class ProtectedDemo in package dpack in which write main() method. Create objects of class B and C and class display method for both these objects

Ans:

**Program:**

// A

package Exp\_6.apack;

public class A {

    public int x=6;

    protected int y=7;

    private int z=9;

}

// B

package Exp\_6.bpack;

import Exp\_6.apack.A;

public class B extends A  {

    public void display(){

        System.out.println("X in B: "+x);

        System.out.println("Y in B: "+y);

        System.out.println("Z in B: "+z);

    }

}

// C

package Exp\_6.cpack;

import Exp\_6.apack.A;

public class C {

    public void display(){

        A a1 = new A();

        System.out.println("X in C: "+a1.x);

        System.out.println("Y in C: "+a1.y);

        System.out.println("Z in C: "+a1.z);

    }

}

// ProtectedDemo

package Exp\_6.dpack;

import Exp\_6.bpack.\*;

import Exp\_6.cpack.\*;

public class ProtectedDemo {

    public static void main(String[] args) {

        B b1 = new B();

        C c1 = new C();

        b1.display();

        c1.display();

    }

}

**Output:**

