OUESTION 1

import java.io.\*;

public class MyClass {

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

String num1 ,num2 ;

int n1,n2, d;

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

try{

num1 = br .readLine();

num2 = br .readLine();

n1 = Integer .parseInt(num1);

n2 = Integer .parseInt(num2);

d = n1 / n2;

System .out.println("output "+ d);

}

catch (NumberFormatException e)

{

System.out.println("input are not valid");

}

catch (ArithmeticException ae)

{

System.out.println ("divide by zero error");

}

}

}



QUESTION 2

import java.io.\*;

class BalanceCheck extends Exception{

BalanceCheck(){

super("Transaction Denied: No min balance found");

}

}

class Bank{

int accountno;

String name;

double balance;

static int min\_amount=500;

Bank(int ano,String nm,double bal,int min){

accountno=ano;

name=nm;

balance=bal;

}

void Withdraw(double cash) throws BalanceCheck{

if((balance-cash)>=min\_amount){

balance=balance-cash;

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

System.out.println("the balance after withdrawl is "+ balance);

}

else{

throw new BalanceCheck();

}

}

void Deposit(double cash){

balance = balance+cash;

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

System.out.println("the balance after deposit is "+balance);

}

void CheckBal(){

System.out.println("the balance is"+balance);

}

}

public class Main

{

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

Bank b1 = new Bank(8259,"hdfc",250000,1000);

Bank b2 = new Bank(3163,”sbi",2000000,2000);

try{

b1.Withdraw(20000);

b2.Withdraw(12000);

}

catch(BalanceCheck b){

System.out.println(b);

}

b1.CheckBal();

b2.CheckBal();

b1.Deposit(10000);

b2.Deposit(18000);

b1.CheckBal();

b2.CheckBal();

}

}



question 3

import java.util.Scanner;

class AgeCheck extends Exception

{

AgeCheck()

{

super("Exception : invalid Age");

}

}

public class MyClass{

public static void main(String args[]) {

int age;

Scanner s = new Scanner(System.in);

age = s.nextInt();

boolean b;

try{

b= CheckAge (age);

System.out.println("valid");

}

catch (AgeCheck ag)

{

System.out.println(ag);

}

}

static boolean CheckAge(int age) throws AgeCheck

{

if(age > 18 && age<=120)

return true;

else

{

throw new AgeCheck();

}

}

}



Question 4

class FullStack extends Exception

{

FullStack()

{

super("Stack is Full");

}

}

class EmptyStack extends Exception

{

EmptyStack()

{

super("Stack is Empty ");

}

}

class Stack

{

int top;

int arr[];

static int max=9;

Stack()

{

top=-1;

arr=new int[max];

}

void push(int x)throws FullStack

{

if(top==max-1)

{

throw new FullStack();

}

else

{

arr[++top]=x;

}

}

int pop()throws EmptyStack

{

if(top==-1)

{

throw new EmptyStack();

}

else

{

return(arr[top--]);

}

}

void print()

{

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

System.out.print(arr[i]+" ");

System.out.println();

}

}

public class MyClass {

public static void main(String args[]) {

Stack s1=new Stack();int x;

for(int i=1;i<=14;i++)

{

try{

s1.push(i);

s1.print();

}

catch(FullStack fs)

{

System.out.println(fs);

}

}

for (int i=1;i<=14;i++)

{

try

{

x=s1.pop();

System.out.print(x +" ");

}

catch(EmptyStack es)

{

System.out.println(es);

}

}

}

}

