Assignment 6

01 Aaditya Sudhir Joshi

1. Create a class MathOperation containing overloaded methods ‘multiply’ to calculate multiplication of following arguments.

a.  two integers

b.  three floats

c.  all elements of array

d. one double and one integer

Answer:

class MathOperation{

void multiply(int i, int j){

System.out.println("Multiplication of Method 1"+(i\*j));

}

void multiply(float a,float b,float c){

System.out.println("Multiplication of Method 2"+(a\*b\*c));

}

void multiply(int[] arr){

int ans=1;

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

ans = ans\*arr[i];

}

System.out.println("Multiplication of Method 3 (Array)"+ans);

}

void multiply(double a,int b){

System.out.println("Multiplication of Method 2"+(a\*b));

}

}

class mathsop{

public static void main(String[] args){

MathOperation m = new MathOperation();

int[] arr = {1,2,3,4,5};

m.multiply(5,6);

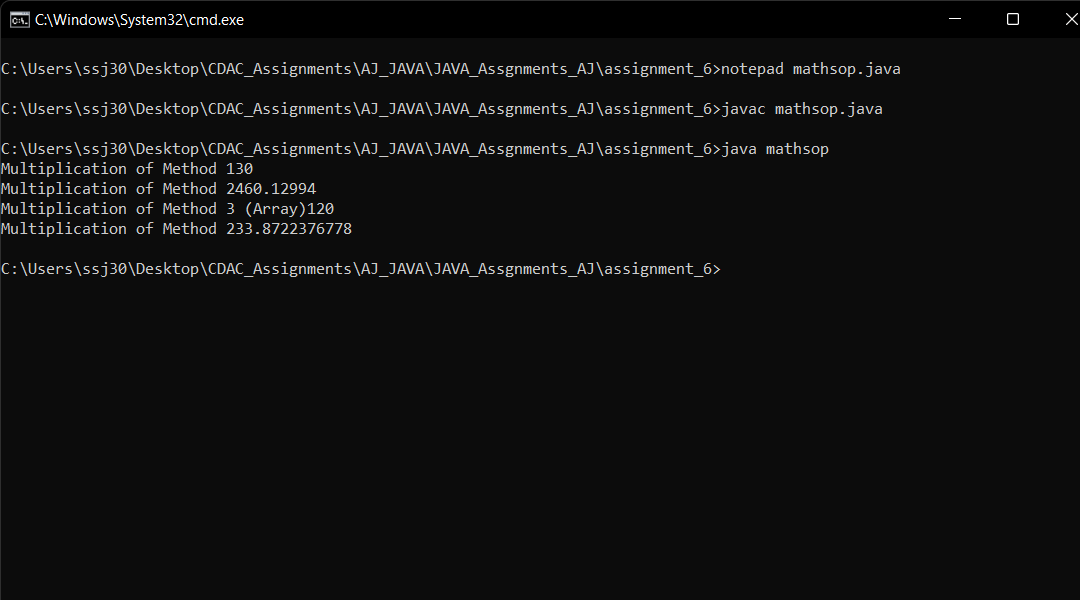
m.multiply(5.5f,9.4f,8.9f);

m.multiply(arr);

m.multiply(5.6453729463,6);

}

}



2. Create a class Person with properties (name and age) with following features.

a. Default age of person should be 18.

b. A person object can be initialized with name and age.

c. Method to display name and age of person

Create another class PersonDemo ( main class ) that demonstrates the functionalities of Person class by creating Person object and calling methods.

Answer:

class Person{

static int age=18;

String name;

Person(int age,String name){

this.age=age;

this.name =name;

}

void display(){

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

}

}

class PersonDemo{

static int age=18;

public static void main(String[] args){

Person p = new Person(22,"Aaditya");

p.display();

Person p1 = new Person(22,"Atharva");

p1.display();

Person p2 = new Person(age,"Saurabh");

p2.display();

}

}

3. Create a class Employee with three data members (empNo, salary and totalSalary) and following features.

a. Only parameterized constructor. [Do not overload the constructor]

b. totalSalary always represents salary total of all the employees created.

c. empNo should be auto incremented.

d. display total employees and totalSalary using a method.

Create another class EmployeeDemo (main class) that creates some Employee objects and calls Employee method to display no. of employees and total of their salaries.

Answer:

import java.util.\*;

class Employee{

int empNo;

double salary;

double totalsalary=0.0;

Employee(int empNo,double salary,double totalsalary){

this.empNo = empNo;

this.salary = salary;

this.totalsalary = totalsalary;

}

void display(){

System.out.println("Employee NO: "+empNo+", And Total Salary till now "+totalsalary);

}

}

class EmployeeDemo{

public static void main(String[] args){

Scanner scn = new Scanner(System.in);

System.out.println("Enter the counts Employees you want to insert :");

int m = scn.nextInt();

Employee[] emp = new Employee[m];

int empNo=1;

double totalsalary=0.0;

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

System.out.println("Enter the salary of employee having id : "+empNo);

double salary = scn.nextDouble();

totalsalary = totalsalary + salary;

emp[i] = new Employee(empNo,salary,totalsalary);

empNo++;

}

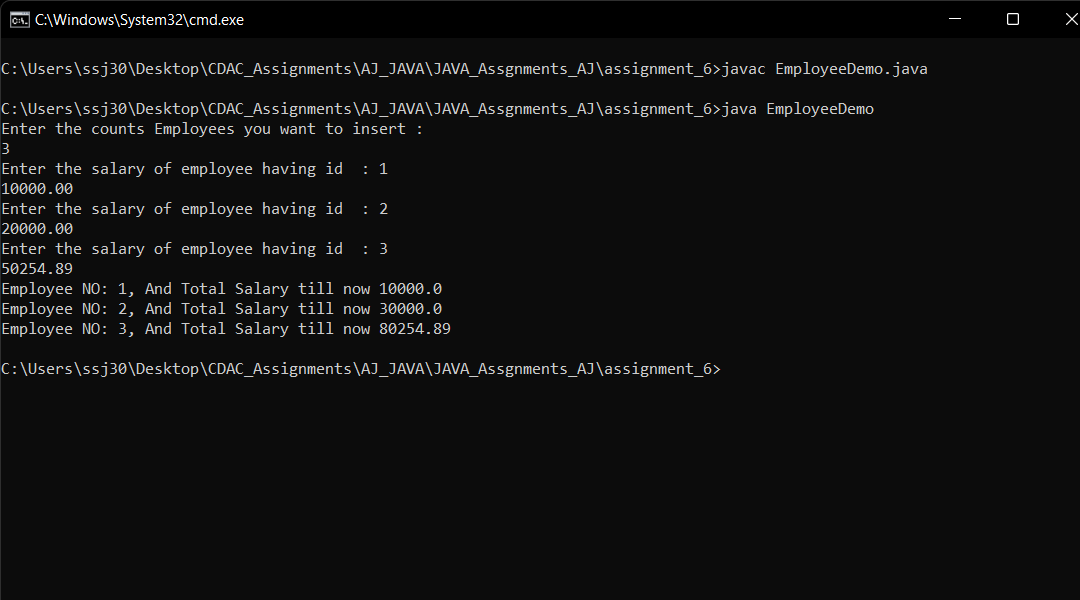
for(Employee e:emp){

e.display();

}

}

}



4. Create class Product with three data members (pid, price, quantity) and parameterized constructor that takes values for all three data members.   Create a main method in different class (say ProductDemo) and perform

following task:

a. Accept information for five Product objects from user and store objects in an array

b. Find pid of product with highest price.

c. Create a static method (with array of product’s object as argument) in

Product class to calculate and return total amount spent on all products. (

amount spent on single product = price of product \* quantity of product )

Answer:

import java.util.\*;

class Product{

int pid;

int price;

int quantity;

void set(int pid,int price,int quantity){

this.pid=pid;

this.price=price;

this.quantity=quantity;

}

int findout(Product[] arr){

int max=0;

int id=0;

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

if(arr[i].price > max){

max = arr[i].price;

id=arr[i].pid;

}

}

return id;

}

static int calculate(Product[] arr){

int sum=0;

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

sum = sum + (arr[i].price\*arr[i].quantity);

}

return sum;

}

}

class ProductDemo{

public static void main(String[] args){

Scanner scn = new Scanner(System.in);

System.out.println("Enter the Number of records : ");

int n = scn.nextInt();

Product[] pr = new Product[n];

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

System.out.println("Enter PID, Price, Quantity");

int pid = scn.nextInt();

int price = scn.nextInt();

int qun = scn.nextInt();

pr[i] = new Product();

pr[i].set(pid,price,qun);

}

Product p = new Product();

System.out.println("Highest Price of from Products : "+p.findout(pr));

System.out.println("Total amount spend : "+p.calculate(pr));

}

}



5. Create a class Student having data members name, roll no., age and score. Write a program to accept 10 records of student and store them in an array. And then arrange the student records based on the score group [0-50], [50- 65], [65-80], [80-100].

Answer:

//Without sorting from main class …..

//With sorting inside Student class……

//either use sorting Method or use sortbyon method

import java.util.\*;

class Student{

String name;

int roll\_no;

int age;

int score;

void set(String name,int roll\_no,int age, int score){

this.name = name;

this.roll\_no = roll\_no;

this.age = age;

this.score = score;

}

void sorting(Student[] arr){

int n=arr.length;

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

for(int j=i+1;j<n;j++){

if(arr[i].score > arr[j].score){

Student temp = arr[i];

arr[i] = arr[j];

arr[j]= temp;

}

}

}

}

void sortbycon(Student[] arr){

int n=arr.length;

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

if(arr[i].score >80){

System.out.println("Students who scored above 80 "+ + arr[i].roll\_no+" | "+arr[i].name+" | "+ arr[i].score);

}else if(arr[i].score > 65 && arr[i].score < 80){

System.out.println("Students who scored between 65 to 80 "+ + arr[i].roll\_no+" | "+arr[i].name+" | "+ arr[i].score);

}else if(arr[i].score > 50 && arr[i].score < 65){

System.out.println("Students who scored between 50 to 65 "+ + arr[i].roll\_no+" | "+arr[i].name+" | "+ arr[i].score);

}else if(arr[i].score < 50){

System.out.println("Students who scored below 50 "+ arr[i].roll\_no+" | "+arr[i].name+" | "+ arr[i].score);

}

}

}

void display(Student[] arr){

int n=arr.length;

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

System.out.println("Roll No "+arr[i].roll\_no+" Name "+arr[i].name+" age "+arr[i].age+" score "+arr[i].score);

}

}

}

class StudentDemo{

public static void main(String[] args){

Scanner scn = new Scanner(System.in);

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

int n = scn.nextInt();

Student[] s = new Student[n];

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

System.out.println("Enter the - name - roll no - age - score");

String name = scn.next();

int roll\_no = scn.nextInt();

int age = scn.nextInt();

int score = scn.nextInt();

s[i] = new Student();

s[i].set(name,roll\_no,age,score);

}

Student stu = new Student();

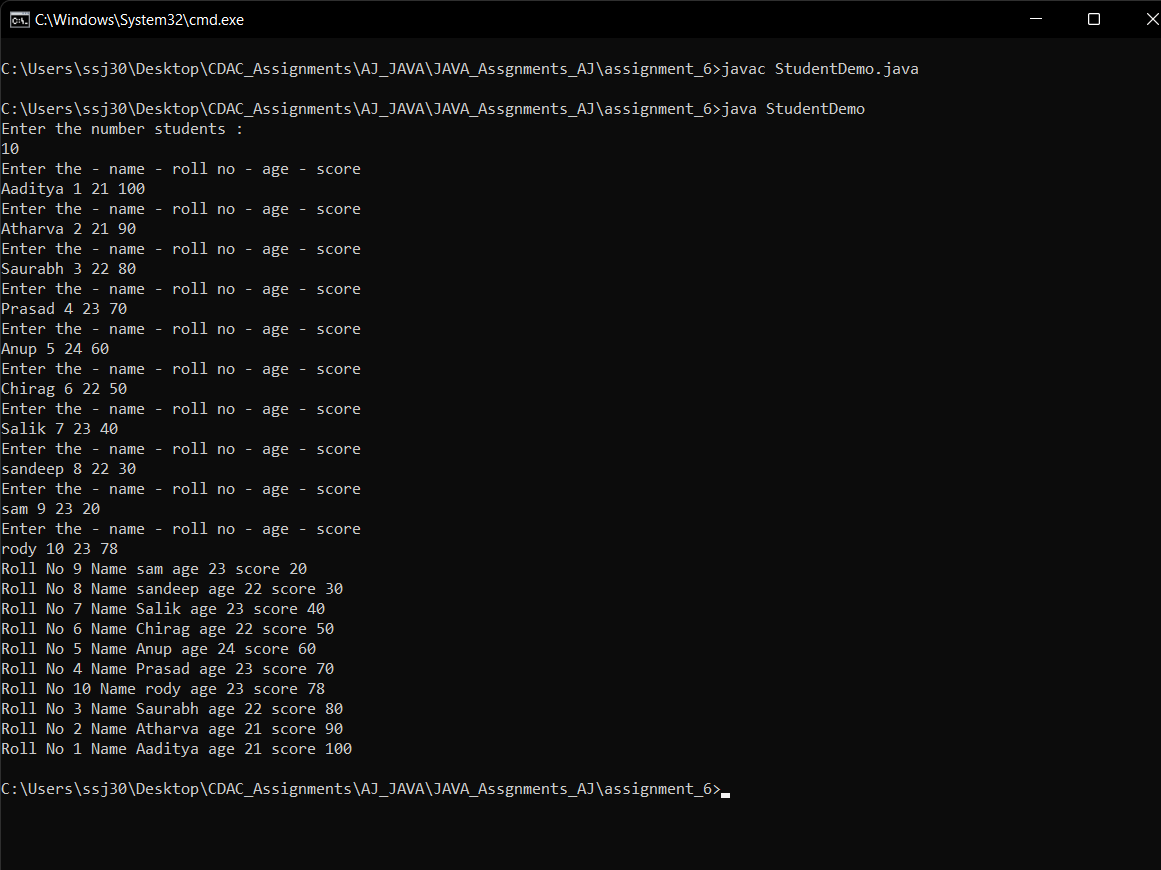
stu.sorting(s);

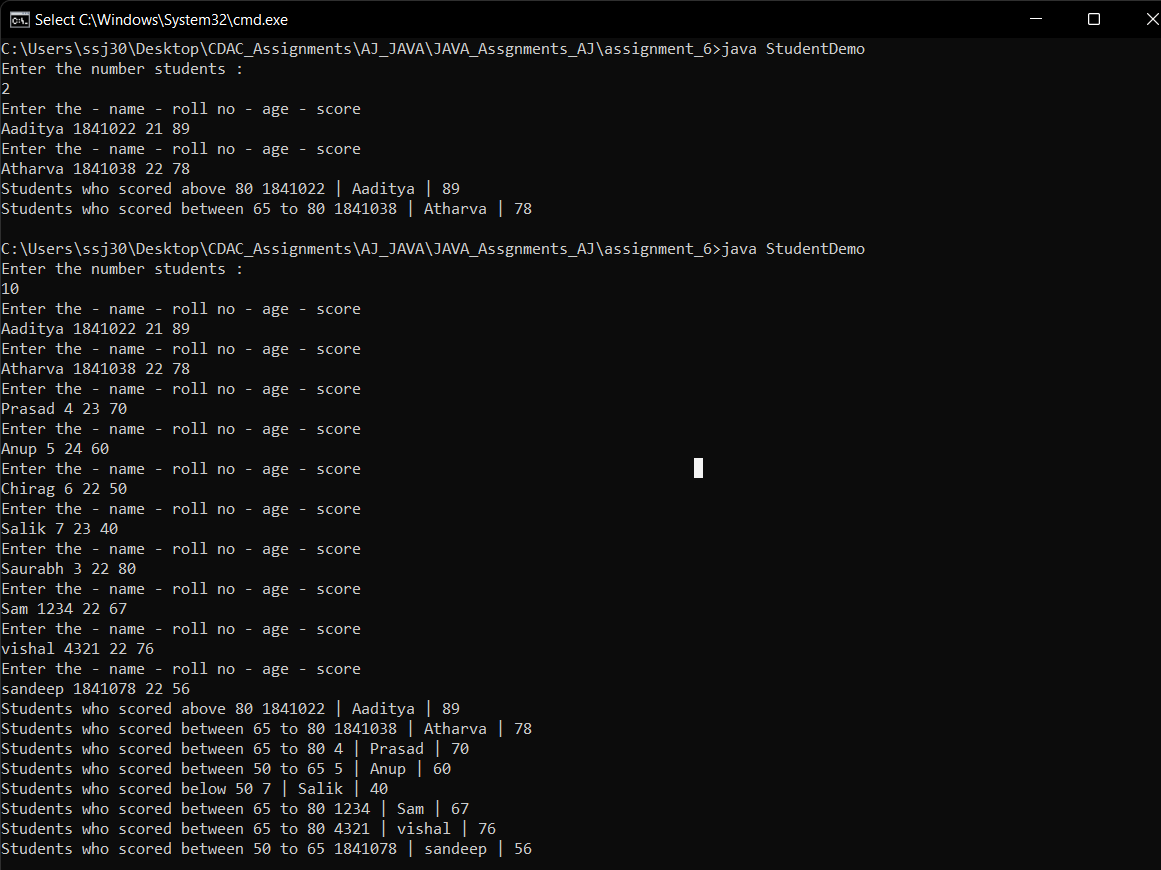
stu.display(s);

stu.sortbycon(s);

}

}





6. Write a program to demonstrate this() construct functionality.

Answer:

class Engineer{

int id;

String name;

Engineer(){

this(1,"Sam");

System.out.println("Welcome Engineer");

}

Engineer(int id , String name){

System.out.println("Welcome to our Application");

this.id=id;

this.name=name;

}

void display(){

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

}

}

class thisconstruct{

public static void main(String[] args){

Engineer e = new Engineer();

e.display();

}

}

**AND**

class Engineer{

int id;

String name;

Engineer(){

System.out.println("Welcome Engineer");

}

Engineer(int id , String name){

this();

System.out.println("Welcome to our Application");

this.id=id;

this.name=name;

}

void display(){

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

}

}

class thisconstruct{

public static void main(String[] args){

Engineer e = new Engineer(1,"Sam");

e.display();

}

}

