Question 1:

import java.util.\*;

class firstClass{

public int sum(int a, int b)

{

return (a+b);

}

}

class secondClass extends firstClass{

}

class Q1 {

public static void main(String[] args)

{

System.out.println("Enter two numbers: ");

Scanner sc =new Scanner(System.in);

int a = sc.nextInt();

int b= sc.nextInt();

secondClass x =new secondClass();

int c = x.sum(a,b);

System.out.println(c);

}}

Question 2:

import java.util.\*;

interface factorsSum{

public int fsum(int num);

}

class division implements factorsSum{

public int fsum(int num){

int sum=0;

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

{

if(num%i==0)

sum+=i;

}

return sum;

}

}

class Q2 {

public static void main(String[] args)

{

division x =new division();

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

Scanner sc =new Scanner(System.in);

int numb=sc.nextInt();

int ans=x.fsum(numb);

System.out.println("Sum of divisors is : "+ ans);

}

}

Question 3:

import java.util.\*;

public class sortDescending {

public static void main(String[] args) {

String[] str = {"-100", "0.12", ".12", "0000.000", "11", "0.6"};

Arrays.sort(str, new Comparator<String>() {

@Override

public int compare(String s1, String s2) {

double d1 = Double.parseDouble(s1);

double d2 = Double.parseDouble(s2);

return Double.compare(d2, d1);

}

});

for (String s : str) {

System.out.println(s);

}

}

}

Question 4:

import java.util.\*;

import java.lang.\*;

public class ExceptionHandling{

public static void getm()

{

System.out.println("java.util.InputMismatchException");

}

public static void main(String[] args)

{

try{

Scanner sc =new Scanner(System.in);

System.out.println("enter two numbers: ");

int a = sc.nextInt();

int b= sc.nextInt();

int c= a/b;

System.out.println(c);

}

catch(ArithmeticException e){

System.out.println("Message: "+ e);

}

catch(InputMismatchException a)

{

getm();

}

}

}

Question 5 :

import java.util.\*;

class SbibankAccount{

int premium\_amount =10000;

int jundhan\_amount=100;

public void deposit(){System.out.println("Your money got deopsited");}

public void withdraw(int Awithdraw){}

public void openAccount(int num){}

}

class premiumAccount extends SbibankAccount{

SbibankAccount sc = new SbibankAccount();

int x = sc.premium\_amount;

public void openAccount(int num)

{

if(num>=5000)

System.out.println("Premium account opened");

else

System.out.println("Amount is insufficient to open (min balance sould be 5000)!!");

}

public void withdraw(int Awithdraw){

int sum=x-Awithdraw;

if(sum<5000)

{

sum+=Awithdraw;

System.out.println("unable to withdraw due to insufficient balance");

}

else

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

}

}

class jundhanAccount extends SbibankAccount{

SbibankAccount sc = new SbibankAccount();

int x = sc.jundhan\_amount;

public void openAccount(int num)

{

if(num>=0)

System.out.println(" jundhan account opened");

else

System.out.println("insufficient balance!!");

}

public void withdraw(int Awithdraw){

int sum=x-Awithdraw;

if(sum<0)

{

sum+=Awithdraw;

System.out.println("unable to withdraw due to insufficient balance");

}

else

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

}

}

class SbiAccount{

public static void main (String[] args)

{

System.out.println("Welcome to Sbi");

System.out.println("Press 1 to create Premium account");

System.out.println("Press 2 to create jundhan account");

System.out.println("Press 2 to create jundhan account");

System.out.println("For other functions press 3");

Scanner sc =new Scanner(System.in);

int num = sc.nextInt();

if(num==1)

{

System.out.println("enter amount :");

int am = sc.nextInt();

premiumAccount pa = new premiumAccount();

pa.openAccount(am);

}

else if(num==2)

{

System.out.println("enter amount :");

int am = sc.nextInt();

jundhanAccount ja = new jundhanAccount();

ja.openAccount(am);

}

else{

System.out.println("Press 4 to deposit");

System.out.println("Press 5 to withdraw" );

int num1 = sc.nextInt();

{

if(num1== 4)

{

System.out.println("enter the amount");

int s= sc.nextInt();

SbibankAccount sa = new SbibankAccount();

sa.deposit();

}

if(num1== 5)

{

System.out.println("Press 8 to withdraw from premium account");

System.out.println("Press 9 to withdraw from jundhan account ");

int num2 =sc.nextInt();

if(num2==8)

{

System.out.println("enter the amount:");

int num3 = sc.nextInt();

premiumAccount pa = new premiumAccount();

pa.withdraw(num3);

}

else if(num2==9)

{

System.out.println("enter the amount:");

int num3 = sc.nextInt();

jundhanAccount ja = new jundhanAccount();

ja.withdraw(num3);

}

}

}

}

System.out.println("Your task is ended succesfully!!");

}

}

Question 6:

import java.util.\*;

interface Account {

void deposit(double amount);

void withdraw(double amount);

double calculateInterest();

double viewBalance();

}

class SavingsAccount implements Account {

private double balance;

private double interestRate;

public SavingsAccount(double balance, double interestRate) {

this.balance = balance;

this.interestRate = interestRate;

}

@Override

public void deposit(double amount) {

balance += amount;

System.out.println("Amount " + amount + " deposited successfully to Savings Account.");

}

@Override

public void withdraw(double amount) {

if (balance >= amount) {

balance -= amount;

System.out.println("Amount " + amount + " withdrawn successfully from Savings Account.");

} else {

System.out.println("Insufficient balance. Withdrawal failed.");

}

}

@Override

public double calculateInterest() {

return balance \* interestRate;

}

@Override

public double viewBalance() {

return balance;

}

public void addInterest() {

double interest = calculateInterest();

balance += interest;

System.out.println("Interest added to Savings Account: " + interest);

}

}

class CurrentAccount implements Account {

private double balance;

private double overdraftLimit;

public CurrentAccount(double balance, double overdraftLimit) {

this.balance = balance;

this.overdraftLimit = overdraftLimit;

}

@Override

public void deposit(double amount) {

balance += amount;

System.out.println("Amount " + amount + " deposited successfully to Current Account.");

}

@Override

public void withdraw(double amount) {

if (balance + overdraftLimit >= amount) {

balance -= amount;

System.out.println("Amount " + amount + " withdrawn successfully from Current Account.");

} else {

System.out.println("Insufficient balance. Withdrawal failed.");

}

}

@Override

public double calculateInterest() {

return 0;

}

@Override

public double viewBalance() {

return balance;

}

public void setOverdraftLimit(double overdraftLimit) {

this.overdraftLimit = overdraftLimit;

System.out.println("Overdraft limit updated: " + overdraftLimit);

}

}

class Bank {

private List<Account> accounts;

public Bank() {

accounts = new ArrayList<>();

}

public void addAccount(Account account) {

accounts.add(account);

}

}

class Q5Account {

public static void main(String[] args) {

Bank bank = new Bank();

SavingsAccount savingsAccount = new SavingsAccount(5000, 0.05);

bank.addAccount(savingsAccount);

CurrentAccount currentAccount = new CurrentAccount(2000, 5000);

bank.addAccount(currentAccount);

savingsAccount.deposit(1000);

savingsAccount.withdraw(2000);

double savingsBalance = savingsAccount.viewBalance();

System.out.println("Savings Account Balance: " + savingsBalance);

currentAccount.deposit(3000);

currentAccount.withdraw(5000);

double currentBalance = currentAccount.viewBalance();

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

savingsAccount.addInterest();

double updatedSavingsBalance = savingsAccount.viewBalance();

System.out.println("Updated Savings Account Balance: " + updatedSavingsBalance);

currentAccount.setOverdraftLimit(100000);

}

}