Practical Assignments - 2

**Q. Add Matrices**

import java.util.Scanner;

public class MatrixAddition {

    public static void main(String[] args){

        Scanner scan = new Scanner(System.in);

        System.out.print("Enter Number Of Rows : ");

        byte row = scan.nextByte();

        System.out.print("Enter Number Of Columns : ");

        byte col = scan.nextByte();

        int[][] matrixOne = new int[row][col];

        int[][] matrixTwo = new int[row][col];

        System.out.println("Matrix 1");

        getMatrix(matrixOne,row,col);

        System.out.println("Matrix 2");

        getMatrix(matrixTwo,row,col);

        addAndDisplayMatrix(matrixOne, matrixTwo, row, col);

    }

    private static void getMatrix(int[][] matrix,byte row,byte col) {

        Scanner scan = new Scanner(System.in);

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

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

                System.out.print("Enter Number for Matrix [" + i + "][" + j + "] : ");

                matrix[i][j] = scan.nextInt();

            }

        }

    }

    private static void addAndDisplayMatrix(int[][] matrix1,int[][] matrix2,byte row,byte col) {

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

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

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

                System.out.print(matrix1[i][j]+matrix2[i][j] + " ");

            }

            System.out.println();

        }

    }

}

**Output :**

**PS D:\MCA\MCA SEM 3\JAVA\Assignment 2> java .\MatrixAddition.java**

**Enter Number Of Rows : 3**

**Enter Number Of Columns : 3**

**Matrix 1**

**Enter Number for Matrix [0][0] : 5**

**Enter Number for Matrix [0][1] : 4**

**Enter Number for Matrix [0][2] : 3**

**Enter Number for Matrix [1][0] : 6**

**Enter Number for Matrix [1][1] : 3**

**Enter Number for Matrix [1][2] : 8**

**Enter Number for Matrix [2][0] : 1**

**Enter Number for Matrix [2][1] : 3**

**Enter Number for Matrix [2][2] : 5**

**Matrix 2**

**Enter Number for Matrix [0][0] : 3**

**Enter Number for Matrix [0][1] : 2**

**Enter Number for Matrix [0][2] : 5**

**Enter Number for Matrix [1][0] : 6**

**Enter Number for Matrix [1][1] : 4**

**Enter Number for Matrix [1][2] : 7**

**Enter Number for Matrix [2][0] : 1**

**Enter Number for Matrix [2][1] : 0**

**Enter Number for Matrix [2][2] : 2**

**Final Matrix**

**8 6 8**

**12 7 15**

**2 3 7**

**Q. Get Environment Variables**

import java.lang.\*;

public class GetEnvironmentVariable {

    public static void main(String[] args)

    {

        System.out.println("\nEnvironment variable USERNAME: ");

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

        System.out.println("\nEnvironment variable TEMP: ");

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

        System.out.println("\nEnvironment variable PATH: ");

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

    }

}

**Output:**

**PS D:\MCA\MCA SEM 3\JAVA\Assignment 2> java .\GetEnvironmentVariable.java**

**Environment variable USERNAME:**

**Xtrem**

**Environment variable TEMP:**

**C:\Users\Xtrem\AppData\Local\Temp**

**Environment variable PATH:**

**C:\Windows\system32;C:\Windows;C:\Windows\System32\Wbem;C:\Windows\System32\WindowsPowerShell\v1.0\;C:\Windows\System32\OpenSSH\;C:\Program Files (x86)\NVIDIA Corporation\PhysX\Common;C:\Program Files\NVIDIA Corporation\NVIDIA NvDLISR;C:\Program Files\Java\jdk-13.0.1\bin;C:\Program Files\Git\cmd;C:\Users\Xtrem\AppData\Local\Microsoft\WindowsApps;C:\Users\Xtrem\AppData\Local\Programs\Microsoft VS Code\bin;C:\Program Files\Java\jdk-13.0.1\bin;**

**Q. Circle**

public class Circle{

    public static void main(String[] args)

    {

        double x,y,r;

        int origin;

        CircleObj[] c= new CircleObj[10];

        for(int i=0;i<10;i++)

        {

            origin=(int) (2\*  Math.random());

            r= (double) (10\* Math.random())+1;

            if(origin==1)

                c[i]=new CircleObj(r);

            else

            {

                x= (double) (100\* Math.random())+1;

                y= (double) (100\* Math.random())+1;

                c[i] = new CircleObj(x,y,r);

            }

        }

        for(int i=0;i<10;i++)

        {

            c[i].display();

        }

    }

}

public class CircleObj

{

    double radius,cntx,cnty;

    CircleObj(double radius)

    {

        this.cntx=0;

        this.cnty=0;

        this.radius=radius;

    }

    CircleObj(double cntx,double cnty,double radius)

    {

        this.cntx=cntx;

        this.cnty=cnty;

        this.radius=radius;

    }

    private double area()

    {

        return (3.14\*this.radius\*this.radius);

    }

    public void display()

    {

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

        System.out.println("Center: ("+this.cntx+", "+this.cnty+")");

        System.out.println("Radius: "+this.radius);

        System.out.println("Area: "+ this.area());

    }

}

**Output :**

**PS D:\MCA\MCA SEM 3\JAVA\Assignment 2> java .\Circle.java**

**-------**

**Center: (71.78505496544743, 17.213464081660057)**

**Radius: 10.101627819491677**

**Area: 320.4146576550785**

**-------**

**Center: (0.0, 0.0)**

**Radius: 5.0617069059482045**

**Area: 80.44955315741257**

**-------**

**Center: (0.0, 0.0)**

**Radius: 10.362498572521107**

**Area: 337.1775227296762**

**-------**

**Center: (89.67442172582656, 67.80023035725446)**

**Radius: 4.762193006142564**

**Area: 71.21043419514488**

**-------**

**Center: (47.914239717727405, 25.360860928995752)**

**Radius: 5.406426040616273**

**Area: 91.78044955253277**

**-------**

**Center: (0.0, 0.0)**

**Radius: 10.088821343762639**

**Area: 319.6027525739722**

**-------**

**Center: (0.0, 0.0)**

**Radius: 10.6691310947075**

**Area: 357.4273251124111**

**-------**

**Center: (0.0, 0.0)**

**Radius: 1.1219839322884109**

**Area: 3.9527825451439673**

**-------**

**Center: (0.0, 0.0)**

**Radius: 3.6632745450908812**

**Area: 42.13748243311192**

**-------**

**Center: (0.0, 0.0)**

**Radius: 3.817459601092299**

**Area: 45.759213110751375**

**Q. Remove duplicate characters from the string.**

import java.util.Arrays;

public class RemoveDuplicate {

    public static void main(String[] args) {

        char[] myString = "Pradip Karmakar".toCharArray();

        System.out.println(StringSlash(myString));

    }

    private static String StringSlash(char[] myString) {

        int index = 0;

        int j;

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

            for (j = 0; j < i; j++) {

                if (myString[i] == myString[j])

                {

                    break;

                }

            }

            if(j == i){

                myString[index++] = myString[i];

            }

        }

        return String.valueOf(Arrays.copyOf(myString, index));

    }

}

**Output :**

**PS D:\MCA\MCA SEM 3\JAVA\Assignment 2> java .\RemoveDuplicate.java**

**Pradip Kmk**

**Q. Write a program to print the pattern using for loop**

**1 1 1**

**2 4 8**

**...........................**

**10 100 1000**

public class SquareCube {

    public static void main(String[] args){

        for (int i = 1; i < 11; i++) {

            for (int j = 1; j < 4; j++) {

                System.out.print((int)Math.pow(i, j) + "  ");

            }

            System.out.println();

        }

    }

}

**Output:**

**PS D:\MCA\MCA SEM 3\JAVA\Assignment 2> java .\SquareCube.java**

**1 1 1**

**2 4 8**

**3 9 27**

**4 16 64**

**5 25 125**

**6 36 216**

**7 49 343**

**8 64 512**

**9 81 729**

**10 100 1000**

**Q. Write a java program to accept distance in meters, time in hours ,minutes and seconds .Calculate the speed in m/sec ,km/hr and miles/hr .(1 mile=1609 m).**

import java.util.Scanner;

public class Speed {

    public static void main(String[] args) {

        Scanner scan = new Scanner(System.in);

        float seconds;

        float minutePerSecond;

        float kiloMeterPerHour;

        float milePerHour;

        System.out.print("Enter distance in meters : ");

        float distance = scan.nextFloat();

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

        float hr = scan.nextFloat();

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

        float min = scan.nextFloat();

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

        float sec = scan.nextFloat();

        seconds = (hr\*3600) + (min\*60) + sec;

        minutePerSecond = distance / seconds;

        kiloMeterPerHour = ( distance/1000.0f ) / ( seconds/3600.0f );

        milePerHour = kiloMeterPerHour / 1.609f;

        System.out.println("Meter/Seconds : " + minutePerSecond);

        System.out.println("KM/H : " + kiloMeterPerHour);

        System.out.println("Miles / Hour : " + milePerHour);

    }

}

**Output :**

**PS D:\MCA\MCA SEM 3\JAVA\Assignment 2> java .\Speed.java**

**Enter distance in meters : 80000**

**Enter hour: 1**

**Enter minutes: 30**

**Enter seconds: 45**

**Meter/Seconds : 14.692378**

**KM/H : 52.89256**

**Miles / Hour : 32.87294**

**Q. Display Java detail**

public class JavaInfo {

    public static void main(String[] args){

        System.out.println("Java Version : " + System.getProperty("java.version"));

        System.out.println("Runtime Version : " + System.getProperty("java.runtime.version"));

        System.out.println("Java Vendor Name : " + System.getProperty("java.vendor"));

        System.out.println("Java Vendor URL : " + System.getProperty("java.vendor.url"));

    }

}

**Output :**

**PS D:\MCA\MCA SEM 3\JAVA\Assignment 2> java .\JavaInfo.java**

**Java Version : 13.0.1**

**Runtime Version : 13.0.1+9**

**Java Vendor Name : Oracle Corporation**

**Java Vendor URL :** [**https://java.oracle.com/**](https://java.oracle.com/)

**Q. Create an abstract class “Monster” which is extended by three classes “Ware wolf “,”Zombie” and “Vampire” .Create 6 types of Monsters in single dimension array and print them.**

public class MonstersObject

{

    public static void main(String[] args)

    {

        int key;

        Monster[] m= new Monster[6];

        for(int i=0;i<6;i++)    //to create

        {

            key=(int) (3 \* Math.random()) +1;   //to create random Monsters

            switch(key)

            {

                case 1: //For Wolf

                    m[i]= new WareWolf(i);

                    break;

                case 2: //for Zombie

                    m[i]= new Zombie(i);

                    break;

                case 3: //for Vampire

                    m[i]= new Vampire(i);

            }

        }

        for(int i=0;i<6;i++)

        {

            m[i].display();

        }

    }

}

public class MonsterObject {

}

abstract class Monster

{

    String type;

    int id;

    Monster(String type)

    {

        this.type=type;

    }

    public void display()

    {

        System.out.println("-----");

        System.out.println("Monster id: "+ this.id);

        System.out.println("Monster type: "+ this.type);

    }

}

class WareWolf extends Monster

{

    WareWolf(int id)

    {

        super("WareWolf");

        this.id=id;

    }

}

class Zombie extends Monster

{

    Zombie(int id)

    {

        super("Zombie");

        this.id=id;

    }

}

class Vampire extends Monster

{

    Vampire(int id)

    {

        super("Vampire");

        this.id=id;

    }

}

**Output :**

**PS D:\MCA\MCA SEM 3\JAVA\Assignment 2> java .\MonsterObject.java**

**-----**

**Monster id: 0**

**Monster type: Vampire**

**-----**

**Monster id: 1**

**Monster type: WareWolf**

**-----**

**Monster id: 2**

**Monster type: Vampire**

**-----**

**Monster id: 3**

**Monster type: Zombie**

**-----**

**Monster id: 4**

**Monster type: Vampire**

**-----**

**Monster id: 5**

**Monster type: Vampire**

**Q. Write a program to accept a string from user and a point from where you want to print next three**

**words. Print the old and the new string.**

**Old String = “The quick brown fox jumps over the lazy dog”**

**New String = “brown fox jumps”**

import java.util.Scanner;

public class StringSplit {

    public static void main(String[] args)

    {

    int index;

    int tmp;

    int count=0;

    String old;

    String New;

    String token;

    Scanner s = new Scanner(System.in);

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

    old=s.nextLine();

    System.out.print("Enter from which word you want to split: ");

    token=s.nextLine();

    index= old.indexOf(token);

    tmp=index;

    while(count<3 && tmp != -1)

    {

        tmp=old.indexOf(" ",tmp+1);

        count++;

    }

    New=old.substring(index,tmp);

    System.out.println("Old string= " + old);

    System.out.println("New string= " + New);

    }

}

**Output:**

**PS D:\MCA\MCA SEM 3\JAVA\Assignment 2> java .\StringSplit.java**

**Enter String: Pradip Karmakar Study in Rollwala Computer Center**

**Enter from which word you want to split: Karmakar**

**Old string= Pradip Karmakar Study in Rollwala Computer Center**

**New string= Karmakar Study in**

**Q. Convert Minutes into years and days**

**For Eg. Input => 3456789**

**Output => 6 Years and 210 days**

import java.util.Scanner;

public class MinuteToYearDay {

    final static int HOUR = 24;

    final static int DAYSINYEAR = 365;

    final static int MINUTES = 60;

    public static void main(String[] args){

        Scanner scan = new Scanner(System.in);

        System.out.print("Enter Values as Minutes : ");

        long minute = scan.nextLong();

        int year = (int) (minute / ((MINUTES\*HOUR)\*DAYSINYEAR));

        int remainingminute = (int) (minute % ((MINUTES\*HOUR)\*DAYSINYEAR));

        int days = remainingminute / (MINUTES \* HOUR);

        System.out.println(year + " Years and " + days + " Days.");

    }

}

**Output :**

**PS D:\MCA\MCA SEM 3\JAVA\Assignment 2> java .\MinuteToYearDay.java**

**Enter Values as Minutes : 12000000**

**22 Years and 303 Days.**

**Q. Generate Invoice**

public class Invoice {

    public static void main(String[] args) {

        Inventory items[] = { new Inventory(101, "Part 1", 2000), new Inventory(102, "Part 2", 200),

                new Inventory(103, "Part 3", 1300), new Inventory(104, "Part 4", 3200), new Inventory(105, "Part 5", 5000) };

        Bill\_Item bitem = new Bill\_Item(items.length);

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

            bitem.add(items[i], 1);

        }

        bitem.display();

    }

}

class Inventory {

    long itemId;

    String description;

    double price;

    int qty;

    Inventory() {

        itemId = 0;

        description = "";

        price = 0;

    }

    Inventory(long itemId, String description, double price) {

        this.itemId = itemId;

        this. description = description;

        this.price = price;

    }

    public void display() {

        System.out.println("Item id : " + itemId);

        System.out.println("\t Description : " + description);

        System.out.println("\t Price " + price);

        System.out.println("\t qty " + qty);

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

    }

}

class Bill\_Item {

    int qty;

    double amount;

    Inventory items[];

    int index = 0;

    Bill\_Item(int length) {

        items = new Inventory[length];

        index = 0;

    }

    public void add(Inventory it, int qty) {

        this.amount = it.price \* qty;

        it.qty += qty;

        items[index] = it;

        index++;

    }

    public void display() {

        int total = 0;

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

            items[i].display();

            total += items[i].qty \* items[i].price;

            System.out.println();

        }

        System.out.print("Amount : " + total);

    }

}

**Output :**

**PS D:\MCA\MCA SEM 3\JAVA\Assignment 2> java .\Invoice.java**

**Item id : 101**

**Description : Part 1**

**Price 2000.0**

**qty 1**

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

**Item id : 102**

**Description : Part 2**

**Price 200.0**

**qty 1**

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

**Item id : 103**

**Description : Part 3**

**Price 1300.0**

**qty 1**

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

**Item id : 104**

**Description : Part 4**

**Price 3200.0**

**qty 1**

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

**Item id : 105**

**Description : Part 5**

**Price 5000.0**

**qty 1**

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

**Amount : 11700**

**Q. Count Words and Print in Reverse**

import java.util.Scanner;

public class WordReverse {

    public static void main(String[] args) {

        String input;

        Scanner scan = new Scanner(System.in);

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

        input = scan.nextLine();

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

        System.out.println("Count of words in String : " + token.length);

        for (int i = token.length - 1; i >= 0; i--) {

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

        }

    }

}

**Output :**

**PS D:\MCA\MCA SEM 3\JAVA\Assignment 2> java .\WordReverse.java**

**Enter String :**

**Pradip Karmakar Study at Rollwala**

**Count of words in String : 5**

**Rollwala at Study Karmakar Pradip**

**Q. Binary Addition**

import java.util.Scanner;

public class BinaryAdd {

   public static void main(String[] args) {

        long b1, b2;

        int i = 0, carry = 0;

        int[] sum = new int[10];

        Scanner scanner = new Scanner(System.in);

        System.out.print("Enter first binary number: ");

        b1 = scanner.nextLong();

        System.out.print("Enter second binary number: ");

        b2 = scanner.nextLong();

        scanner.close();

        while (b1 != 0 || b2 != 0)

        {

            sum[i++] = (int)((b1 % 10 + b2 % 10 + carry) % 2);

            carry = (int)((b1 % 10 + b2 % 10 + carry) / 2);

            b1 = b1 / 10;

            b2 = b2 / 10;

        }

        if (carry != 0) {

            sum[i++] = carry;

        }

        --i;

        System.out.print("Output: ");

        while (i >= 0) {

            System.out.print(sum[i--]);

        }

        System.out.print("\n");

   }

}

**Output :**

**PS D:\MCA\MCA SEM 3\JAVA\Assignment 2> java .\BinaryAdd.java**

**Enter first binary number: 1010**

**Enter second binary number: 1100**

**Output: 10110**

**Q. Print Java**

public class PrintJava {

    public static void main(String[] args) {

        System.out.println("    JJ       A     V       V     A");

        System.out.println("    JJ      A A     V     V     A A");

        System.out.println("    JJ     A   A     V   V     A   A");

        System.out.println("    JJ    AAAAAAA     V V     AAAAAAA");

        System.out.println(" JJJJJ   A       A     v     A       A");

    }

}

**Output :**

**PS D:\MCA\MCA SEM 3\JAVA\Assignment 2> java .\PrintJava.java**

**JJ A V V A**

**JJ A A V V A A**

**JJ A A V V A A**

**JJ AAAAAAA V V AAAAAAA**

**JJJJJ A A v A A**

**Q. Print Face**

public class PrintFace {

    public static void main(String[] args) {

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

        System.out.println("[| 0   0 |]");

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

        System.out.println(" |  '\_'  | ");

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

    }

}

**Output :**

**PS D:\MCA\MCA SEM 3\JAVA\Assignment 2> java .\PrintFace.java**

**+"""""""+**

**[| 0 0 |]**

**| ^ |**

**| '\_' |**

**+----------+**

**Q. Print Pattern**

public class OneZeroPattern {

    public static void main(String[] args) {

        int setter = 1;

        int numberOflines = 9;

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

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

                System.out.print(setter);

                if(setter == 1) setter = 0;

                else setter = 1;

            }

            System.out.println("");

        }

    }

}

**Output :**

**PS D:\MCA\MCA SEM 3\JAVA\Assignment 2> java .\OneZeroPattern.java 5**

**1**

**01**

**010**

**1010**

**10101**

**010101**

**0101010**

**10101010**

**101010101**

**Q. 14)Create an employee class which has attributes : empid , fname , lname, salary, designation, bonus. It has all the necessary constructors (at least 2) and methods.It must have one abstract method calculate\_bonus().**

public class Data {

    public static void main(String[] args) {

        Employee emp1 = new Manager(101, "Pradip", "Karmakar", 30\_00\_0F, "Product Manager", 6);

        Employee emp2 = new Clerk(201, "Sudip", "Karmakar", 10\_00\_0F, "Production", 300);

        Employee emp3 = new Manager(102, "Sandip", "Karmakar", 23\_00\_0F, "Sales Manager", 2);

        Employee emp4 = new Clerk(202, "Jeet", "Karmakar", 30\_00\_0F, "Sales", 240);

        Employee emp5 = new Manager(103, "Samar", "Karmakar", 40\_00\_0F, "Human Resource Manager", 12);

        emp1.CalculateBonus();

        emp1.PrintDetail();

        emp2.CalculateBonus();

        emp2.PrintDetail();

        emp3.CalculateBonus();

        emp3.PrintDetail();

        emp4.CalculateBonus();

        emp4.PrintDetail();

        emp5.CalculateBonus();

        emp5.PrintDetail();

    }

}

abstract class Employee {

    int empid;

    String fname;

    String lname;

    float salary;

    String designation;

    float bonus;

    public Employee() {

        empid = 0;

        fname = "Not Specified";

        lname = "Not Specified";

        salary = 0.0F;

        designation = "Not Specified";

        bonus = 0.0F;

    }

    public Employee(int empid, String fname, String lname, float salary, String designation) {

        this.empid = empid;

        this.fname = fname;

        this.lname = lname;

        this.salary = salary;

        this.designation = designation;

    }

    public abstract void CalculateBonus();

    public abstract void PrintDetail();

}

class Manager extends Employee {

    int noOfProjectsHandled;

    Manager() {

        super();

    }

    Manager(int empid, String fname, String lname, float salary, String designation, int noOfProjectsHandled) {

        super(empid, fname, lname, salary, designation);

        this.noOfProjectsHandled = noOfProjectsHandled;

    }

    public void CalculateBonus() {

            this.bonus = (noOfProjectsHandled \* 1000);

    }

    public void PrintDetail() {

        System.out.println("Employee Id : " + this.empid);

        System.out.println("Name : " + this.fname + " " + this.lname);

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

        System.out.println("Designation : " + this.designation);

        System.out.println("Total Projects Handeled : " + this.noOfProjectsHandled);

        System.out.println("Bonus : " + this.bonus);

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

    }

}

class Clerk extends Employee {

    int noOfHoursworked;

    Clerk() {

        super();

    }

    Clerk(int empid, String fname, String lname, float salary, String designation, int noOfHoursworked) {

        super(empid, fname, lname, salary, designation);

        this.noOfHoursworked = noOfHoursworked;

    }

    public void CalculateBonus() {

        if(this.noOfHoursworked > 250) {

            this.bonus = (noOfHoursworked - 250) \* 200;

        }

    }

    public void PrintDetail() {

        System.out.println("Employee Id : " + this.empid);

        System.out.println("Name : " + this.fname + " " + this.lname);

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

        System.out.println("Designation : " + this.designation);

        System.out.println("Total Hours Worked : " + this.noOfHoursworked);

        System.out.println("Bonus : " + this.bonus);

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

    }

}

**Output :**

**PS D:\MCA\MCA SEM 3\JAVA\Assignment 2> java .\Data.java**

**Employee Id : 101**

**Name : Pradip Karmakar**

**Salary : 30000.0**

**Designation : Product Manager**

**Total Projects Handeled : 6**

**Bonus : 6000.0**

**++++++++++++++++++++++++**

**Employee Id : 201**

**Name : Sudip Karmakar**

**Salary : 10000.0**

**Designation : Production**

**Total Hours Worked : 300**

**Bonus : 10000.0**

**++++++++++++++++++++++++**

**Employee Id : 102**

**Name : Sandip Karmakar**

**Salary : 23000.0**

**Designation : Sales Manager**

**Total Projects Handeled : 2**

**Bonus : 2000.0**

**++++++++++++++++++++++++**

**Employee Id : 202**

**Name : Jeet Karmakar**

**Salary : 30000.0**

**Designation : Sales**

**Total Hours Worked : 240**

**Bonus : 0.0**

**++++++++++++++++++++++++**

**Employee Id : 103**

**Name : Samar Karmakar**

**Salary : 40000.0**

**Designation : Human Resource Manager**

**Total Projects Handeled : 12**

**Bonus : 12000.0**

**++++++++++++++++++++++++**