**Week 1**

**Question 1 : Write a Java program to print your name.**

**Source Code :**

public class name {

public static void main(String[] args) {

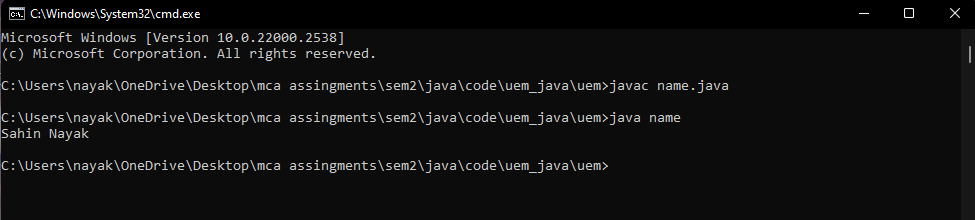
// Print Sahin Nayak

System.out.println("Sahin Nayak");

}}

**Output :**

**Question 2 : Write a Java program to add two numbers.**



**Source Code :**

import java.util.Scanner;

public class add {

public static void main(String[] args) {

Scanner sc = new Scanner(System.in);

System.out.print("Enter the first number: "); // taking input of 1st number from user

int num1 = sc.nextInt();

System.out.print("Enter the second number: "); // taking input of 2nd number from user

int num2 = sc.nextInt(); // Close the scanner to prevent resource leak

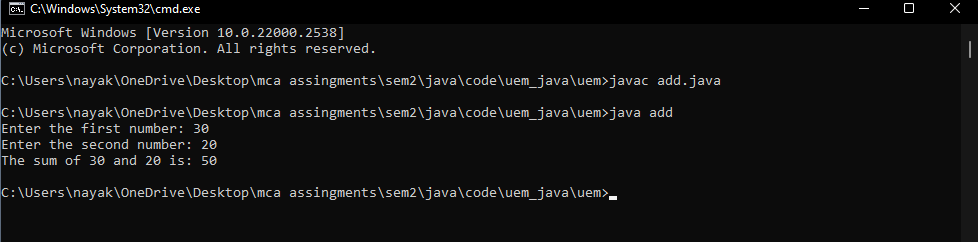
sc.close();

int sum = num1 + num2; // Add the two numbers

System.out.println("The sum of " + num1 + " and " + num2 + " is: " + sum);

}}

**Output :**



**Question 3 : Write a Java program to change temperature from Celsius to Fahrenheit.**

**Source Code :**

import java.util.Scanner;

public class Celsius\_Fahrenheit {

public static void main(String[] args) {

Scanner sc = new Scanner(System.in);

System.out.println("Enter temperature in Celsius: ");

double c = sc.nextDouble();

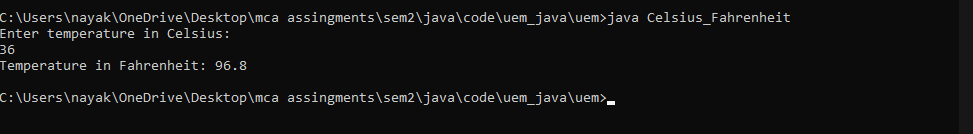
double f = (c \* 9/5) + 32; // convert the Celsius to Fahrenheit

System.out.println("Temperature in Fahrenheit: " + f);

sc.close();

}}

**Output :**



**Question 4 : Write a Java program to change temperature from Fahrenheit to Celsius.**

**Source Code :**

import java.util.Scanner;

public class Fahrenheit\_Celsius {

public static void main(String[] args) {

Scanner sc = new Scanner(System.in);

System.out.print("Enter temperature in Fahrenheit: ");

double f = sc.nextDouble();

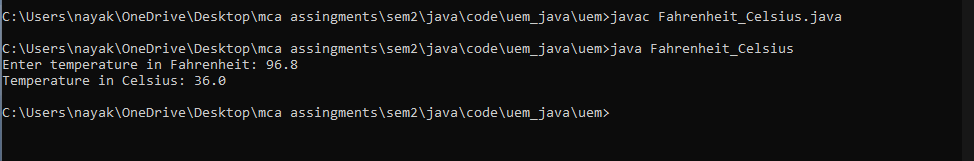
double c = (f - 32) \* 5/9; // Convert Fahrenheit to Celsius

System.out.println("Temperature in Celsius: " + c);

sc.close();

}}

**Output :**



**Question 5 : Write a Java program to find area and perimeter of a rectangle.**

**Source Code :**

import java.util.Scanner;

public class rectangle {

public static void main(String[] args) {

Scanner sc = new Scanner(System.in);

System.out.print("Enter the length of the rectangle: ");

double l = sc.nextDouble();

System.out.print("Enter the width of the rectangle: ");

double w = sc.nextDouble();

double area = l \* w;

double peri = 2 \* (l + w);

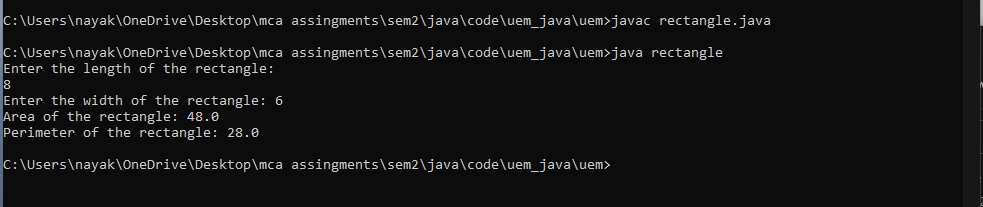
System.out.println("Area of the rectangle: " + area);

System.out.println("Perimeter of the rectangle: " + peri);

sc.close();

}}

**Output :**



**Question 6 : Write a Java program to find area and perimeter of a circle.**

**Source Code :**

import java.util.Scanner;

public class circle {

public static void main(String[] args) {

Scanner sc = new Scanner(System.in);

System.out.print("Enter the radius of the circle: ");

double r = sc.nextDouble();

double area = Math.PI \* r \* r;

double peri = 2 \* Math.PI \* r;

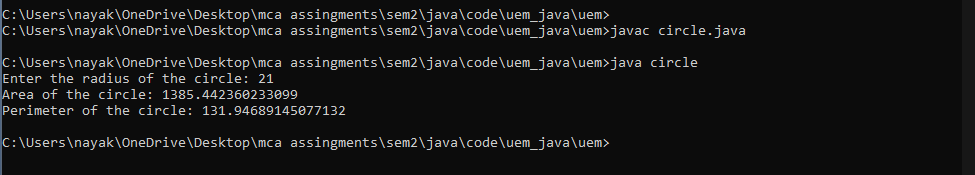
System.out.println("Area of the circle: " + area);

System.out.println("Perimeter of the circle: " + peri);

sc.close();

}}

**Output :**



**Question 7 : Write a Java Program to display whether a number is odd or even.**

**Source Code :**

import java.util.Scanner;

public class odd\_even {

public static void main(String[] args) {

Scanner sc = new Scanner(System.in);

System.out.print("Enter a number: ");

int n = sc.nextInt();

if (n % 2 == 0) { // Check if the number is odd or even

System.out.println(n + " is even.");

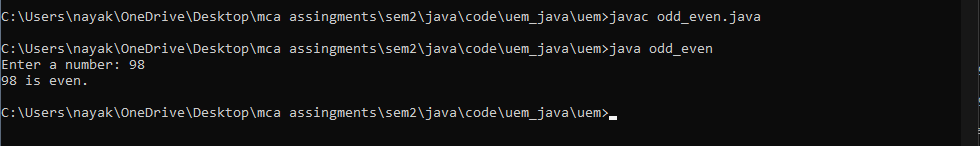
} else {

System.out.println(n + " is odd.");

}sc.close();

}}

**Output :**



**Question 8 : Write a Java Program to check if a number is Positive or Negative.**

**Source Code :**

import java.util.Scanner;

public class positive\_negative {

public static void main(String[] args) {

Scanner sc = new Scanner(System.in);

System.out.print("Enter a number: ");

int n = sc.nextInt();

if (n>0) { // Check if the number is positive or negative

System.out.println(n + " is positive.");

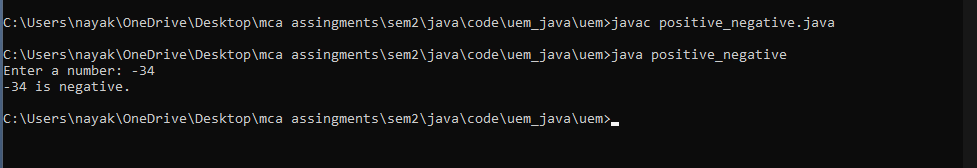
} else {

System.out.println(n + " is negative.");}

sc.close();

}}

**Output :**



**Question 9 : . Write a Java program to find maximum of three numbers.**

**Source Code :**

import java.util.Scanner;

public class max\_three {

public static void main(String[] args) {

Scanner sc = new Scanner(System.in);

System.out.println("Enter three numbers:"); // taking input of three numbers from user

System.out.print("1st number: ");

double num1 = sc.nextDouble();

System.out.print("2nd number: ");

double num2 = sc.nextDouble();

System.out.print("3rd number: ");

double num3 = sc.nextDouble();

double max = num1;

if (num2 > max) { // find the maximum of the three numbers

max = num2;

} if (num3 > max) {

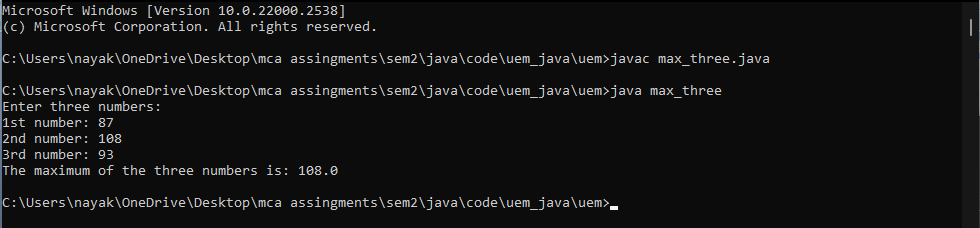
max = num3;}

System.out.println("The maximum of the three numbers is: " + max);

sc.close();

}}

**Output :**



**Question 10 : Write a Java program to swap two numbers.**

**Source Code :**

import java.util.Scanner;

public class swap{

public static void main(String[] args) {

Scanner sc = new Scanner(System.in);

System.out.print("Enter the first number: ");// take input from user

double num1 = sc.nextDouble();

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

double num2 = sc.nextDouble();

num1 = num1 + num2;

num2 = num1 - num2;

num1 = num1 - num2;

System.out.println("After swapping:");

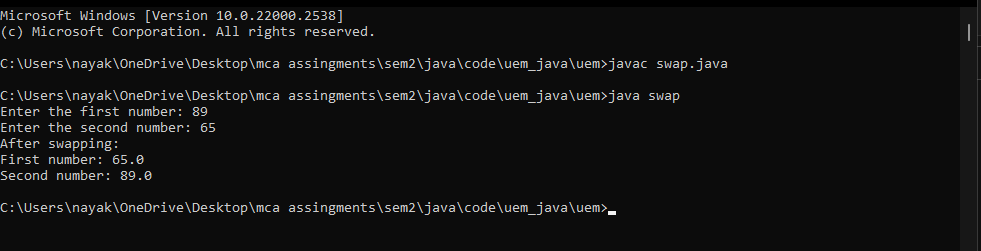
System.out.println("First number: " + num1);

System.out.println("Second number: " + num2);

sc.close();

}}

**Output :**



**Question 11 : Write a Java program to convert miles to kilometers.**

**Source Code :**

import java.util.Scanner;

public class miles\_kilometers {

public static void main(String[] args) {

Scanner sc = new Scanner(System.in);

System.out.print("Enter distance in miles: "); // take input from user of distance (in miles)

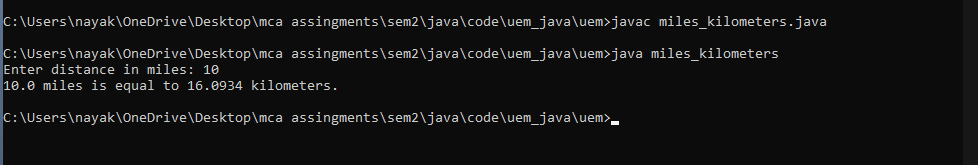
double m = sc.nextDouble();

double km = m \* 1.60934; // Convert miles to kilometers

System.out.println(m + " miles is equal to " + km + " kilometers.");

sc.close(); }}

**Output :**



**Question 12 :** **Write a Java program to check whether a year is leap year or not.**

**Source Code :**

import java.util.Scanner;

public class leapyear {

public static void main(String[] args) {

Scanner sc = new Scanner(System.in);

System.out.print("Enter a year: ");

int year = sc.nextInt();

if ((year % 4 == 0 && year % 100 != 0) || (year % 400 == 0)) {

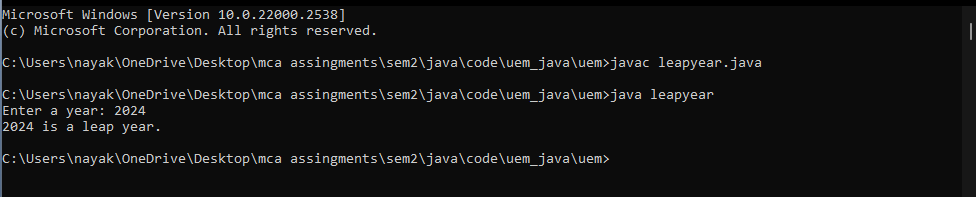
System.out.println(year + " is a leap year."); }

else {

System.out.println(year + " is not a leap year."); }

sc.close(); }}

**Output :**



**Question 13 : Write a Java program for following grading system. Note: Percentage>=90% ,Grade A Percentage>=80% ,**

**Grade B Percentage>=70% , Grade C Percentage>=60% , Grade D Percentage>=40% , Grade E**

**Percentage.**

**Source Code :**

import java.util.Scanner;

public class grade {

public static void main(String[] args) {

Scanner sc = new Scanner(System.in); // Create a Scanner object \

System.out.print("Enter the percentage: "); // take user input of percentage

double percentage = sc.nextDouble();

char grade; // Determine the grade based on the percentage

if (percentage >= 90) {

grade = 'A';

} else if (percentage >= 80) {

grade = 'B';

} else if (percentage >= 70) {

grade = 'C';

} else if (percentage >= 60) {

grade = 'D';

} else if (percentage >= 40) {

grade = 'E';

} else {

grade = 'F';

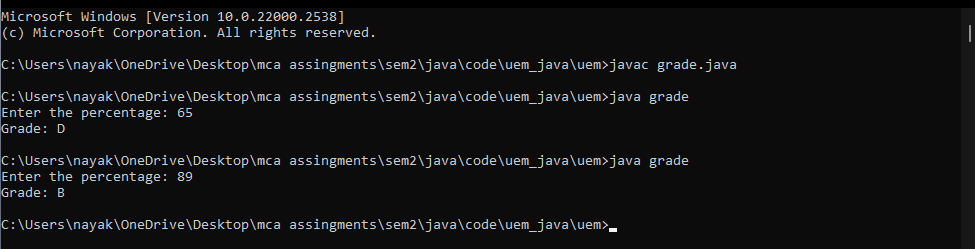
}

System.out.println("Grade: " + grade); // Print the grade

sc.close(); // Close the scanner

}}

**Output :**



**Question 14 : Write a Java program to check weather a number divisible by 5 or not**

**Source Code :**

import java.util.Scanner;

public class divisible\_by\_5 {

public static void main(String[] args) {

Scanner sc = new Scanner(System.in);

System.out.print("Enter a number: ");

int n = sc.nextInt();

// Check if the number is odd or even

if (n % 5 == 0) {

System.out.println(n + " is divisible by 5.");

} else {

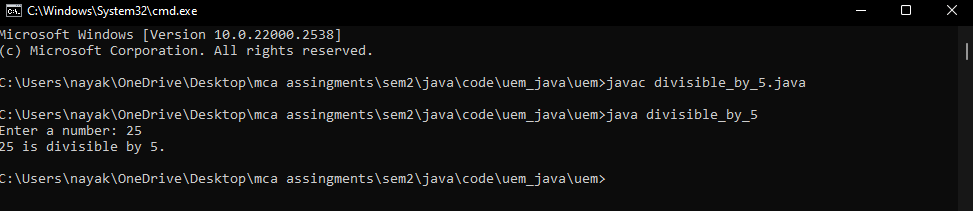
System.out.println(n + " is not divisible by 5.");

}

sc.close();

}}

**Output :**



**Week 2**

**Question 1 : Write a Java program to check whether a number is Buzz or not.**

**Source Code :**

import java.util.Scanner;

public class buzz\_number {

public static void main(String[] args) {

Scanner sc = new Scanner(System.in); // Create a Scanner object

System.out.print("Enter a number: "); // taking user input of a numbe

int number = sc.nextInt();

if ((number % 7 == 0) || (number % 10 == 7)) // Print the result {

System.out.println(number + " is a Buzz number.");

} else {

System.out.println(number + " is not a Buzz number.");

}

sc.close(); // Close the scanner

}}

**Output :**



**Question 2 : . Write a Java program to calculate factorial of 12.**

**Source Code :**

import java.util.Scanner;

public class factorial {

public static void main(String[] args) {

Scanner sc = new Scanner(System.in); // Create a Scanner object

System.out.print("Enter a number: "); // taking user input

int number = sc.nextInt();

long factorial = 1; // Calculate the factorial of the number

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

factorial \*= i;

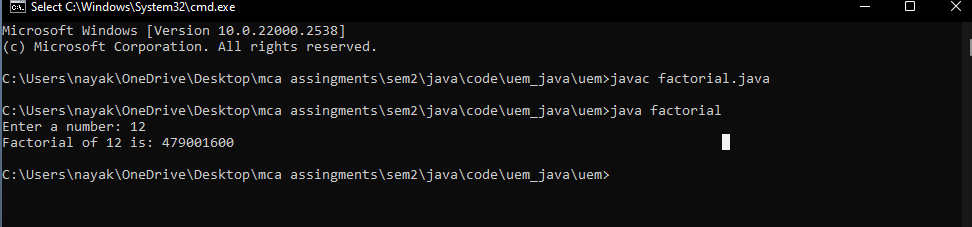
}

System.out.println("Factorial of " + number + " is: " + factorial); // Print the factorial

sc.close(); // Close the scanner

}}

**Output :**



**Question 3 :** **Write a Java program for Fibonacci series.**

**Source Code :**

import java.util.Scanner;

public class fibonacci {

public static void main(String[] args) {

Scanner sc = new Scanner(System.in); // Create a Scanner object

System.out.print("Enter the number of terms for Fibonacci series: ");

int num = sc.nextInt(); // taking range from user

int prev = 0, curr = 1; // Generate Fibonacci series

System.out.println("Fibonacci series:");

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

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

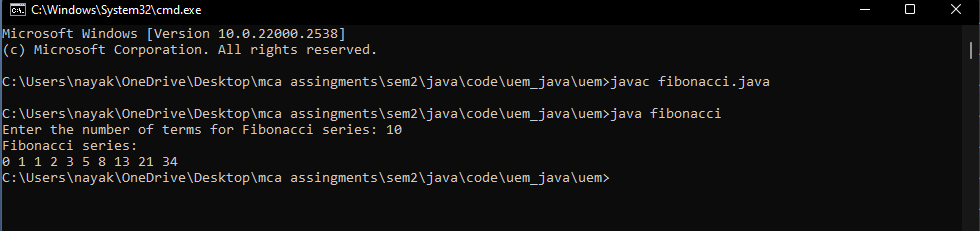
int next = prev + curr;

prev = curr;

curr = next; }

sc.close(); }} // Close the scanner

**Output :**



**Question 4 :** **Write a Java program to reverse a number**

**Source Code :**

import java.util.Scanner;

public class reverse {

public static void main(String[] args) {

Scanner sc = new Scanner(System.in); // Create a Scanner

System.out.print("Enter a number: ");

int number = sc.nextInt();

int reverse = 0; // Reverse the number

while (number != 0) {

int digit = number % 10;

reverse = reverse \* 10 + digit;

number /= 10;

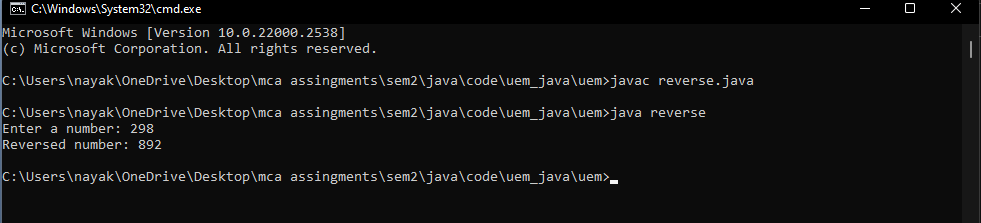
}

System.out.println("Reversed number: " + reverse); // Print the reversed number

sc.close(); // Close the scanner

}}

**Output :**



**Question 5 : Admission to a professional course is subject to the following conditions: (a) marks in Mathematics >= 60**

**(b) marks in Physics >=50**

**(c) marks in Chemistry >=40**

**(d) Total in all 3 subjects >=200 (Or) Total in Maths & Physics>=150**

**Given the marks in the 3 subjects of n (user input) students, write a program to process the applications to list the eligible candidates.**

**Source Code :**

import java.util.Scanner;

public class professional\_course\_admission {

public static void main(String[] args) {

Scanner sc = new Scanner(System.in);

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

int n = sc.nextInt(); // taking total no of strength from user

int[] math = new int[n]; // Create arrays to store marks of students in Mathematics,

int[] phy = new int[n]; // Physics, and Chemistry

int[] chem = new int[n];

for (int i = 0; i < n; i++) // Read marks of each student {

System.out.println("Enter marks of student " + (i + 1) + ":");

System.out.print("Mathematics: ");

math[i] = sc.nextInt();

System.out.print("Physics: ");

phy[i] = sc.nextInt();

System.out.print("Chemistry: ");

chem[i] = sc.nextInt();}

sc.close(); // Close the scanner

System.out.println("List of eligible candidates:"); // Process applications to list eligible candidates

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

if (math[i] >= 60 && phy[i] >= 50 && chem[i] >= 40 &&

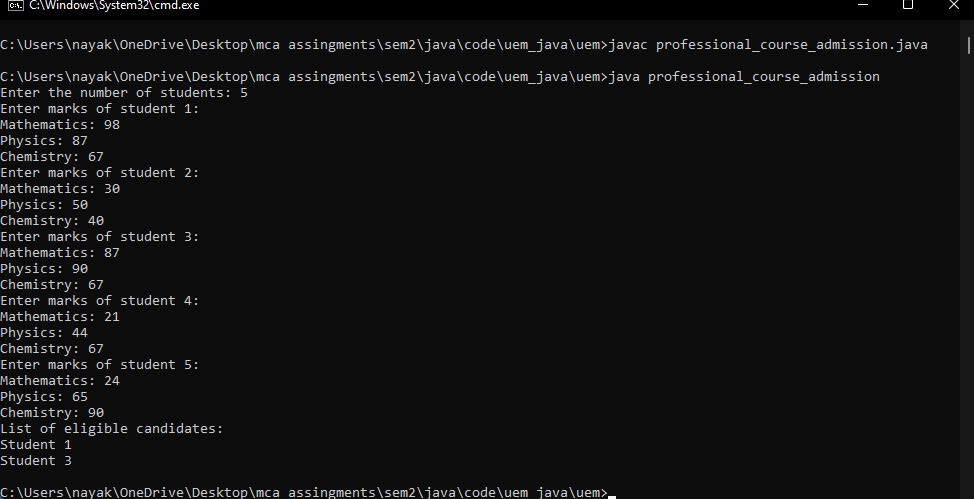
(math[i] + phy[i] + chem[i] >= 200 ||

math[i] + phy[i] >= 150)) {

System.out.println("Student " + (i + 1));

}}}}

**Output :**



**Question 6 :** **Write a Java program to find all roots of a quadratic equation**.

**Source Code :**

import java.util.Scanner;

public class quadratic {

public static void main(String[] args) {

Scanner sc = new Scanner(System.in); // Create a Scanner object to read input

System.out.println("Enter the coefficients of the quadratic equation (a, b, c):"); // Enter coefficients of the equation

double a = sc.nextDouble();

double b = sc.nextDouble();

double c = sc.nextDouble();

sc.close(); // Close the scanner

double dis = b \* b - 4 \* a \* c; // Calculate discriminant

if (dis > 0) // Check the nature of roots and calculate roots accordingly {

double root1 = (-b + Math.sqrt(dis)) / (2 \* a);

double root2 = (-b - Math.sqrt(dis)) / (2 \* a);

System.out.println("Roots are real and different.");

System.out.println("Root 1 = " + root1);

System.out.println("Root 2 = " + root2);

} else if (dis == 0) {

double root = -b / (2 \* a);

System.out.println("Roots are real and equal.");

System.out.println("Root = " + root);

} else {

double real = -b / (2 \* a);

double imag = Math.sqrt(-dis) / (2 \* a);

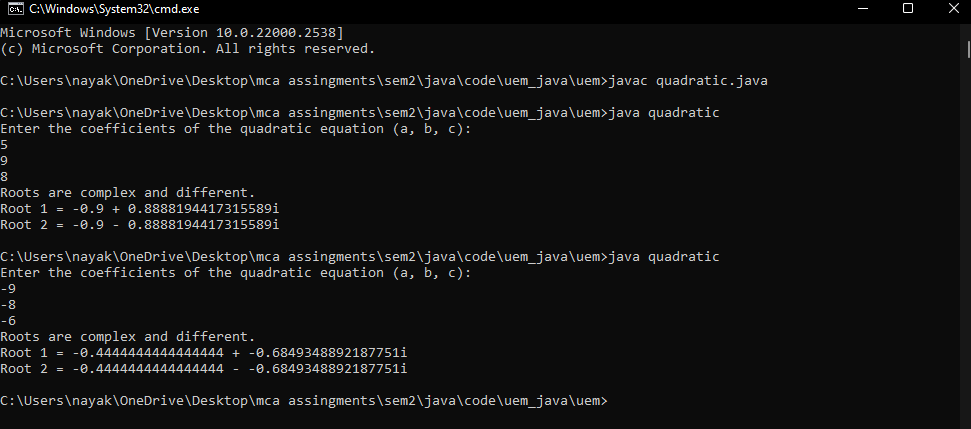
System.out.println("Roots are complex and different.");

System.out.println("Root 1 = " + real + " + " + imag + "i");

System.out.println("Root 2 = " + real + " - " + imag + "i");

}}}

**Output :**



**Question 7 : Write a Java program to calculate the sum of natural numbers up to a certain range**

**Source Code :**

import java.util.Scanner;

public class sum\_natural\_numbers {

public static void main(String[] args) {

Scanner sc = new Scanner(System.in); // Create a Scanner object

System.out.print("Enter the range (up to which natural numbers to sum): "); // Prompt the user to enter the range

int range = sc.nextInt();

sc.close(); // Close the scanner

int sum = 0; // Calculate the sum of natural numbers up to the specified range

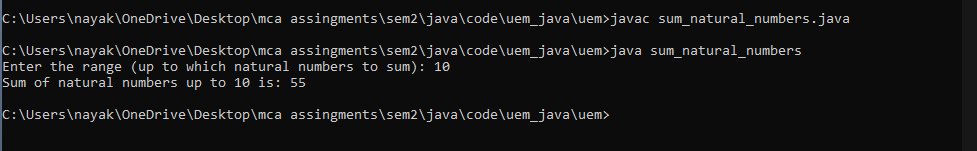
for (int i = 1; i <= range; i++) {

sum += i; }

System.out.println("Sum of natural numbers up to " + range + " is: " + sum); // Print the sum

}}

**Output :**



**Question 8 : Write a Java program to print all multiple of 10 between a given interval.**

**Source Code :**

import java.util.Scanner;

public class multiples\_of\_ten {

public static void main(String[] args) {

Scanner sc = new Scanner(System.in);

System.out.print("Enter the start of the interval: ");

int start = sc.nextInt();

System.out.print("Enter the end of the interval: ");

int end = sc.nextInt();

sc.close();

System.out.println("Multiples of 10 within the interval [" + start + ", " + end + "]:");

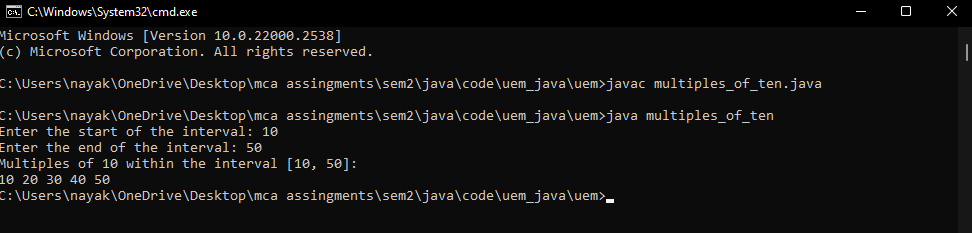
for (int i = start; i <= end; i++) {

if (i % 10 == 0) {

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

}}}}

**Output :**



**Question 9 : Write a Java program to generate multiplication table.**

**Source Code :**

import java.util.Scanner;

public class multiplication\_table {

public static void main(String[] args) {

Scanner sc = new Scanner(System.in);

System.out.print("Enter the number for the multiplication table: ");

int number = sc.nextInt();

System.out.print("Enter the range for the multiplication table: ");

int range = sc.nextInt();

sc.close();

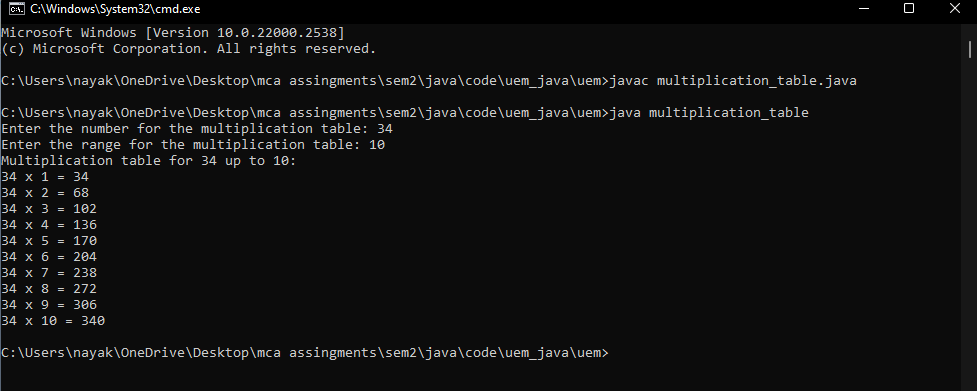
System.out.println("Multiplication table for " + number + " up to " + range + ":");

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

System.out.println(number + " x " + i + " = " + (number \* i));

}}}

**Output :**



**Question 10 : Write a Java program to find HCF of two Numbers**

**Source Code :**

import java.util.Scanner;

public class hcf {

public static void main(String[] args) {

Scanner sc = new Scanner(System.in);

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

int num1 = sc.nextInt();

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

int num2 = sc.nextInt();

sc.close();

int hcf = findHCF(num1, num2); // Find the HCF using the Euclidean algorithm

System.out.println("HCF of " + num1 + " and " + num2 + " is: " + hcf);}

public static int findHCF(int num1, int num2) { // Method to find HCF using the Euclidean algorithm

while (num2 != 0) {

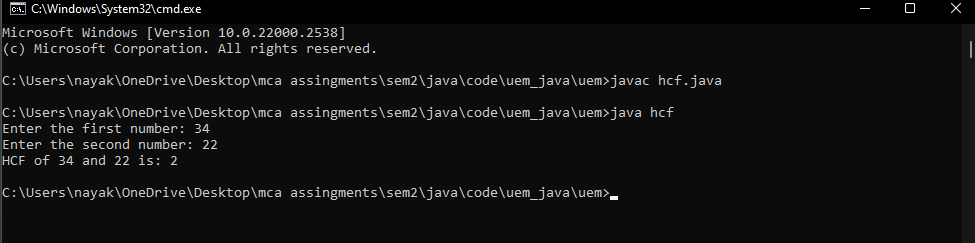
int temp = num2;

num2 = num1 % num2;

num1 = temp;}

return num1; }}

**Output :**



**Question 11 : Write a Java program to find LCM of two Numbers.**

**Source Code :**

import java.util.Scanner;

public class lcm {

public static void main(String[] args) {

Scanner sc = new Scanner(System.in);

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

int num1 = sc.nextInt();

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

int num2 = sc.nextInt();

sc.close();

int lcm = (num1 \* num2) / findHCF(num1, num2);

System.out.println("LCM of " + num1 + " and " + num2 + " is: " + lcm);}

public static int findHCF(int num1, int num2) { // Method to find HCF using the Euclidean algorithm

while (num2 != 0) {

int temp = num2;

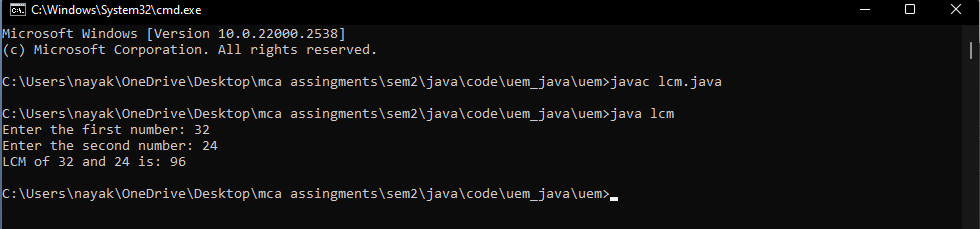
num2 = num1 % num2;

num1 = temp;}

return num1;

}}

**Output :**



**Question 12 : Write a Java program to count the number of digits of an integer**

**Source Code :**

import java.util.Scanner;

public class count\_digits {

public static void main(String[] args) {

Scanner sc = new Scanner(System.in);

System.out.print("Enter an integer: ");

int number = sc.nextInt();

sc.close();

int count = 0; // Count the number of digits

int temp = number;

while (temp != 0) {

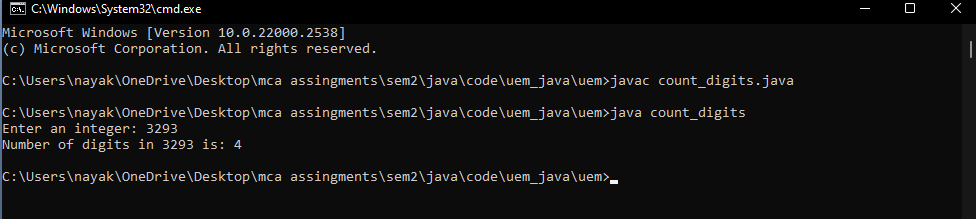
temp /= 10;

count++; }

System.out.println("Number of digits in " + number + " is: " + count);

}}

**Output :**



**Question 13 :** **Write a Java program to calculate the exponential of a number.**

**Source Code :**

import java.util.Scanner;

public class exponential {

public static void main(String[] args) {

Scanner sc = new Scanner(System.in);

System.out.print("Enter the base number: ");

double base = sc.nextDouble();

System.out.print("Enter the exponent: ");

int exponent = sc.nextInt();

sc.close();

double result=1 ;

if (exponent < 0) {

base = 1 / base;

exponent = -exponent; }

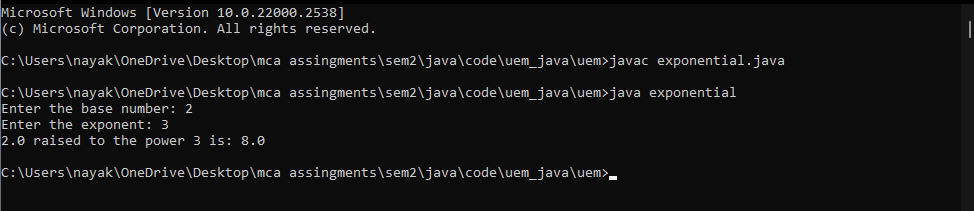
for (int i = 0; i < exponent; i++) {

result \*= base; }

System.out.println(base + " raised to the power " + exponent + " is: " + result);

}}

**Output :**



**Question 14 : Write a Java program to check whether a number is palindrome or not.**

**Source Code :**

import java.util.Scanner;

public class palindrome {

public static void main(String[] args) {

Scanner sc = new Scanner(System.in);

System.out.print("Enter a number: ");

int n = sc.nextInt();

sc.close();

// Check if the number is a palindrome

int ori = n;

int rev = 0;

while (n != 0) {

int digit = n % 10;

rev = rev \* 10 + digit;

n /= 10; }

if (ori == rev) {

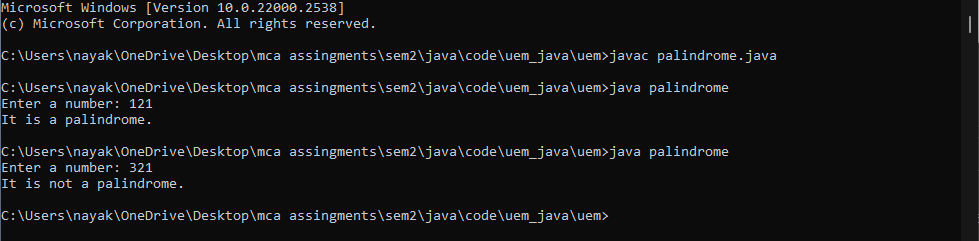
System.out.println("It is a palindrome.");

} else {

System.out.println("It is not a palindrome.");

}}}

**Output :**



**Question 15: . Write a Java program to check whether a number is prime or not.**

**Source Code :**

import java.util.Scanner;

public class prime {

public static void main(String[] args) {

Scanner sc = new Scanner(System.in);

System.out.print("Enter a number: ");

int num = sc.nextInt();

sc.close(); // Close the scanner

boolean isPrime = checkPrime(num);// Check if the number is prime

if (isPrime) {

System.out.println(num + " is a prime number.");

} else {

System.out.println(num + " is not a prime number.");

}}

public static boolean checkPrime(int num) { // Method to check if a number is prime

if (num <= 1) { // 0 and 1 are not prime numbers

return false;}

for (int i = 2; i <= Math.sqrt(num); i++) {

if (num % i == 0) {

return false;

}} return true; }}

**Output :**



**Question 16 : Write a Java program to convert a Binary Number to Decimal and Decimal to Binary.**

**Source Code :**

import java.util.Scanner;

public class binary\_decimal {

public static void main(String[] args) {

Scanner sc = new Scanner(System.in);

System.out.println("Choose conversion:");

System.out.println("1. Binary to Decimal");

System.out.println("2. Decimal to Binary");

System.out.print("Enter your choice (1 or 2): ");

int choice = sc.nextInt();

switch (choice) { // Perform the conversion based on the choice

case 1:

binaryToDecimal();

break;

case 2:

decimalToBinary();

break;

default:

System.out.println("Invalid choice. Please enter 1 or 2."); }

sc.close(); }

public static void binaryToDecimal() { // Method to convert binary to decimal

Scanner sc = new Scanner(System.in);

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

String binaryString = sc.nextLine();

sc.close();

int decimal = Integer.parseInt(binaryString, 2); // Convert binary string to decimal

System.out.println("Decimal equivalent: " + decimal); // Print the decimal equivalent

}

public static void decimalToBinary() { // Method to convert decimal to binary

Scanner sc = new Scanner(System.in);

System.out.print("Enter a decimal number: ");

int decimal = sc.nextInt();

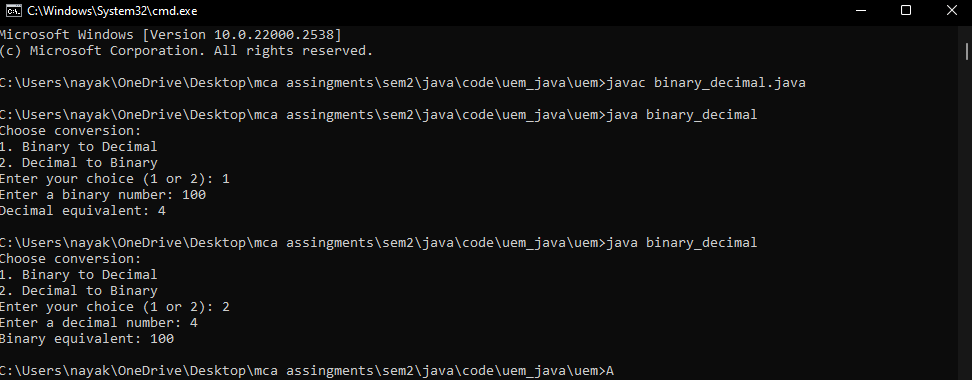
sc.close();

String binaryString = Integer.toBinaryString(decimal);// Convert decimal to binary string

System.out.println("Binary equivalent: " + binaryString);// Print the binary equivalent

}}

**Output :**



**Question 17 : Write a Java program to find median of a set of numbers.**

**Source Code :**

import java.util.Scanner;

public class median {

public static void main(String[] args) {

Scanner scanner = new Scanner(System.in);

System.out.print("Enter the number of elements: ");

int n = scanner.nextInt();

int[] numbers = new int[n]; // Create an array to store the elements

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

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

numbers[i] = scanner.nextInt();

}

sort(numbers); // Sort the array in ascending order

double median;

if (n % 2 == 0) {

median = (double) (numbers[n / 2 - 1] + numbers[n / 2]) / 2;

} else {

median = numbers[n / 2];}

System.out.println("Median: " + median);

scanner.close();// Close the scanner

}

public static void sort(int[] arr) { // Method to perform selection sort

int n = arr.length;

for (int i = 0; i < n - 1; i++) {

int minIndex = i;

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

if (arr[j] < arr[minIndex]) {

minIndex = j;

}}

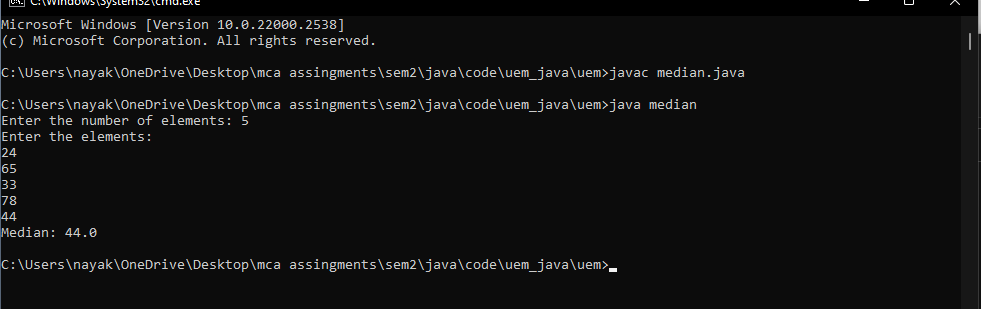
int temp = arr[minIndex];

arr[minIndex] = arr[i];

arr[i] = temp;

}}}

**Output :**



**Question 18 : Write a program to compute the value of Euler’s number that is used as the base of natural logarithms. Use**

**the following formula. e= 1+ 1/1! +1 /2! + 1/3+................ 1/n!**

**Source Code :**

public class euler {

public static void main(String[] args) {

int n = 10; // You can change the value of n to adjust the accuracy

double e = 1.0;

double factorial = 1.0;

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

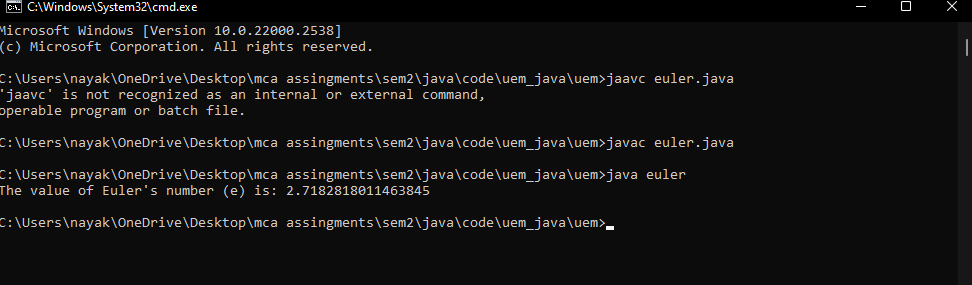
factorial \*= i;

e += 1.0 / factorial;}

System.out.println("The value of Euler's number (e) is: " + e);

}}

**Output :**



**Question 19 : Write a Java program to generate all combination of 1, 2, or 3 using loop.**

**Source Code :**

public class combination {

public static void main(String[] args) {

int[] nums = {1, 2, 3};

System.out.println("All combinations:");

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

System.out.println(nums[i]);

for (int j = i + 1; j < nums.length; j++) {

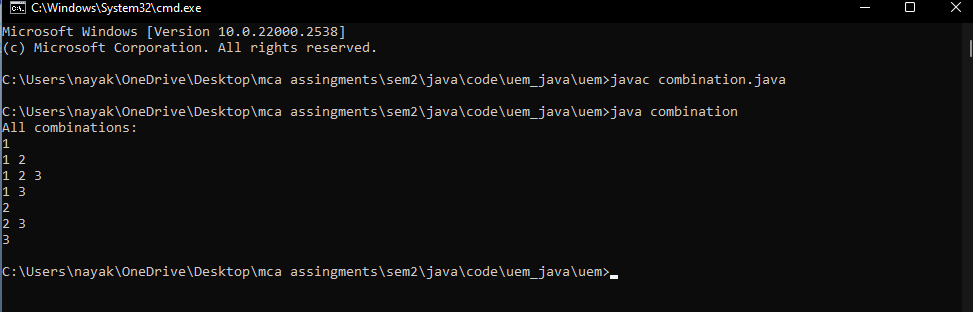
System.out.println(nums[i] + " " + nums[j]);

for (int k = j + 1; k < nums.length; k++) {

System.out.println(nums[i] + " " + nums[j] + " " + nums[k]);

}}}}}

**Output :**



**Question 20 : Write a Java program to read two integer values m and n and to and print whether m is multiple of n.**

**Source Code :**

import java.util.Scanner;

public class multiple {

public static void main(String[] args) {

Scanner sc = new Scanner(System.in);

System.out.print("Enter the value of m: ");

int m = sc.nextInt();

System.out.print("Enter the value of n: ");

int n = sc.nextInt();

if (m % n == 0) { // Check if m is a multiple of n

System.out.println(m + " is a multiple of " + n);

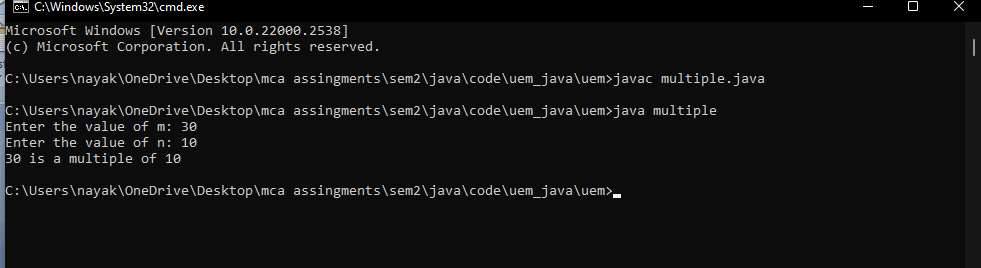
} else {

System.out.println(m + " is not a multiple of " + n);}

sc.close();

}}

**Output :**



**Question 21 :** . **Write a Java program to display prime numbers between a given interval.**

**Source Code :**

import java.util.Scanner;

public class prime\_range {

public static void main(String[] args) {

Scanner sc = new Scanner(System.in);

System.out.print("Enter the starting number of the interval: ");

int start = sc.nextInt();

System.out.print("Enter the ending number of the interval: ");

int end = sc.nextInt();

System.out.println("Prime numbers between " + start + " and " + end + ":");

for (int i = start; i <= end; i++) { // Iterate through the interval and check for prime numbers

if (isPrime(i)) {

System.out.println(i); }}

sc.close();}

public static boolean isPrime(int num) { // Method to check if a number is prime

if (num <= 1) {

return false;

}

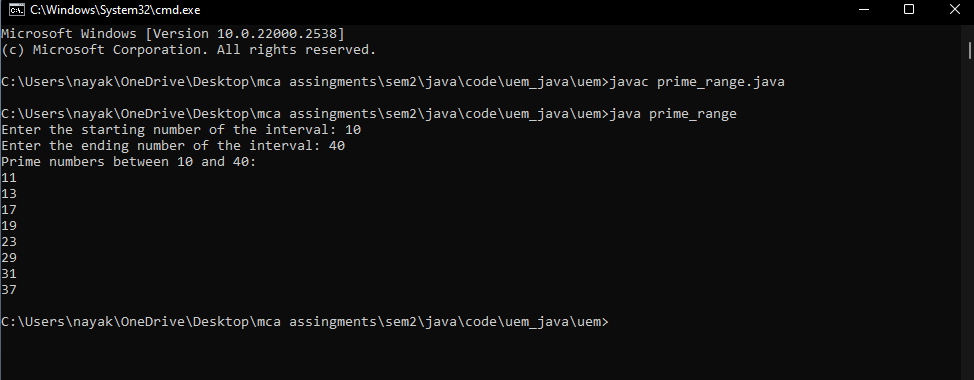
for (int i = 2; i <= Math.sqrt(num); i++) {

if (num % i == 0) {

return false; }

}return true; }}

**Output :**



**Question 22 : . Write a Java program to check whether a given number is Armstrong Number or not.**

**Source Code :**

import java.util.Scanner;

public class armstrong {

public static void main(String[] args) {

Scanner sc = new Scanner(System.in);

System.out.print("Enter a number: ");

int num = sc.nextInt();

int ori, rem, result = 0, n = 0;

ori = num;

for (ori = num; ori != 0; ori /= 10) {

++n; }

ori = num;

while (ori != 0) { // Calculate result

rem = ori % 10;

result += Math.pow(rem, n);

ori /= 10; }

if (result == num) { // Check if the number is Armstrong

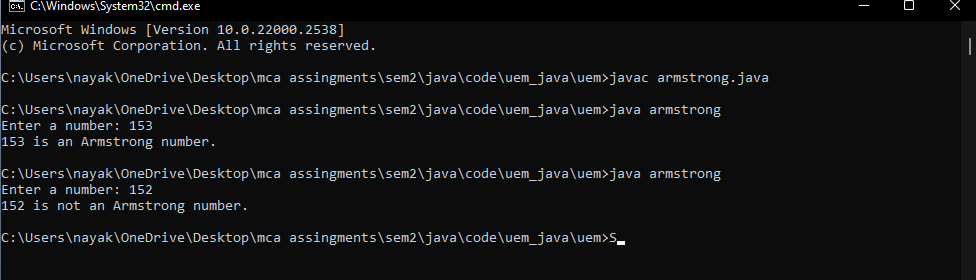
System.out.println(num + " is an Armstrong number.");

} else {

System.out.println(num + " is not an Armstrong number.");

}sc.close();}}

**Output :**



**WEEK 3**

**Question 1 : Write a Java program to calculate Sum & Average of an integer array.**

**Source Code :**

import java.util.Scanner;

public class sum\_average\_array {

public static void main(String[] args) {

Scanner sc = new Scanner(System.in);

System.out.print("Enter the number of elements in the array: ");

int size = sc.nextInt();

int[] array = new int[size]; // Declare an array of the given size

System.out.println("Enter the elements of the array:");

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

System.out.print("Element " + (i + 1) + ": ");

array[i] = sc.nextInt(); }

int sum = 0; // Calculate sum

for (int num : array) {

sum += num; }

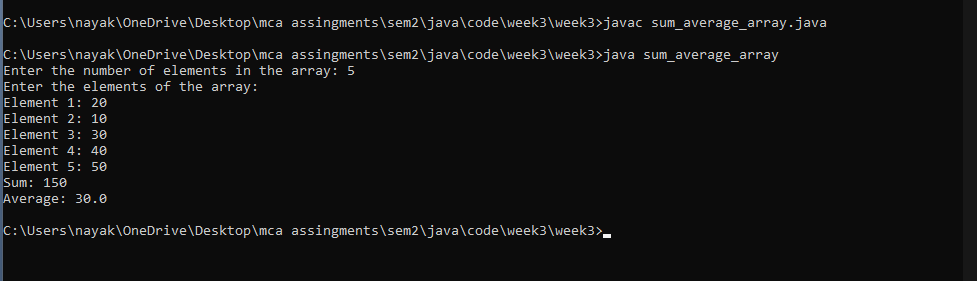
double average = (double) sum / size; // Calculate average

System.out.println("Sum: " + sum); // Display sum and average

System.out.println("Average: " + average);

sc.close(); }}

**Output :**



**Question 2 : Write a Java program to implement stack using array.**

**Source Code :**

import java.util.Scanner;

public class stack {

private static final int MAX\_SIZE = 100;

private static int[] stack = new int[MAX\_SIZE];

private static int top = -1;

public static void push(int value) {

if (top == MAX\_SIZE - 1) {

System.out.println("Stack Overflow. Cannot push element.");

} else {

stack[++top] = value;

System.out.println(value + " pushed into the stack.");

}}

public static void pop() {

if (top == -1) {

System.out.println("Stack Underflow. Cannot pop element.");

} else {

System.out.println("Popped element: " + stack[top--]); }}

public static int topElement() {

if (top == -1) {

System.out.println("Stack is empty.");

return -1;

} else {

return stack[top]; }}

public static void display() {

if (top == -1) {

System.out.println("Stack is empty.");

} else {

System.out.print("Stack elements: ");

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

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

System.out.println(); }}

public static void main(String[] args) {

Scanner sc = new Scanner(System.in);

int choice, element;

do {

System.out.println("\nStack Operations Menu:");

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

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

System.out.println("3. Get Top Element");

System.out.println("4. Display Stack");

System.out.println("5. Exit");

System.out.print("Enter your choice: ");

choice = sc.nextInt();

switch (choice) {

case 1:

System.out.print("Enter element to push: ");

element = sc.nextInt();

push(element);

break;

case 2:

pop();

break;

case 3:

element = topElement();

if (element != -1) {

System.out.println("Top element: " + element);

}

break;

case 4:

display();

break;

case 5:

System.out.println("Exiting program. Goodbye!");

break;

default:

System.out.println("Invalid choice. Please enter a valid option.");

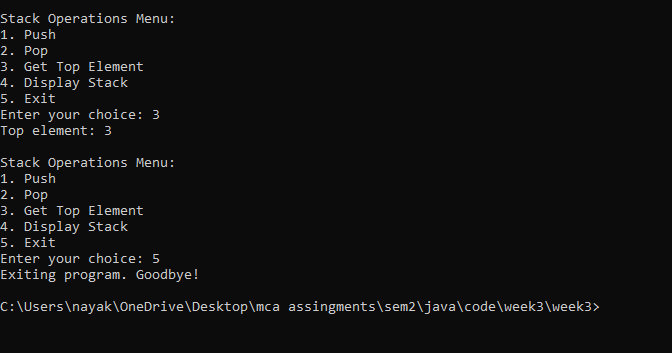
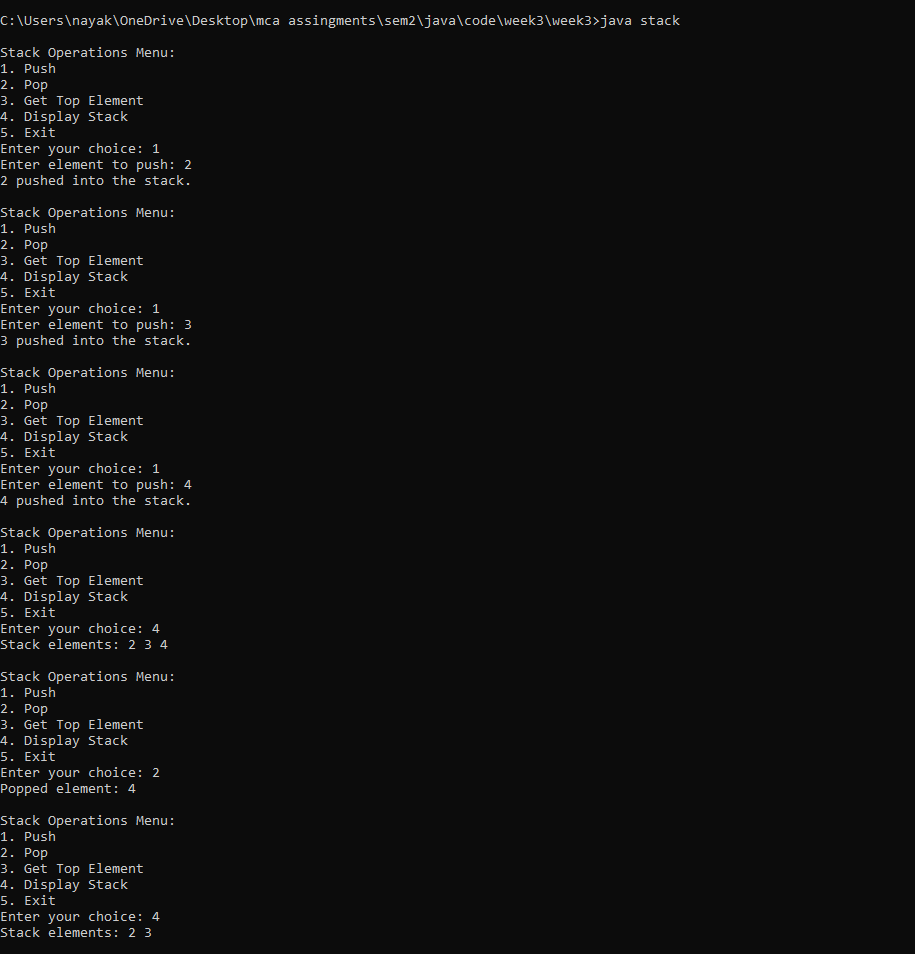
}

} while (choice != 5);

sc.close();

}}

**Output :**



**Question 3 : Write a Java program to implement Queue using array**

**Source Code :**

import java.util.Scanner;

public class queue {

private static final int MAX = 10;

private static int[] queue = new int[MAX];

private static int front = -1, rear = -1;

public static void enqueue(int element) {

if (rear == MAX - 1) {

System.out.println("Queue is full. Cannot enqueue.");

} else {

if (front == -1) {

front = 0;

}

rear++;

queue[rear] = element;

System.out.println(element + " has been enqueued."); }}

public static void dequeue() {

if (front == -1) {

System.out.println("Queue is empty. Cannot dequeue.");

} else {

System.out.println(queue[front] + " has been dequeued.");

if (front == rear) {

front = rear = -1;

} else {

front++;

}}}

public static int peek() {

if (front == -1) {

System.out.println("Queue is empty. No front element to peek.");

return -1; // Return an error value

} else {

return queue[front];

}}

public static void display() {

if (front == -1) {

System.out.println("Queue is empty.");

} else {

System.out.print("Queue elements: ");

for (int i = front; i <= rear; i++) {

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

System.out.println(); }}

public static void main(String[] args) {

Scanner sc = new Scanner(System.in);

int choice, element;

System.out.println("\nMenu:");

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

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

System.out.println("3. Peek");

System.out.println("4. Display");

System.out.println("5. Exit");

while (true) {

System.out.print("Enter your choice: ");

choice = sc.nextInt();

switch (choice) {

case 1:

System.out.print("Enter element to enqueue: ");

element = sc.nextInt();

enqueue(element);

break;

case 2:

dequeue();

break;

case 3:

element = peek();

if (element != -1) {

System.out.println("Front element: " + element);

}

break;

case 4:

display();

break;

case 5:

sc.close();

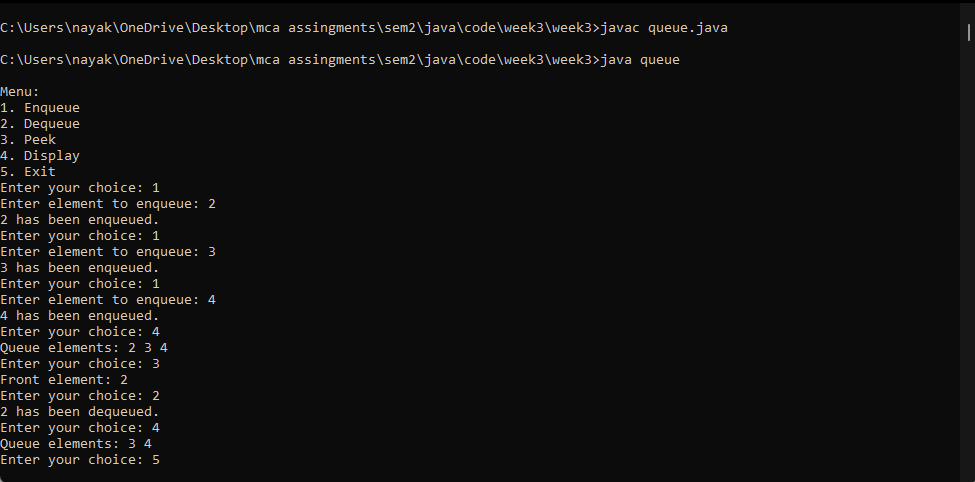
System.exit(0);

default:

System.out.println("Invalid choice. Please try again.");

}}}}

**Output :**



**Question 4 : Write a Java program to calculate Sum of two 2-dimensional arrays.**

**Source Code :**

import java.util.Scanner;

public class sum\_2D\_array {

public static void main(String[] args) {

Scanner scanner = new Scanner(System.in);

System.out.print("Enter the number of rows for the arrays: ");

int rows = scanner.nextInt();

System.out.print("Enter the number of columns for the arrays: ");

int cols = scanner.nextInt();

// Initialize arrays

int[][] array1 = new int[rows][cols];

int[][] array2 = new int[rows][cols];

int[][] sum= new int[rows][cols];

System.out.println("Enter elements for the first array:");

for (int i = 0; i < array1.length; i++) {

for (int j = 0; j < array1[0].length; j++) {

System.out.print("Enter element at position [" + i + "][" + j + "]: ");

array1[i][j] = scanner.nextInt();

}}

System.out.println("Enter elements for the second array:");

for (int i = 0; i < array2.length; i++) {

for (int j = 0; j < array2[0].length; j++) {

System.out.print("Enter element at position [" + i + "][" + j + "]: ");

array2[i][j] = scanner.nextInt();

}}

for (int i = 0; i < rows; i++) { // Calculate sum

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

sum[i][j] = array1[i][j] + array2[i][j];

}}

System.out.println("Sum of the two arrays:");

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

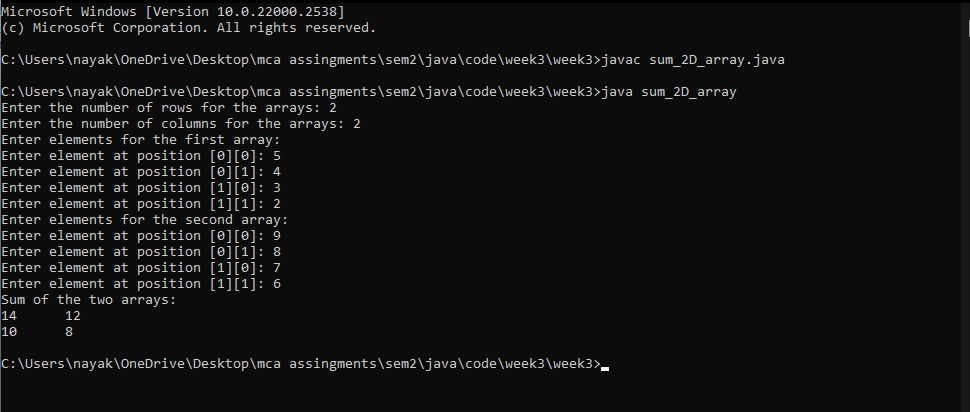
for (int j = 0; j < sum[0].length; j++) {

System.out.print(sum[i][j] + "\t");

}System.out.println();}

scanner.close(); }}

**Output :**



**Question 5 : Write a Java program to find the range of a 1D array.**

**Source Code :**

import java.util.Scanner;

public class range\_array {

public static void main(String[] args) {

Scanner sc = new Scanner(System.in);

System.out.print("Enter the size of the array: "); // Input the size of the array

int size = sc.nextInt();

int[] array = new int[size];

System.out.println("Enter the elements of the array:");

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

System.out.print("Element " + (i + 1) + ": ");

array[i] = sc.nextInt();}

int min = array[0];

int max = array[0];

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

if (array[i] < min) {

min = array[i]; }

if (array[i] > max) {

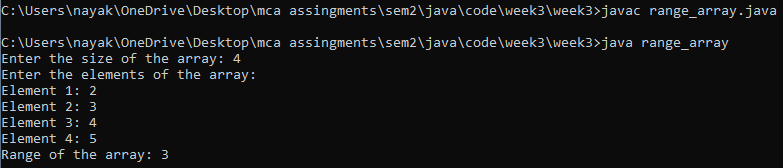
max = array[i]; }}

int range = max - min;

System.out.println("Range of the array: " + range);

sc.close();

}}  
**Output :**

****

**Question 6 : Write a Java program to search an element in an array.**

**Source Code :**

import java.util.Scanner;

public class linear\_search {

public static void main(String[] args) {

Scanner sc = new Scanner(System.in);

System.out.print("Enter the size of the array: ");

int size = sc.nextInt();

int[] array = new int[size]; // Declare the array

System.out.println("Enter the elements of the array:");

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

System.out.print("Element " + (i + 1) + ": ");

array[i] = sc.nextInt(); }

System.out.print("Enter the element to search: "); // Input the element to be searched

int target = sc.nextInt();

int index = linearSearch(array, target); // Perform linear search

if (index != -1) {

System.out.println("Element " + target + " found at index " + index);

} else {

System.out.println("Element " + target + " not found in the array");

}sc.close();}

public static int linearSearch(int[] array, int target) { // Method to perform search

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

if (array[i] == target) {

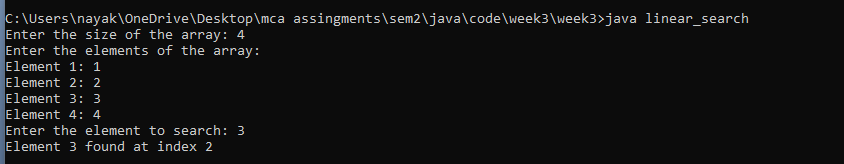
return i; // Return the index where the element is found

}}

return -1; // Return -1 if element is not found

}}

**Output :**



**Question 7 : Write a Java program to find the sum of even numbers in an integer array.**

**Source Code :**

import java.util.Scanner;

public class sum\_even\_numbers\_array {

public static void main(String[] args) {

Scanner sc= new Scanner(System.in);

System.out.print("Enter the size of the array: ");

int size = sc.nextInt();

int[] array = new int[size];

// Input elements into the array

System.out.println("Enter the elements of the array:");

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

System.out.print("Element " + (i + 1) + ": ");

array[i] = sc.nextInt(); }

int sum = 0;

for (int num : array) {

if (num % 2 == 0) {

sum += num;

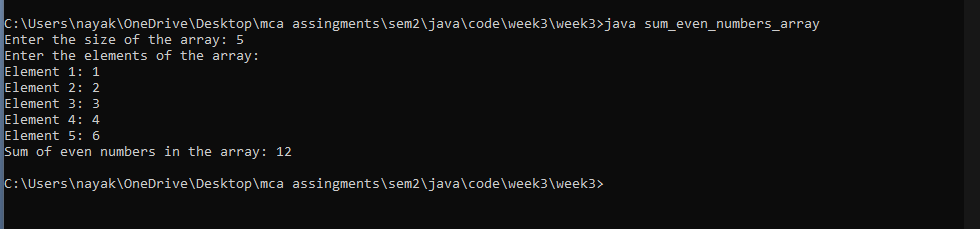
}}

System.out.println("Sum of even numbers in the array: " + sum);

sc.close();

}}

**Output :**



**Question 8 : Write a Java program to find the sum of diagonal elements in a 2D array**

**Source Code :**

import java.util.Scanner;

public class sum\_diagonals {

public static void main(String[] args) {

Scanner sc = new Scanner(System.in);

System.out.print("Enter the size of the square matrix: ");

int size = sc.nextInt();

int[][] matrix = new int[size][size];

// Input elements into the matrix

System.out.println("Enter the elements of the square matrix:");

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

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

System.out.print("Element at position [" + i + "][" + j + "]: ");

matrix[i][j] = sc.nextInt(); }}

int sum = 0;

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

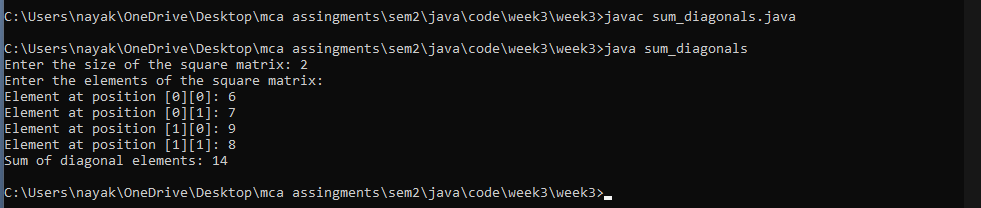
sum += matrix[i][i]; // Add elements from the main diagonal}

System.out.println("Sum of diagonal elements: " + sum);

sc.close();

}}

**Output :**



**Question 9 : Reverse the elements in an array of integers without using a second array.**

**Source Code :**

import java.util.Scanner;

public class reverse\_array {

public static void main(String[] args) {

Scanner sc = new Scanner(System.in);

System.out.print("Enter the size of the array: ");

int size = sc.nextInt();

int[] array = new int[size];

// Input elements into the array

System.out.println("Enter the elements of the array:");

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

System.out.print("Element " + (i + 1) + ": ");

array[i] = sc.nextInt();

}

reverseArray(array); // Reverse the array

System.out.println("Reversed array:"); // Display the reversed array

for (int num : array) {

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

}sc.close(); }

public static void reverseArray(int[] array) {

int start = 0;

int end = array.length - 1;

while (start < end) {

int temp = array[start];

array[start] = array[end];

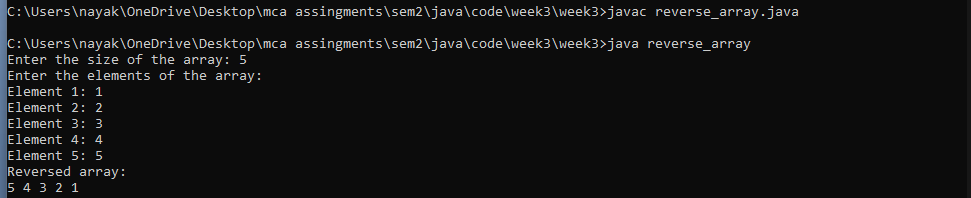
array[end] = temp;

start++;

end--;

}}}

**Output :**



**Question 10 : Write a Java program to enter n elements in an array and find smallest number among them**

**Source Code :**

import java.util.Scanner;

public class min\_array {

public static void main(String[] args) {

Scanner sc = new Scanner(System.in);

System.out.print("Enter the number of elements in the array: ");

int n = sc.nextInt();

int[] array = new int[n];

System.out.println("Enter the elements of the array:");

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

System.out.print("Element " + (i + 1) + ": ");

array[i] = sc.nextInt(); }

int min = array[0];

for (int i = 1; i < n; i++) { // Find the smallest number in the array

if (array[i] < min) {

min = array[i];

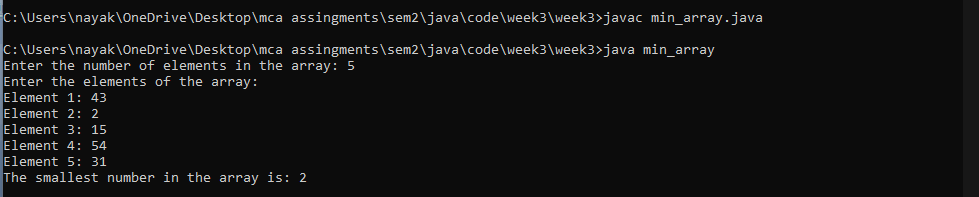
}}

System.out.println("The smallest number in the array is: " + min);

sc.close();

}}

**Output :**



**Question 11 : . Write Java program to find the sum of all odd numbers in a 2D array.**

**Source Code :**

import java.util.Scanner;

public class sum\_odd\_numbers\_matrix {

public static void main(String[] args) {

Scanner sc = new Scanner(System.in);

System.out.print("Enter the number of rows: ");

int rows = sc.nextInt();

System.out.print("Enter the number of columns: ");

int cols = sc.nextInt();

int[][] array = new int[rows][cols];

System.out.println("Enter the elements of the array:");

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

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

array[i][j] = sc.nextInt(); }}

int sum = 0;

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

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

if (array[i][j] % 2 != 0) {

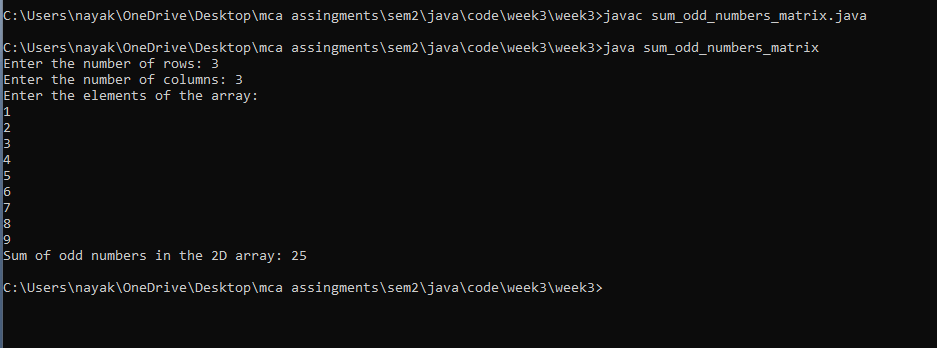
sum += array[i][j];

}} }

System.out.println("Sum of odd numbers in the 2D array: " + sum);

sc.close();}}

**Output :**



**Question 12 : . Write a Java program to print transpose of matrix.**

**Source Code :**

import java.util.Scanner;

public class transpose\_matrix {

public static void main(String[] args) {

Scanner sc = new Scanner(System.in);

System.out.print("Enter the number of rows: ");

int rows = sc.nextInt();

System.out.print("Enter the number of columns: ");

int cols = sc.nextInt();

int[][] matrix = new int[rows][cols];

System.out.println("Enter the elements of the matrix:");

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

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

matrix[i][j] = sc.nextInt(); }}

int[][] transpose = new int[cols][rows];

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

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

transpose[i][j] = matrix[j][i]; }}

System.out.println("Transpose of the matrix:");

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

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

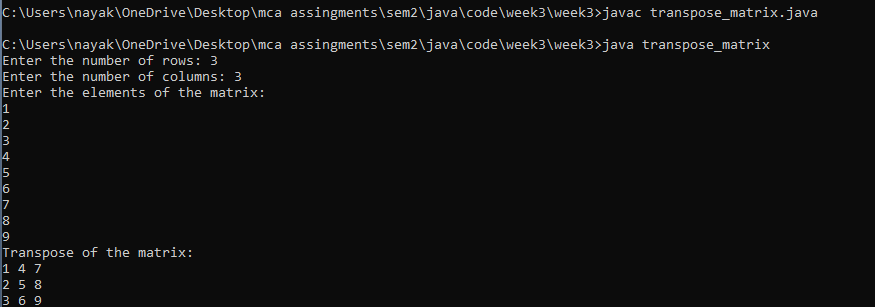
System.out.print(transpose[i][j] + " "); }

System.out.println();

}sc.close();

} }

**Output :**



**Question 13 : . Write a Java program to check whether a given matrix is sparse or not.**

**Source Code :**

import java.util.Scanner;

public class sparse\_matrix{

public static void main(String[] args) {

Scanner sc = new Scanner(System.in);

System.out.print("Enter the number of rows: ");

int rows = sc.nextInt();

System.out.print("Enter the number of columns: ");

int cols = sc.nextInt();

int[][] matrix = new int[rows][cols];

System.out.println("Enter the elements of the matrix:");

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

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

matrix[i][j] = sc.nextInt();}}

int zero = 0;

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

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

if (matrix[i][j] == 0) {

zero++; }}}

double ratio = (double) zero / (rows \* cols);

if (ratio > 0.5) { // If more than 50% of elements are zero

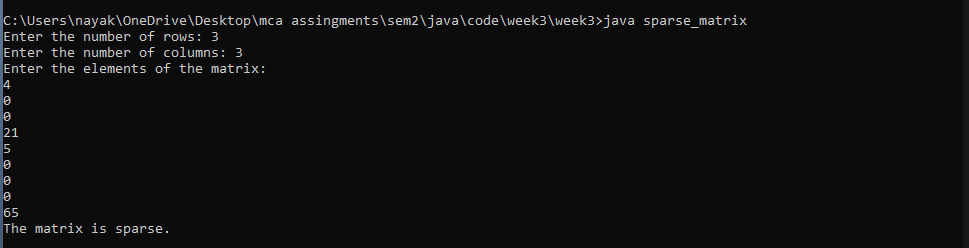
System.out.println("The matrix is sparse.");

} else {

System.out.println("The matrix is not sparse.");

}sc.close();}}

**Output :**

****

**Question 14 : . Write a Java program to count the prime numbers in an array.**

**Source Code :**

import java.util.Scanner;

public class count\_prime\_array {

static boolean isPrime(int num) { // Function to check if a number is prime

if (num <= 1) {

return false;

}

for (int i = 2; i \* i <= num; i++) {

if (num % i == 0) {

return false; }}

return true;}

public static void main(String[] args) {

Scanner sc = new Scanner(System.in);

System.out.print("Enter the size of the array: ");

int size = sc.nextInt();

int[] arr = new int[size]; // Input the elements of the array

System.out.println("Enter the elements of the array:");

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

arr[i] = sc.nextInt();

}

int c = 0; // Count prime numbers in the array

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

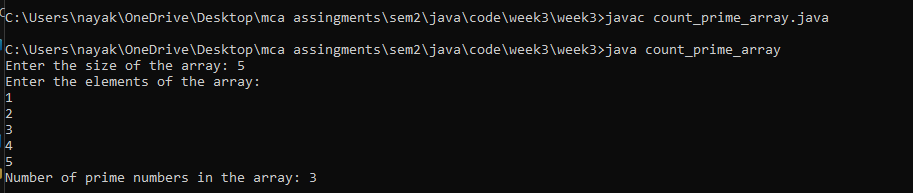
if (isPrime(arr[i])) {

c++;}}

System.out.println("Number of prime numbers in the array: " +c);

sc.close();}}

**Output :**



**Question 15 : . Write a Java program to find second highest element of an array**

**Source Code :**

import java.util.\*;

class second\_max {

static void sec\_max(Integer arr[], int arr\_size){

Arrays.sort(arr, Collections.reverseOrder()); // Sort the array in descending order

for (int i = 1; i < arr\_size; i++) {

if (arr[i] != arr[0]) {

System.out.printf("The second largest "+ "element is %d\n",arr[i]);

return;}}

System.out.printf("There is no second "+ "largest element\n");}

public static void main(String[] args){

Scanner sc=new Scanner(System.in);

System.out.print("Enter the size of the array: "); // Input the size of the array

int n = sc.nextInt();

Integer[] arr = new Integer[n];

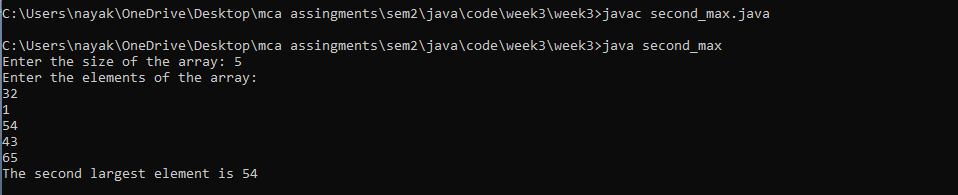
System.out.println("Enter the elements of the array:");

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

arr[i] = sc.nextInt();

}sec\_max(arr, n);}}

**Output :**



**Question 16 :** . **Write a Java program which counts the non-zero elements in an integer array**

**Source Code :**

import java.util.Scanner;

public class count\_non\_zero\_elements\_array {

public static void main(String[] args) {

Scanner sc = new Scanner(System.in);

System.out.print("Enter the size of the array: ");

int size = sc.nextInt();

int[] arr = new int[size]; // Input the elements of the array

System.out.println("Enter the elements of the array:");

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

arr[i] = sc.nextInt();}

int c = 0;

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

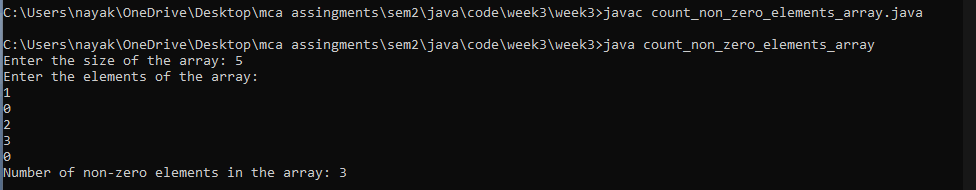
if (arr[i] != 0) {

c++; }}

System.out.println("Number of non-zero elements in the array: " + c);

sc.close();}}

**Output :**



**Question 17 : . Write a Java program to merge two float arrays.**

**Source Code :**

import java.util.Scanner;

public class merge\_array {

public static void main(String[] args) {

Scanner sc = new Scanner(System.in);

System.out.print("Enter the size of the first array: ");

int size1 = sc.nextInt();

float[] arr1 = new float[size1];

System.out.println("Enter the elements of the first array:");

for (int i = 0; i < size1; i++) {

arr1[i] = sc.nextFloat(); }

System.out.print("Enter the size of the second array: ");

int size2 = sc.nextInt();

float[] arr2 = new float[size2];

System.out.println("Enter the elements of the second array:");

for (int i = 0; i < size2; i++) {

arr2[i] = sc.nextFloat(); }

float[] mergedArray = mergeArrays(arr1, arr2);

System.out.println("Merged array:");

for (float num : mergedArray) {

System.out.print(num + " ");}

sc.close(); }

public static float[] mergeArrays(float[] arr1, float[] arr2) {

int size1 = arr1.length;

int size2 = arr2.length;

float[] mergedArray = new float[size1 + size2];

for (int i = 0; i < size1; i++) {

mergedArray[i] = arr1[i];

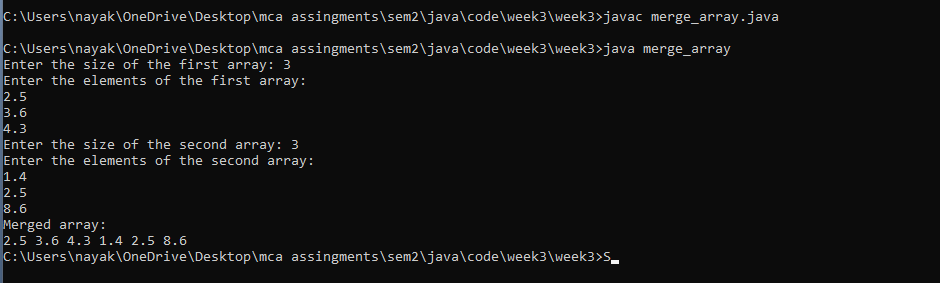
}

for (int i = 0; i < size2; i++) {

mergedArray[size1 + i] = arr2[i];

}return mergedArray; }}

**Output :**



**Question 18 : . Write a Java program where elements of two integer arrays get added index wise and get stored into a**

**third array.**

**Source Code :**

import java.util.Scanner;

public class add\_array {

public static void main(String[] args) {

Scanner sc = new Scanner(System.in);

System.out.print("Enter the size of the arrays: ");

int size = sc.nextInt();

int[] arr1 = new int[size]; // Input the elements of the first array

System.out.println("Enter the elements of the first array:");

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

arr1[i] = sc.nextInt(); }

int[] arr2 = new int[size]; // Input the elements of the second array

System.out.println("Enter the elements of the second array:");

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

arr2[i] = sc.nextInt(); }

int[] resultArray = new int[size]; // Add the elements index-wise and store into a third array

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

resultArray[i] = arr1[i] + arr2[i];}

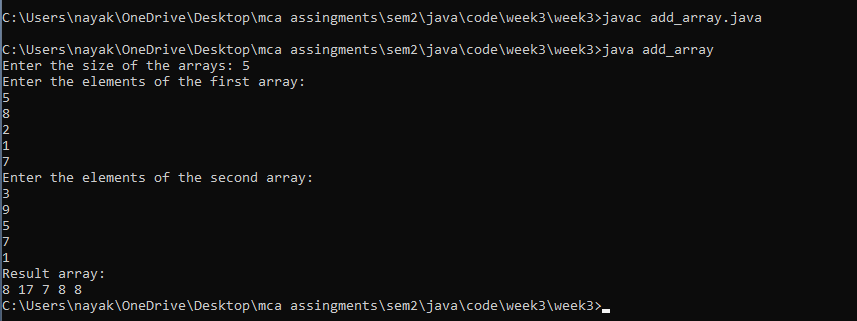
System.out.println("Result array:");

for (int num : resultArray) {

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

}sc.close();}}

**Output :**



**Question 19 : . Write a Java program to multiply two matrices.**

**Source Code :**

import java.util.Scanner;

public class multiply\_matrix {

public static void main(String[] args) {

Scanner sc= new Scanner(System.in);

System.out.print("Enter the number of rows of the first matrix: ");

int rows1 = sc.nextInt();

System.out.print("Enter the number of columns of the first matrix: ");

int cols1 = sc.nextInt();

System.out.print("Enter the number of rows of the second matrix: ");

int rows2 = sc.nextInt();

System.out.print("Enter the number of columns of the second matrix: ");

int cols2 = sc.nextInt();

if (cols1 != rows2) { // Check if matrix multiplication is possible

System.out.println("Matrix multiplication is not possible.");

return;}

int[][] matrix1 = new int[rows1][cols1]; // Input elements of the first matrix

System.out.println("Enter the elements of the first matrix:");

for (int i = 0; i < rows1; i++) {

for (int j = 0; j < cols1; j++) {

matrix1[i][j] = sc.nextInt();

}}

int[][] matrix2 = new int[rows2][cols2]; // Input elements of the second matrix

System.out.println("Enter the elements of the second matrix:");

for (int i = 0; i < rows2; i++) {

for (int j = 0; j < cols2; j++) {

matrix2[i][j] = sc.nextInt();

}}

int[][] result = multiply(matrix1, matrix2);

System.out.println("Resultant matrix after multiplication:");

for (int i = 0; i < rows1; i++) {

for (int j = 0; j < cols2; j++) {

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

}System.out.println();

}sc.close();}

public static int[][] multiply(int[][] matrix1, int[][] matrix2) {

int rows1 = matrix1.length;

int cols1 = matrix1[0].length;

int rows2 = matrix2.length;

int cols2 = matrix2[0].length;

int[][] result = new int[rows1][cols2];

for (int i = 0; i < rows1; i++) {

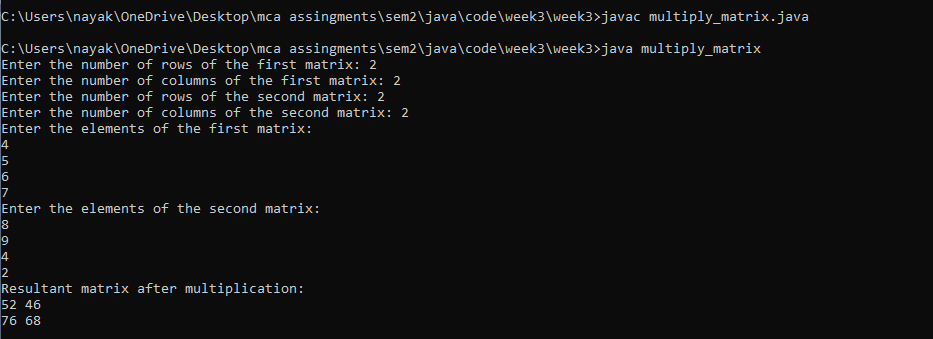
for (int j = 0; j < cols2; j++) {

for (int k = 0; k < cols1; k++) {

result[i][j] += matrix1[i][k] \* matrix2[k][j];

}}}return result;}}

**Output :**



**Question 20 : . Write a Java program to subtract two matrices**

**Source Code :**

import java.util.Scanner;

public class subtract\_matrix {

public static void main(String[] args) {

Scanner sc= new Scanner(System.in);

System.out.print("Enter the number of rows of the matrices: ");

int rows = sc.nextInt();

System.out.print("Enter the number of columns of the matrices: ");

int cols = sc.nextInt();

int[][] matrix1 = new int[rows][cols]; // Input elements of the first matrix

System.out.println("Enter the elements of the first matrix:");

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

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

matrix1[i][j] = sc.nextInt();}}

int[][] matrix2 = new int[rows][cols]; // Input elements of the second matrix

System.out.println("Enter the elements of the second matrix:");

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

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

matrix2[i][j] = sc.nextInt();}}

int[][] result = subtract(matrix1, matrix2); // Subtract matrices

System.out.println("Resultant matrix after subtraction:");

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

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

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

}System.out.println();

}sc.close();}

public static int[][] subtract(int[][] matrix1, int[][] matrix2) {

int rows = matrix1.length;

int cols = matrix1[0].length;

int[][] result = new int[rows][cols];

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

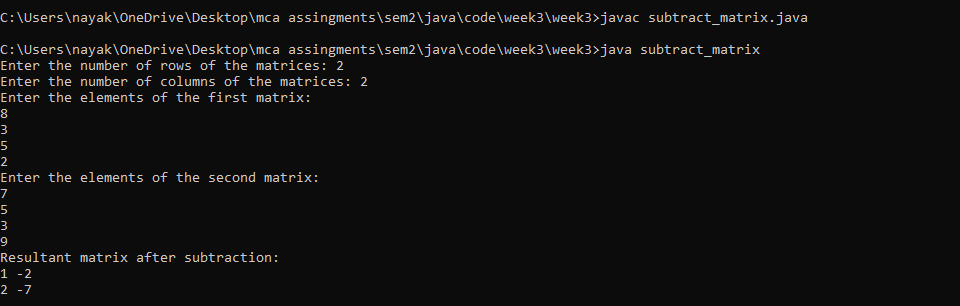
for (int j = 0; j < cols; j++) {

result[i][j] = matrix1[i][j] - matrix2[i][j];

}

}return result;}}

**Output :**



**Question 21 : Write a Java program to find duplicate elements in a 1D array and find their frequency of occurrence**

**Source Code :**

import java.util.\*;

public class duplicates\_frequency\_array{

public static void main(String[] args) {

Scanner sc=new Scanner(System.in);

System.out.print("Enter the size of the array: ");

int n = sc.nextInt();

int[] array = new int[n];// Input the elements of the array

System.out.println("Enter the elements of the array:");

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

array[i] = sc.nextInt();}

Arrays.sort(array);

int i,j,frequency; //declaring the variables

System.out.println("These elements are repeated along with its frequency-");

for(i=0; i<array.length; i++){ //loop for logic implementation

frequency = 1;

for(j=i+1; j<array.length; j++){

if(array[j] == array[i]){

frequency++;

} else {

break;}}

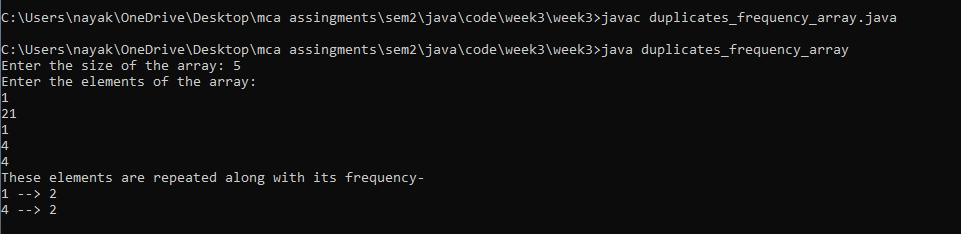
i=j-1;

if(frequency > 1){

System.out.println(array[i] + " --> " + frequency);

}}}}

**Output :**



**Question 22 : . Write a Java program to print every alternate number of a given array.**

**Source Code :**

import java.util.Scanner;

public class alternate\_number\_array {

public static void main(String[] args) {

Scanner sc = new Scanner(System.in);

System.out.print("Enter the size of the array: "); // Input the size of the array

int size = sc.nextInt();

int[] array = new int[size]; // Input the elements of the array

System.out.println("Enter the elements of the array:");

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

array[i] = sc.nextInt();}

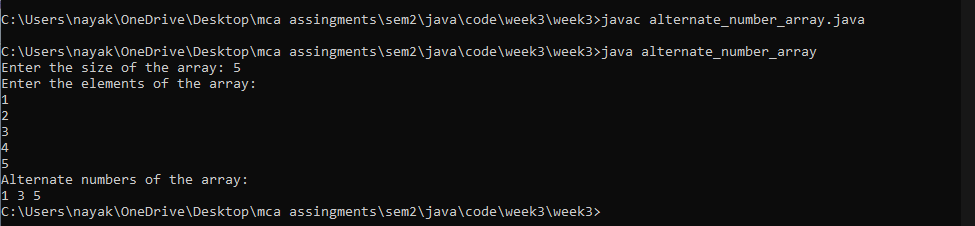
System.out.println("Alternate numbers of the array:");

for (int i = 0; i < size; i += 2) {

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

sc.close();}}

**Output :**



**Question 23 : . Given are two one-dimensional arrays A & B, which are sorted in ascending order. Write a Java program**

**to merge them into single sorted array C that contains every item from arrays A & B, in ascending order.**

**Source Code :**

import java.util.Scanner;

public class merge\_two\_array\_sort {

public static void main(String[] args) {

Scanner scanner = new Scanner(System.in);

System.out.print("Enter the size of the first array (A): ");

int sizeA = scanner.nextInt();

int[] A = new int[sizeA];

System.out.println("Enter the elements of the first array (A) in ascending order:");

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

A[i] = scanner.nextInt(); }

System.out.print("Enter the size of the second array (B): ");

int sizeB = scanner.nextInt();

int[] B = new int[sizeB];

System.out.println("Enter the elements of the second array (B) in ascending order:");

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

B[i] = scanner.nextInt(); }

int[] C = mergeArrays(A, B);

System.out.println("Merged array (C):"); // Display the merged array

for (int num : C) {

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

}scanner.close();}

public static int[] mergeArrays(int[] A, int[] B) {

int sizeA = A.length;

int sizeB = B.length;

int[] C = new int[sizeA + sizeB];

int i = 0, j = 0, k = 0;

while (i < sizeA && j < sizeB) {

if (A[i] < B[j]) {

C[k++] = A[i++];

} else {

C[k++] = B[j++];}}

while (i < sizeA) {

C[k++] = A[i++];}

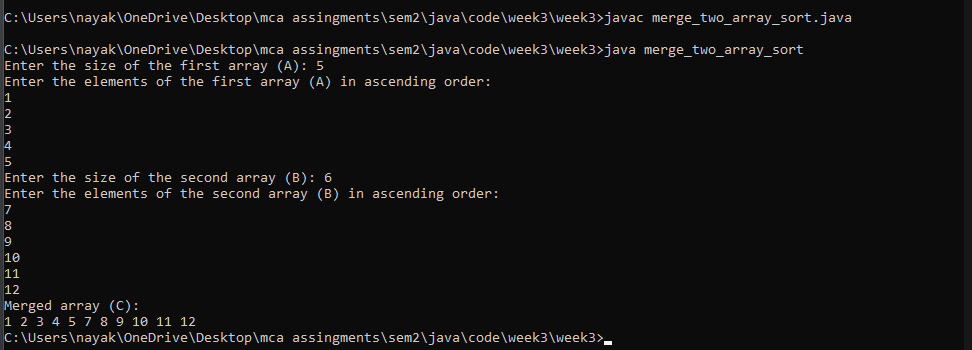
while (j < sizeB) {

C[k++] = B[j++];}

return C;

}}

**Output :**



**Question 24 : . Write a Java program to show 0-arguments constructor.**

**Source Code :**

class MyClass {

public MyClass() {

System.out.println("This is a 0-argument constructor.");}

public void display() {

System.out.println("hi, how are you?"); }}

public class zero\_argument {

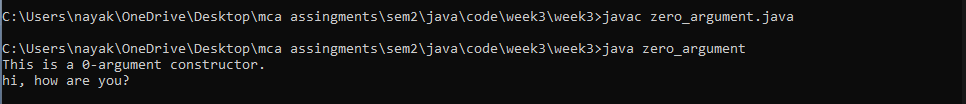
public static void main(String[] args) {

MyClass obj = new MyClass();// Creating an object of MyClass

obj.display(); // Calling a method of the object

}}

**Output :**



**Question 25 : . Write a Java program to show parameterized constructor.**

**Source Code :**

class Student {

private String name;

private int age;

public Student(String name, int age) { // Parameterized constructor

this.name = name;

this.age = age;}

public void displayDetails() { // Method to display student details

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

System.out.println("Age: " + age);

}}

public class parameterized\_constructor{

public static void main(String[] args) {

Student stu1 = new Student("Sahin", 22);

System.out.println("Student 1 Details:");

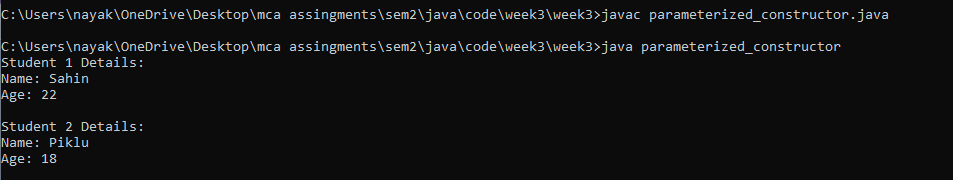
stu1.displayDetails();

Student stu2 = new Student("Piklu", 18);

System.out.println("\nStudent 2 Details:");

stu2.displayDetails();}}

**Output :**



**Question 26 : Write a Java program to show constructor overloading.**

**Source Code :**

class Box {

double width;

double height;

double depth;

Box() { // Constructor with no parameters

width = 1;

height = 1;

depth = 1;}

Box(double w, double h, double d) { // Constructor with three parameters

width = w;

height = h;

depth = d;}

Box(double len) { // Constructor with a single parameter to create a cube

width = len;

height = len;

depth = len;}

double volume() { // Method to calculate and return the volume of the box

return width \* height \* depth;}}

public class constructor\_overloading{

public static void main(String[] args) {

Box box1 = new Box(); // Default constructor

Box box2 = new Box(5, 3, 4); // Constructor with three parameters

Box box3 = new Box(2.5); // Constructor with a single parameter

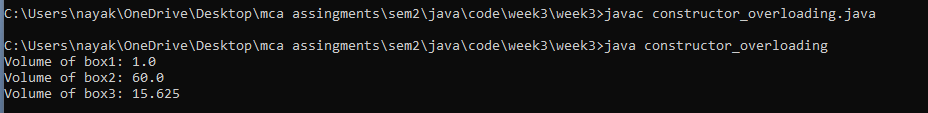
System.out.println("Volume of box1: " + box1.volume());

System.out.println("Volume of box2: " + box2.volume());

System.out.println("Volume of box3: " + box3.volume());

}}

**Output :**



**Question 27 : . Write a class, Grader, which has an instance variable, score, an appropriate constructor and appropriate**

**methods. A method, letterGrade() that returns the letter grade as O/E/A/B/C/F. Now write a demo class to**

**test the Grader class by reading a score from the user, using it to create a Grader object after validating**

**that the value is not negative and is not greater then 100. Finally, call the letterGrade() method to get and**

**print the grade.**

**Source Code :**

import java.util.Scanner;

class Grader {

private int score;

public Grader(int score) { // Constructor

this.score = score;}

public String letterGrade() {

if (score < 0 || score > 100) {

return "Invalid Score";

} else if (score >= 90) {

return "A";

} else if (score >= 80) {

return "B";

} else if (score >= 70) {

return "C";

} else if (score >= 60) {

return "D";

} else if (score >= 50) {

return "E";

} else {

return "F"; }}}

public class grader\_demo {

public static void main(String[] args) {

Scanner sc = new Scanner(System.in);

System.out.print("Enter the score: ");

int score = sc.nextInt();

if (score < 0 || score > 100) {

System.out.println("Invalid score. Score must be between 0 and 100.");

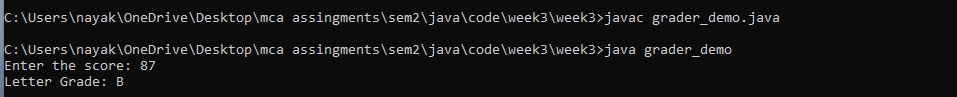
} else {

Grader ob = new Grader(score);

System.out.println("Letter Grade: " + ob.letterGrade());

}sc.close();}}

**Output :**



**Question 28 : Write a class, Commission, which has an instance variable,sales;an appropriate constructor; and a**

**method, commission()that returns the commission. Now write a demo class to test the Commission class by**

**reading a sale from the user, using it to create a Commission object after validating that the value is not**

**negative. Finally, call the commission() method to get and print the commission. If the sales are negative,**

**your demo should print the message “Invalid Input”.**

**Source Code :**

import java.util.Scanner;

class Commission {

private double sales;

public Commission(double sales) {

this.sales = sales;}

public double commission() { // Method to calculate commission

if (sales < 0) {

return -1; // Indicates invalid input

} else {

return sales \* 0.10; // Assuming commission rate is 10%

}}}

public class commission\_demo {

public static void main(String[] args) {

Scanner sc = new Scanner(System.in);

System.out.print("Enter the sales amount: ");

double sales = sc.nextDouble();

if (sales < 0) { // Validate sales

System.out.println("Invalid Input");

} else {

Commission ob = new Commission(sales);

double amount = ob.commission();

if (amount == -1) {

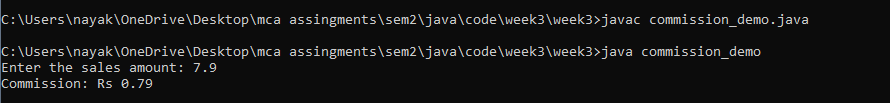
System.out.println("Invalid Input");

} else {

System.out.println("Commission: Rs " + amount); }}

sc.close();}}

**Output :**



**Week 4**

**Question 1 : Write a Java program to implement the concept of inheritance.**

**Source Code :**

class stu {

String name;

public void message() {

System.out.println("I am student of MCA");

}}

class Name extends stu { // inherit from stu

public void display() { // new method in subclass

System.out.println("My name is " + name);

}}

class inheritance\_eg {

public static void main(String[] args) {

Name ob = new Name(); // create an object of the subclass

ob.name = "Sahin";

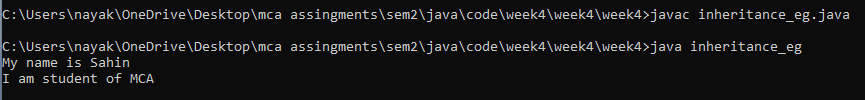
ob.display();

// call method of superclass

// using object of subclass

ob.message();}}

**Output :**



**Question 2 : Write a Java program to show method overloading.**

**Source Code :**

import java.util.Scanner;

public class method\_overloading {

static int add(int a, int b) {// Method with two integer parameters

return a + b; }

static int add(int a, int b, int c) { // Method with three integer parameters

return a + b + c; }

static double add(double a, double b){ // Method with two double parameters

return a + b; }

public static void main(String[] args){

Scanner sc= new Scanner(System.in);

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

int num1 = sc.nextInt();

int num2 = sc.nextInt();

System.out.println("Sum of " + num1 + " and " + num2 + " is: " + add(num1, num2));

System.out.println("Enter three integers:");

int num3 = sc.nextInt();

int num4 = sc.nextInt();

int num5 = sc.nextInt();

System.out.println("Sum of " + num3 + ", " + num4 + " and " + num5 + " is: " + add(num3, num4, num5));

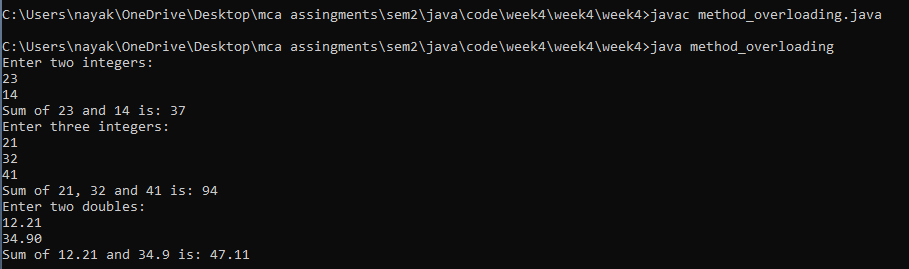
System.out.println("Enter two doubles:");

double num6 = sc.nextDouble();

double num7 = sc.nextDouble();

System.out.println("Sum of " + num6 + " and " + num7 + " is: " + add(num6, num7)); }}

**Output :**



**Question 3 : Write a Java program to show method overriding.**

**Source Code :**

import java.util.Scanner;

class Animal {

public void sound() {// Method to make sound

System.out.println("Animal makes a sound"); }}

class Dog extends Animal { // Child class inheriting from Animal

public void sound() { // Overriding the makeSound method

System.out.println("Dog barks"); }}

class Cat extends Animal { // Child class inheriting from Animal

public void sound() {

System.out.println("Cat meows"); }}

public class method\_overriding {

public static void main(String[] args) {

Scanner sc= new Scanner(System.in);

System.out.println("Enter 'dog' or 'cat' to hear the sound: ");

String aype = sc.nextLine();

Animal ob;// Creating object of Animal class

if (aype.equalsIgnoreCase("dog")) { // Depending on user input, create an object of Dog or Cat class

ob= new Dog(); // Dog object created

} else if (aype.equalsIgnoreCase("cat")) {

ob= new Cat(); // Cat object created

} else {

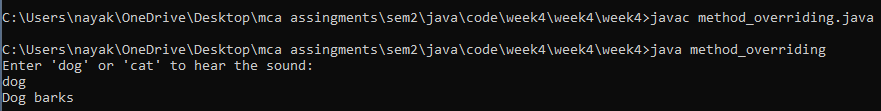
System.out.println("Invalid input. Please enter 'dog' or 'cat'.");

return; }

ob.sound(); // Invoke makeSound method of the respective class

sc.close();}}

**Output :**

****

**Question 4 : Write a Java program to show method hiding.**

**Source Code :**

class Superclass {

static void display() {

System.out.println("Static method in Superclass");}}

class Subclass extends Superclass {

static void display() {

System.out.println("Static method in Subclass");}}

public class method\_hiding {

public static void main(String[] args) {

Superclass.display(); // Call the static method in Superclass

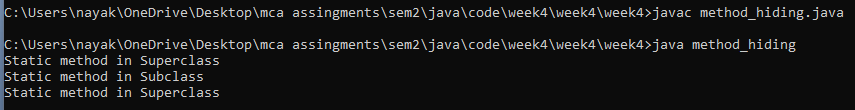
Subclass.display(); // Call the static method in Subclass

// Access the static method in Subclass using Superclass reference

Superclass ref = new Subclass();

ref.display();}}

**Output :**



**Question 5 : Create a general class ThreeDObject and derive the classes Box, Cube, Cylinder and Cone from it. The class**

**ThreeDObject has methods wholeSurfaceArea ( ) and volume ( ). Override these two methods in each of the**

**derived classes to calculate the volume and whole surface area of each type of three-dimensional objects. The**

**dimensions of the objects are to be taken from the users and passed through the respective constructors of**

**each derived class. Write a main method to test these classes.**

**Source Code :**

import java.util.Scanner;

class ThreeDObject {

ThreeDObject() {} // Default constructor

double wholeSurfaceArea() { // Method to calculate whole surface area

return 0.0; }

double volume() { // Method to calculate volume

return 0.0;}}

class Box extends ThreeDObject {

double length, width, height;

Box(double l, double w, double h) {

length = l;

width = w;

height = h; }

double wholeSurfaceArea() { // Override method to calculate whole surface area

return 2 \* (length \* width + length \* height + width \* height); }

double volume() { // Override method to calculate volume

return length \* width \* height; }}

class Cube extends Box {

Cube(double side) {

super(side, side, side); }}

class Cylinder extends ThreeDObject {

double radius, height;

Cylinder(double r, double h) {

radius = r;

height = h;}

@Override

double wholeSurfaceArea() { // Override method to calculate whole surface area

return 2 \* Math.PI \* radius \* (radius + height);}

double volume() {

return Math.PI \* radius \* radius \* height;}}

class Cone extends Cylinder {

Cone(double r, double h) {

super(r, h);}

double wholeSurfaceArea() {

return Math.PI \* radius \* (radius + Math.sqrt(radius \* radius + height \* height));}

double volume() { // Override method to calculate volume

return (1.0/3.0) \* Math.PI \* radius \* radius \* height;}}

public class three\_objects {

public static void main(String[] args) {

Scanner sc = new Scanner(System.in);

System.out.println("Enter length, width, and height of the Box:");

double boxLength = sc.nextDouble();

double boxWidth = sc.nextDouble();

double boxHeight = sc.nextDouble();

Box b = new Box(boxLength, boxWidth, boxHeight);

System.out.println("Volume of Box: " + b.volume());

System.out.println("Whole Surface Area of Box: " + b.wholeSurfaceArea());

System.out.println("\nEnter side of the Cube:");

double cubeSide = sc.nextDouble();

Cube c = new Cube(cubeSide);

System.out.println("Volume of Cube: " + c.volume());

System.out.println("Whole Surface Area of Cube: " + c.wholeSurfaceArea());

System.out.println("\nEnter radius and height of the Cylinder:");

double cylinderRadius = sc.nextDouble();

double cylinderHeight = sc.nextDouble();

Cylinder cy = new Cylinder(cylinderRadius, cylinderHeight);

System.out.println("Volume of Cylinder: " + cy.volume());

System.out.println("Whole Surface Area of Cylinder: " + cy.wholeSurfaceArea());

System.out.println("\nEnter radius and height of the Cone:");

double coneRadius = sc.nextDouble();

double coneHeight = sc.nextDouble();

Cone cone = new Cone(coneRadius, coneHeight);

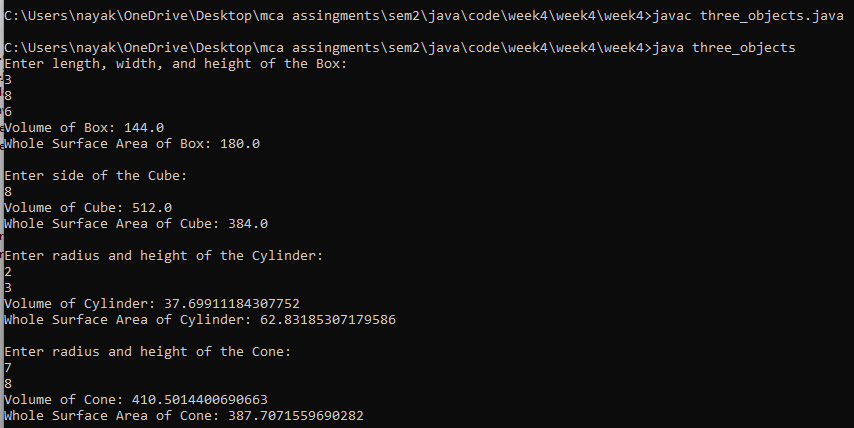
System.out.println("Volume of Cone: " + cone.volume());

System.out.println("Whole Surface Area of Cone: " + cone.wholeSurfaceArea());

sc.close();

}}

**Output :**

****

**Question 6 : Write a program to create a class named Vehicle having protected instance variables regnNumber, speed, color,**

**ownerName and a method showData ( ) to show “This is a vehicle class”. Inherit the Vehicle class into subclasses named Bus and Car having individual private instance variables routeNumber in Bus and manufacturerName in Car and both of them having showData ( ) method showing all details of Bus and Car respectively with content of the super class’s showData ( ) method.**

**Source Code :**

import java.util.Scanner;

class Vehicle {

protected String regnNumber;

protected int speed;

protected String color;

protected String ownerName;

public Vehicle(String regnNumber, int speed, String color, String ownerName) {

this.regnNumber = regnNumber;

this.speed = speed;

this.color = color;

this.ownerName = ownerName;}

protected void showData() {

System.out.println("This is a vehicle class");}}

class Bus extends Vehicle {

private String routeNumber;

public Bus(String regnNumber, int speed, String color, String ownerName, String routeNumber) {

super(regnNumber, speed, color, ownerName);

this.routeNumber = routeNumber;}

public void showData() {

super.showData();

System.out.println("Registration Number: " + regnNumber);

System.out.println("Speed: " + speed);

System.out.println("Color: " + color);

System.out.println("Owner Name: " + ownerName);

System.out.println("Route Number: " + routeNumber);}}

class Car extends Vehicle {

private String manufacturerName;

public Car(String regnNumber, int speed, String color, String ownerName, String manufacturerName) {

super(regnNumber, speed, color, ownerName);

this.manufacturerName = manufacturerName;}

public void showData() {

super.showData();

System.out.println("Registration Number: " + regnNumber);

System.out.println("Speed: " + speed);

System.out.println("Color: " + color);

System.out.println("Owner Name: " + ownerName);

System.out.println("Manufacturer Name: " + manufacturerName);}}

public class vehicle\_bus\_car {

public static void main(String[] args) {

Scanner sc = new Scanner(System.in);

System.out.println("Enter Bus Details:");

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

String busRegnNumber = sc.nextLine();

System.out.print("Speed: ");

int busSpeed = sc.nextInt();

sc.nextLine(); // Consume newline

System.out.print("Color: ");

String busColor = sc.nextLine();

System.out.print("Owner Name: ");

String busOwnerName = sc.nextLine();

System.out.print("Route Number: ");

String busRouteNumber = sc.nextLine();

Bus b = new Bus(busRegnNumber, busSpeed, busColor, busOwnerName, busRouteNumber);

System.out.println("\nBus Details which is entered by the user are:");

b.showData();

System.out.println("\nEnter Car Details:");

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

String carRegnNumber = sc.nextLine();

System.out.print("Speed: ");

int carSpeed = sc.nextInt();

sc.nextLine(); // Consume newline

System.out.print("Color: ");

String carColor = sc.nextLine();

System.out.print("Owner Name: ");

String carOwnerName = sc.nextLine();

System.out.print("Manufacturer Name: ");

String carManufacturerName = sc.nextLine();

Car c = new Car(carRegnNumber, carSpeed, carColor, carOwnerName, carManufacturerName);

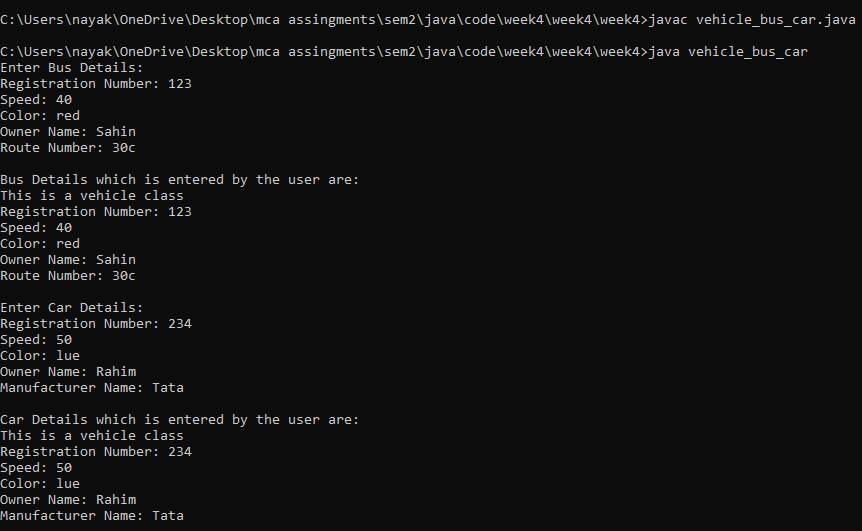
System.out.println("\nCar Details which is entered by the user are:");

c.showData();

sc.close();

}}

**Output :**



**Question 7 : An educational institution maintains a database of its employees. The database is divided into a number of**

**classes whose hierarchical relationships are shown below. Write all the classes and define the methods to**

**create the database and retrieve individual information as and when needed. Write a driver program to test the**

**classes.**

**Source Code :**

import java.util.\*;

class Staff { // Base class Staff

protected int code;

protected String name;

public Staff(int code, String name) {

this.code = code;

this.name = name;}

public void display() {

System.out.println("Code: " + code);

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

class Teacher extends Staff { // Subclass Teacher

private String subject;

private String publication;

public Teacher(int code, String name, String subject, String publication) {

super(code, name);

this.subject = subject;

this.publication = publication;}

public void display() {

super.display();

System.out.println("Subject: " + subject);

System.out.println("Publication: " + publication);}}

class Officer extends Staff {// Subclass Officer

private String grade;

public Officer(int code, String name, String grade) {

super(code, name);

this.grade = grade;}

public void display() {

super.display();

System.out.println("Grade: " + grade);}}

class Typist extends Staff {// Subclass Typist

private int speed;

public Typist(int code, String name, int speed) {

super(code, name);

this.speed = speed;}

public void display() {

super.display();

System.out.println("Speed: " + speed);}}

class RegularTypist extends Typist {// Subclass RegularTypist

private double remuneration;

public RegularTypist(int code, String name, int speed, double remuneration) {

super(code, name, speed);

this.remuneration = remuneration;}

public void display() {

super.display();

System.out.println("Remuneration: " + remuneration);}}

class CasualTypist extends Typist {// Subclass CasualTypist

private double dailyWages;

public CasualTypist(int code, String name, int speed, double dailyWages) {

super(code, name, speed);

this.dailyWages = dailyWages;}

public void display() {

super.display();

System.out.println("Daily Wages: " + dailyWages);}}

public class staff\_teacher\_officer\_typist {

public static void main(String[] args) {

Scanner sc = new Scanner(System.in);

System.out.println("Enter details for Teacher:");

System.out.print("Code: ");

int teacherCode = sc.nextInt();

sc.nextLine(); // Consume newline

System.out.print("Name: ");

String teacherName = sc.nextLine();

System.out.print("Subject: ");

String subject = sc.nextLine();

System.out.print("Publication: ");

String publication = sc.nextLine();

Teacher teacher = new Teacher(teacherCode, teacherName, subject, publication);

System.out.println("\nEnter details for Officer:");

System.out.print("Code: ");

int officerCode = sc.nextInt();

sc.nextLine(); // Consume newline

System.out.print("Name: ");

String officerName = sc.nextLine();

System.out.print("Grade: ");

String grade = sc.nextLine();

Officer officer = new Officer(officerCode, officerName, grade);

System.out.println("\nEnter details for Regular Typist:");

System.out.print("Code: ");

int regularTypistCode = sc.nextInt();

sc.nextLine(); // Consume newline

System.out.print("Name: ");

String regularTypistName = sc.nextLine();

System.out.print("Speed: ");

int speed = sc.nextInt();

System.out.print("Remuneration: ");

double remuneration = sc.nextDouble();

RegularTypist regularTypist = new RegularTypist(regularTypistCode, regularTypistName, speed, remuneration);

System.out.println("\nEnter details for Casual Typist:");

System.out.print("Code: ");

int casualTypistCode = sc.nextInt();

sc.nextLine(); // Consume newline

System.out.print("Name: ");

String casualTypistName = sc.nextLine();

System.out.print("Speed: ");

int casualTypistSpeed = sc.nextInt();

System.out.print("Daily Wages: ");

double dailyWages = sc.nextDouble();

CasualTypist casualTypist = new CasualTypist(casualTypistCode, casualTypistName, casualTypistSpeed, dailyWages);

System.out.println("\nDetails of entered employees:");

System.out.println("\nTeacher:");

teacher.display();

System.out.println("\nOfficer:");

officer.display();

System.out.println("\nRegular Typist:");

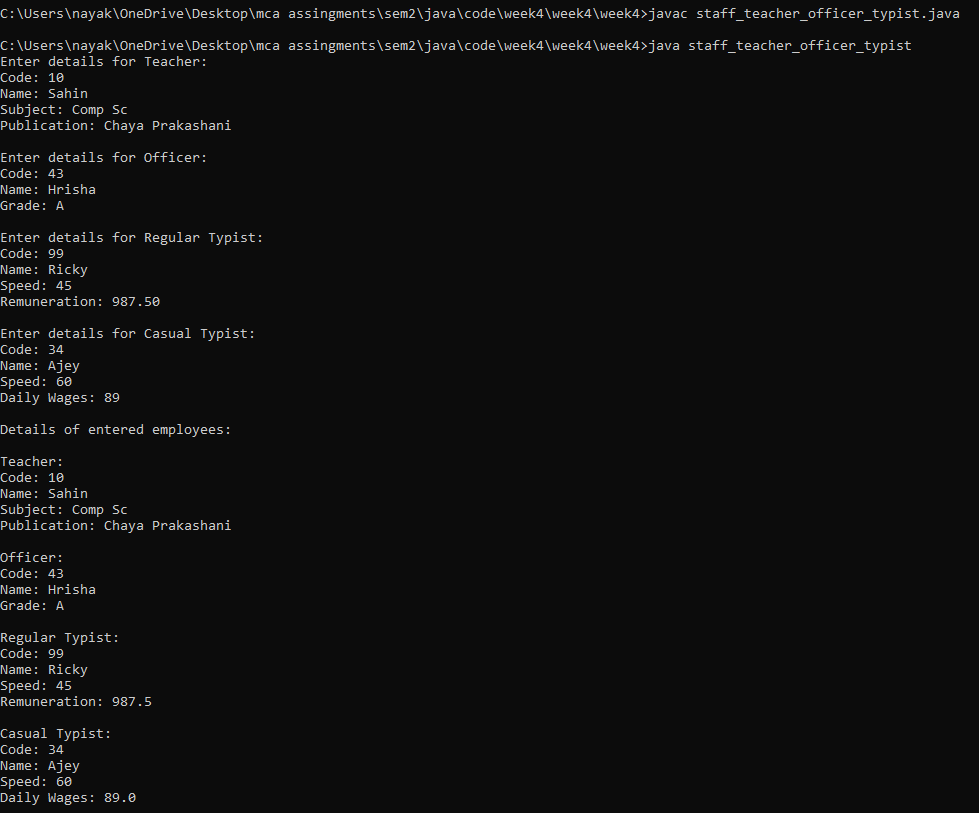
regularTypist.display();

System.out.println("\nCasual Typist:");

casualTypist.display();

sc.close();}}

**Output :**



**Question 8 : Create a base class Building that stores the number of floors of a building, number of rooms and it’s total**

**footage. Create a derived class House that inherits Building and also stores the number of bedrooms and**

**bathrooms. Demonstrate the working of the classes.**

**Source Code :**

import java.util.Scanner;

// Base class Building

class Building {

protected int Floors;

protected int Rooms;

protected double Footage;

public Building(int Floors, int Rooms, double Footage) {

this.Floors = Floors;

this.Rooms = Rooms;

this.Footage = Footage; }

public void display() {

System.out.println("Number of Floors: " + Floors);

System.out.println("Number of Rooms: " + Rooms);

System.out.println("Total Footage: " + Footage + " sqft"); }}

class House extends Building {

private int Bedrooms;

private int Bathrooms;

public House(int Floors, int Rooms, double Footage, int Bedrooms, int Bathrooms) {

super(Floors, Rooms, Footage);

this.Bedrooms = Bedrooms;

this.Bathrooms = Bathrooms;}

public void display() {

super.display();

System.out.println("Number of Bedrooms: " + Bedrooms);

System.out.println("Number of Bathrooms: " + Bathrooms);}}

public class building\_house {

public static void main(String[] args) {

Scanner sc = new Scanner(System.in);

System.out.println("Enter details for Building:");

System.out.print("Number of Floors: ");

int Floors = sc.nextInt();

System.out.print("Number of Rooms: ");

int Rooms = sc.nextInt();

System.out.print("Total Footage: ");

double Footage = sc.nextDouble();

System.out.println("\nEnter details for House:");

System.out.print("Number of Bedrooms: ");

int Bedrooms = sc.nextInt();

System.out.print("Number of Bathrooms: ");

int Bathrooms = sc.nextInt();

Building b = new Building(Floors, Rooms, Footage);

House h= new House(Floors, Rooms, Footage, Bedrooms, Bathrooms);

System.out.println("\nDetails of Building:");

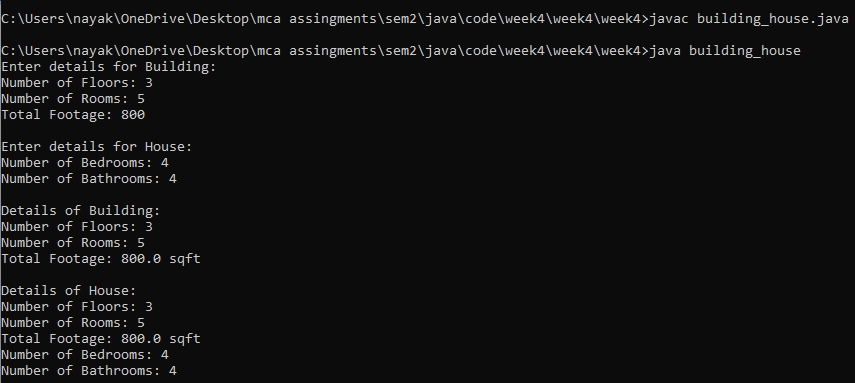
b.display();

System.out.println("\nDetails of House:");

h.display();

sc.close();}}

**Output :**

****

**Question 9 :**  **In the earlier program, create a second derived class Office that inherits Building and stores the number of**

**telephones and tables. Now demonstrate the working of all three classes.**

**Source Code :**

import java.util.Scanner;

class Building {

protected int Floors;

protected int Rooms;

protected double Footage;

public Building(int Floors, int Rooms, double Footage) {

this.Floors = Floors;

this.Rooms = Rooms;

this.Footage = Footage;}

public void display() {

System.out.println("Number of Floors: " + Floors);

System.out.println("Number of Rooms: " + Rooms);

System.out.println("Total Footage: " + Footage + " sqft");}}

class House extends Building {

private int Bedrooms;

private int Bathrooms;

public House(int Floors, int Rooms, double Footage, int Bedrooms, int Bathrooms) {

super(Floors, Rooms, Footage);

this.Bedrooms = Bedrooms;

this.Bathrooms = Bathrooms;}

public void display() {

super.display();

System.out.println("Number of Bedrooms: " + Bedrooms);

System.out.println("Number of Bathrooms: " + Bathrooms);}}

class Office extends Building { // Derived class Office

private int Telephones;

private int Tables;

public Office(int Floors, int Rooms, double Footage, int Telephones, int Tables) {

super(Floors, Rooms, Footage);

this.Telephones = Telephones;

this.Tables = Tables;}

public void display() {

super.display();

System.out.println("Number of Telephones: " + Telephones);

System.out.println("Number of Tables: " + Tables);}}

public class office\_building {

public static void main(String[] args) {

Scanner sc= new Scanner(System.in);

System.out.println("Enter details for Building:");

System.out.print("Number of Floors: ");

int Floors = sc.nextInt();

System.out.print("Number of Rooms: ");

int Rooms = sc.nextInt();

System.out.print("Total Footage: ");

double Footage = sc.nextDouble();

System.out.println("\nEnter details for House:");

System.out.print("Number of Bedrooms: ");

int Bedrooms = sc.nextInt();

System.out.print("Number of Bathrooms: ");

int Bathrooms = sc.nextInt();

System.out.println("\nEnter details for Office:");

System.out.print("Number of Telephones: ");

int Telephones = sc.nextInt();

System.out.print("Number of Tables: ");

int Tables = sc.nextInt();

Building b = new Building(Floors, Rooms, Footage);

House h = new House(Floors, Rooms, Footage, Bedrooms, Bathrooms);

Office o = new Office(Floors, Rooms, Footage, Telephones, Tables);

System.out.println("\nDetails of Building:");

b.display();

System.out.println("\nDetails of House:");

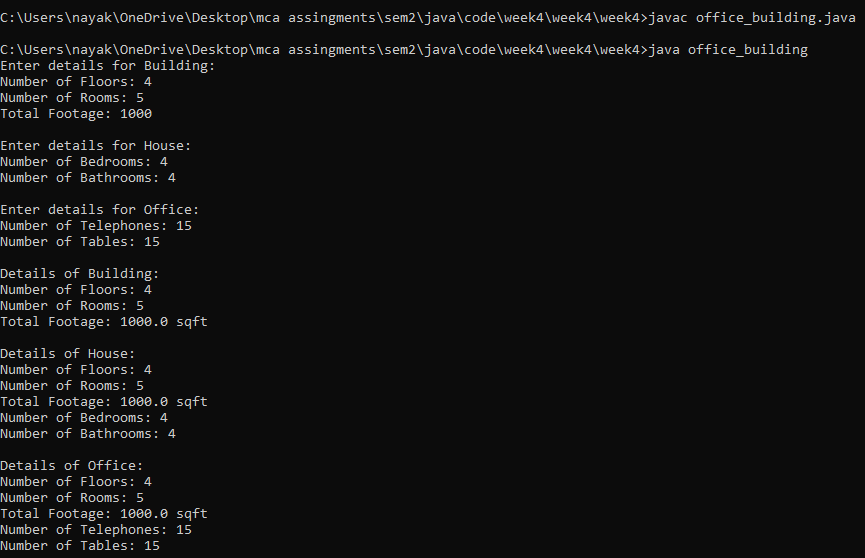
h.display();

System.out.println("\nDetails of Office:");

o.display();

sc.close();}}

**Output :**



**Question 10 : Write a Java program which creates a base class Num and contains an integer number along with a method**

**shownum() which displays the number. Now create a derived class HexNum which inherits Num and**

**overrides shownum() which displays the hexadecimal value of the number. Demonstrate the working of the**

**classes.**

**Source Code :**

class Superclass {

import java.util.Scanner;

class Num {

protected int number;

public Num(int number) {

this.number = number;}

public void showNum() {

System.out.println("Number: " + number); }}

class HexNum extends Num { // Derived class HexNum

public HexNum(int number) {

super(number); }

public void showNum() {

System.out.println("Hexadecimal Value: " + Integer.toHexString(number)); }}

public class num\_hexnum{

public static void main(String[] args) {

Scanner sc = new Scanner(System.in);

System.out.print("Enter an integer number: ");

int number = sc.nextInt();

Num num = new Num(number);// Creating objects

HexNum hexNum = new HexNum(number);

System.out.println("\nDisplaying number using Num:");

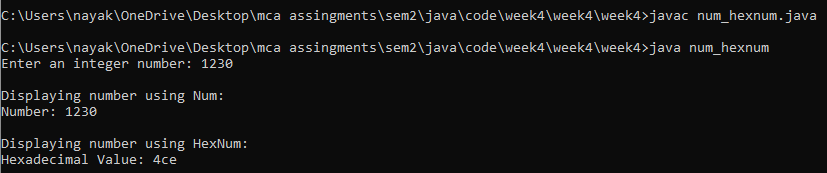
num.showNum();

System.out.println("\nDisplaying number using HexNum:");

hexNum.showNum();

sc.close(); }}

**Output :**

****

**Question 11 : Write a Java program which creates a base class Num and contains an integer number along with a method**

**shownum() which displays the number. Now create a derived class OctNum which inherits Num and**

**overrides shownum() which displays the octal value of the number. Demonstrate the working of the classes.**

**Source Code :**

import java.util.Scanner;

class Num {

protected int number;

public Num(int number) {

this.number = number;}

public void showNum() {

System.out.println("Number: " + number);}}

class OctNum extends Num { // Derived class OctNum

public OctNum(int number) {

super(number); }

public void showNum() {

System.out.println("Octal Value: " + Integer.toOctalString(number)); }}

public class num\_octnum {

public static void main(String[] args) {

Scanner sc = new Scanner(System.in);

System.out.print("Enter an integer number: ");

int number = sc.nextInt();

Num num = new Num(number);

OctNum octNum = new OctNum(number);

System.out.println("\nDisplaying number using Num:");

num.showNum();

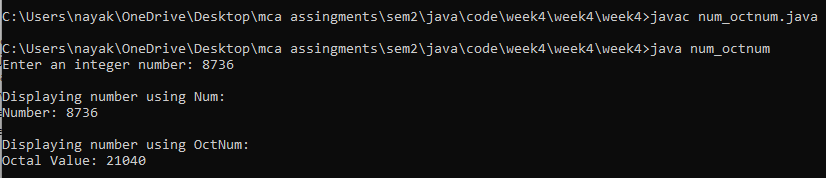
System.out.println("\nDisplaying number using OctNum:");

octNum.showNum();

sc.close();

}}

**Output :**

****

**Question 12 : Combine Question number 10 and 11 and have all the three classes together. Now describe the working of**

**all classes**

**Source Code :**

import java.util.Scanner;

class Num {

protected int number;

public Num(int number) {

this.number = number;}

public void showNum() {

System.out.println("Number: " + number); }}

class HexNum extends Num { // Derived class HexNum

public HexNum(int number) {

super(number); }

public void showNum() {

System.out.println("Hexadecimal Value: " + Integer.toHexString(number)); }}

class OctNum extends Num { // Derived class OctNum

public OctNum(int number) {

super(number);}

public void showNum() {

System.out.println("Octal Value: " + Integer.toOctalString(number)); }}

public class num\_hexnum\_octnum {

public static void main(String[] args) {

Scanner scanner = new Scanner(System.in);

System.out.print("Enter an integer number: ");

int number = scanner.nextInt();

Num num = new Num(number);

HexNum hexNum = new HexNum(number);

OctNum octNum = new OctNum(number);

System.out.println("\nDisplaying number using Num:");

num.showNum();

System.out.println("\nDisplaying number using HexNum:");

hexNum.showNum();

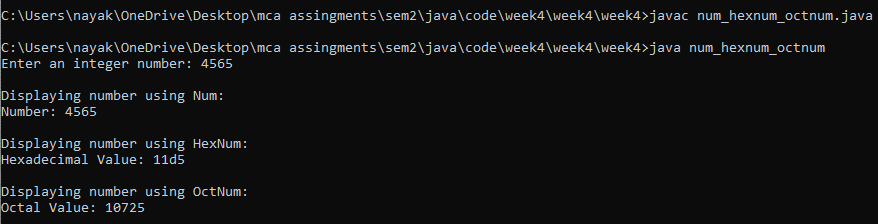
System.out.println("\nDisplaying number using OctNum:");

octNum.showNum();

scanner.close();

}}

**Output :**

****

**Question 13 : Create a base class Distance which stores the distance between two locations in miles and a method**

**travelTime(). The method prints the time taken to cover the distance when the speed is 60 miles per hour.**

**Now in a derived class DistanceMKS, override travelTime() so that it prints the time assuming the distance**

**is in kilometers and the speed is 100 km per second. Demonstrate the working of the classes**

**Source Code :**

import java.util.Scanner;

class Distance {

protected double distanceInMiles;

public Distance(double distanceInMiles) {

this.distanceInMiles = distanceInMiles;}

public void travelTime() {

double timeInHours = distanceInMiles / 60.0; // Speed is 60 miles per hour

System.out.println("Time taken to cover the distance: " + timeInHours + " hours");}}

class DistanceMKS extends Distance { // Derived class DistanceMKS

public DistanceMKS(double distanceInMiles) {

super(distanceInMiles);}

public void travelTime() {

double distanceInKilometers = distanceInMiles \* 1.60934; // Conversion from miles to kilometers

double timeInSeconds = distanceInKilometers / 100.0; // Speed is 100 kilometers per hour

System.out.println("Time taken to cover the distance: " + timeInSeconds + " seconds");}}

public class distance\_miles {

public static void main(String[] args) {

Scanner sc= new Scanner(System.in);

System.out.print("Enter the distance between two locations in miles: ");

double distanceInMiles = sc.nextDouble();

Distance distance = new Distance(distanceInMiles);

DistanceMKS distanceMKS = new DistanceMKS(distanceInMiles);

System.out.println("\nTravel time assuming speed is 60 miles per hour:");

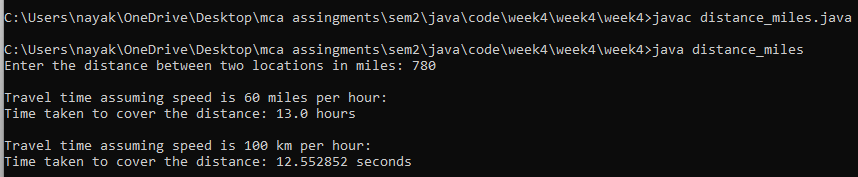
distance.travelTime();

System.out.println("\nTravel time assuming speed is 100 km per hour:");

distanceMKS.travelTime();

sc.close();}}

**Output :**

****

**Question 14 : Create a base class called “vehicle” that stores number of wheels and speed. Create the following derived**

**classes – “car” that inherits “vehicle” and also stores number of passengers. “truck” that inherits “vehicle”**

**and also stores the load limit. Write a main function to create objects of these two derived classes and**

**display all the information about “car” and “truck”. Also compare the speed of these two vehicles - car and**

**truck and display which one is faster.**

**Source Code :**

import java.util.Scanner;

class Vehicle {

protected int Wheels;

protected double speed;

public Vehicle(int Wheels, double speed) {

this.Wheels = Wheels;

this.speed = speed; }

public double getSpeed() {

return speed; }

public void displayInfo() {

System.out.println("Number of Wheels: " + Wheels);

System.out.println("Speed: " + speed + " mph"); }}

class Car extends Vehicle { // Derived class Car

private int Passengers;

public Car(int Wheels, double speed, int Passengers) {

super(Wheels, speed);

this.Passengers = Passengers; }

public void displayInfo() {

super.displayInfo();

System.out.println("Number of Passengers: " + Passengers);}}

class Truck extends Vehicle { // Derived class Truck

private double loadLimit;

public Truck(int Wheels, double speed, double loadLimit) {

super(Wheels, speed);

this.loadLimit = loadLimit;}

public void displayInfo() {

super.displayInfo();

System.out.println("Load Limit: " + loadLimit + " tons");}}

public class vehicle\_car\_truck {

public static void main(String[] args) {

Scanner sc = new Scanner(System.in);

System.out.println("Enter details for Car:");

System.out.print("Number of Wheels: ");

int carWheels = sc.nextInt();

System.out.print("Speed (mph): ");

double carSpeed = sc.nextDouble();

System.out.print("Number of Passengers: ");

int passengers = sc.nextInt();

System.out.println("\nEnter details for Truck:");

System.out.print("Number of Wheels: ");

int truckWheels = sc.nextInt();

System.out.print("Speed (mph): ");

double truckSpeed = sc.nextDouble();

System.out.print("Load Limit (tons): ");

double loadLimit = sc.nextDouble();

Car c = new Car(carWheels, carSpeed, passengers);

Truck t = new Truck(truckWheels, truckSpeed, loadLimit);

System.out.println("\nInformation about Car:");

c.displayInfo();

System.out.println("\nInformation about Truck:");

t.displayInfo();

if (c.getSpeed() > t.getSpeed()) { // Comparing speeds

System.out.println("\nCar is faster than Truck.");

} else if (c.getSpeed() < t.getSpeed()) {

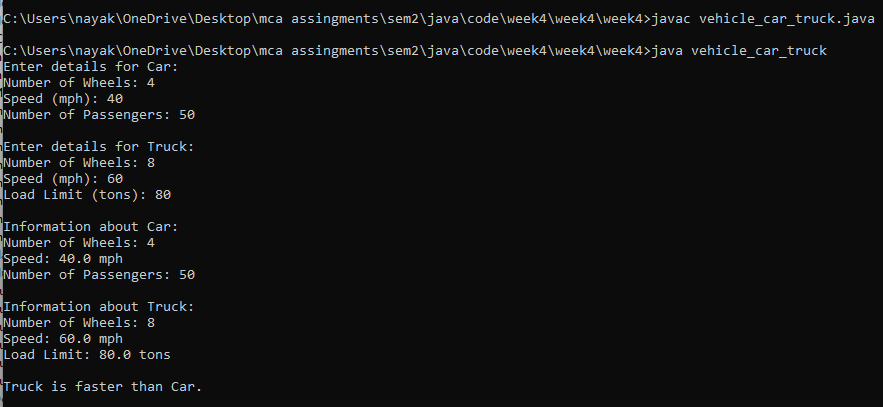
System.out.println("\nTruck is faster than Car.");

} else {

System.out.println("\nCar and Truck have the same speed.");}

sc.close(); }}

**Output :**

****

**Question 15 : Write a Java program to explain “multilevel inheritance.”**

**Source Code :**

class Animal {

public void eat() {

System.out.println("Animal is eating."); }}

class Mammal extends Animal { // Derived class inheriting from Animal

public void walk() {

System.out.println("Mammal is walking."); }}

class Dog extends Mammal { // Further derived class inheriting from Mammal

public void bark() {

System.out.println("Dog is barking."); }}

public class multilevel\_inheritance{

public static void main(String[] args) {

Dog dog = new Dog();

// Calling methods from Animal, Mammal, and Dog classes

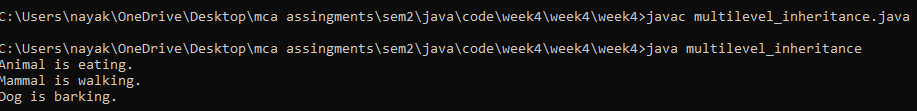
dog.eat(); // Inherited from Animal class

dog.walk(); // Inherited from Mammal class

dog.bark(); // Defined in Dog class

}}

**Output :**

****

**Week 5**

**Question 1 : Create a “circle” class & a “point” class. The coordinates of the circle are given and used within the “circle”**

**class as object of the “point” class. Display the area of circle.**

**Source Code :**

import java.util.Scanner;

class Point {

private double x;

private double y;

public Point(double x, double y) {

this.x = x;

this.y = y;}

public double getX() {

return x;}

public double getY() {

return y; }}

class Circle {

private Point center;

private double radius;

public Circle(Point center, double radius) {

this.center = center;

this.radius = radius;}

public double getArea() {

return Math.PI \* radius \* radius;}}

public class circle\_point {

public static void main(String[] args) {

Scanner sc = new Scanner(System.in);

System.out.println("Enter the x-coordinate of the center:");

double x = sc.nextDouble();

System.out.println("Enter the y-coordinate of the center:");

double y = sc.nextDouble();

System.out.println("Enter the radius of the circle:");

double radius = sc.nextDouble();

Point center = new Point(x, y);

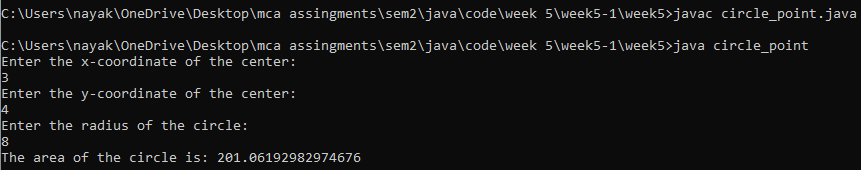
Circle ob= new Circle(center, radius);

double area = ob.getArea();

System.out.println("The area of the circle is: " + area);

sc.close(); }}

**Output :**

****

**Question 2 : Create a class called Time, which has three private instance variables – hour, min and sec. It contains a method**

**called add( ) which takes one Time object as parameter and print the added value of the calling Time object and**

**passes Time object. In the main method, declare two Time objects and assign values using constructor and call**

**the add() method.**

**Source Code :**

import java.util.Scanner;

class Time {

private int hour;

private int min;

private int sec;

public Time(int hour, int min, int sec) {

this.hour = hour;

this.min = min;

this.sec = sec;}

public void add(Time other) {

this.sec += other.sec;

this.min += other.min + this.sec / 60;

this.hour += other.hour + this.min / 60;

this.sec %= 60;

this.min %= 60;

this.hour %= 24;

System.out.println("Added time: " + this.hour + " hours, " + this.min + " minutes, " + this.sec + " seconds"); }}

public class time\_ {

public static void main(String[] args) {

Scanner sc = new Scanner(System.in);

System.out.println("Enter time for first object (hours minutes seconds):");

int hour1 = sc.nextInt();

int min1 = sc.nextInt();

int sec1 = sc.nextInt();

System.out.println("Enter time for second object (hours minutes seconds):");

int hour2 = sc.nextInt();

int min2 = sc.nextInt();

int sec2 = sc.nextInt();

Time ob1 = new Time(hour1, min1, sec1);

Time ob2 = new Time(hour2, min2, sec2);

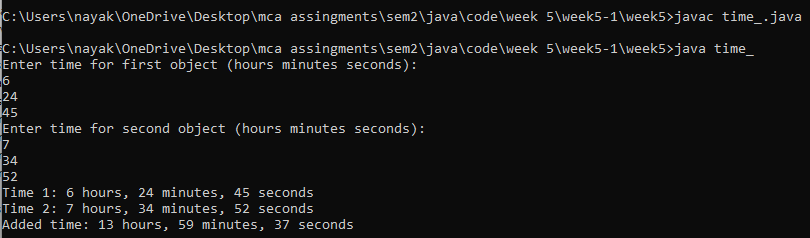
System.out.println("Time 1: " + hour1 + " hours, " + min1 + " minutes, " + sec1 + " seconds");

System.out.println("Time 2: " + hour2 + " hours, " + min2 + " minutes, " + sec2 + " seconds");

ob1.add(ob2);

sc.close();}}

**Output :**

****

**Question 3 : Create a class called Complex, which has three private instance variables –real and imaginary. It contains a**

**method called add( ) which takes one Complex object as parameter and print the added value of the calling**

**Complex object and passes Complex object. In the main method, declare two Complex objects and assign**

**values using constructor and call the add() method.**

**Source Code :**

import java.util.Scanner;

class Complex {

private double real;

private double imaginary;

public Complex(double real, double imaginary) {

this.real = real;

this.imaginary = imaginary; }

public void add(Complex other) {

double sumReal = this.real + other.real;

double sumImaginary = this.imaginary + other.imaginary;

System.out.println("Sum: " + sumReal + " + " + sumImaginary + "i"); }}

public class complex\_ {

public static void main(String[] args) {

Scanner sc = new Scanner(System.in);

System.out.println("Enter real and imaginary parts for first complex number:");

double real1 = sc.nextDouble();

double imaginary1 = sc.nextDouble();

System.out.println("Enter real and imaginary parts for second complex number:");

double real2 = sc.nextDouble();

double imaginary2 = sc.nextDouble();

Complex ob1 = new Complex(real1, imaginary1);

Complex ob2 = new Complex(real2, imaginary2);

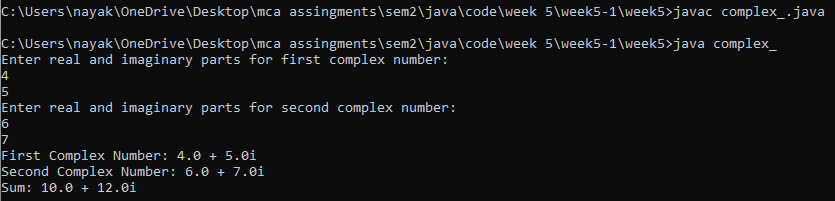
System.out.println("First Complex Number: " + real1 + " + " + imaginary1 + "i");

System.out.println("Second Complex Number: " + real2 + " + " + imaginary2 + "i");

ob1.add(ob2);

sc.close(); }}

**Output :**

****

**Question 4 : Write a program to define a class having one 3-digit number, num as data member. Initialize and display**

**reverse of that number.**

**Source Code :**

import java.util.Scanner;

class ThreeDigitNumber {

private int num;

public ThreeDigitNumber(int num) {

if (num < 100 || num > 999) {

System.out.println("Error: Number must be a 3-digit number.");

System.exit(1); // Exit the program with an error status

}

this.num = num;}

public void reverseAndDisplay() {

int originalNum = num;

int reverse = 0;

while (originalNum != 0) {

int digit = originalNum % 10;

reverse = reverse \* 10 + digit;

originalNum /= 10;}

System.out.println("Original Number: " + num);

System.out.println("Reverse of the Number: " + reverse);}}

public class reverse\_number{

public static void main(String[] args) {

Scanner sc= new Scanner(System.in);

System.out.print("Enter a 3-digit number: ");

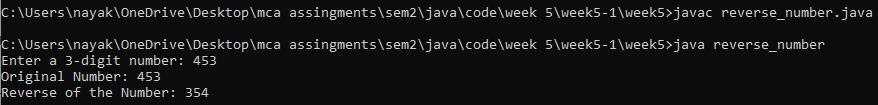
int number = sc.nextInt();

ThreeDigitNumber obj = new ThreeDigitNumber(number);

obj.reverseAndDisplay();

sc.close(); }}

**Output :**

****

**Question 5 : Write a program to define a class Student with four data members such as name, roll no., sub1, and sub2.**

**Define appropriate methods to initialize and display the values of data members. Also calculate total marks**

**and percentage scored by student.**

**Source Code :**

import java.util.Scanner;

class Student {

private String name;

private int rollNo;

private int sub1;

private int sub2;

public void initialize(String name, int rollNo, int sub1, int sub2) {

this.name = name;

this.rollNo = rollNo;

this.sub1 = sub1;

this.sub2 = sub2;}

public void display() {

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

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

System.out.println("Marks in Subject 1: " + sub1);

System.out.println("Marks in Subject 2: " + sub2);}

public int calculateTotalMarks() {

return sub1 + sub2;}

public double calculatePercentage() {

int totalMarks = calculateTotalMarks();

return (totalMarks / 2.0); // Considering two subjects, hence dividing by 2.0 for decimal result

}}

public class student\_total\_percent {

public static void main(String[] args) {

Scanner sc = new Scanner(System.in);

System.out.println("Enter student details:");

System.out.print("Name: ");

String name = sc.nextLine();

System.out.print("Roll No: ");

int rollNo = sc.nextInt();

System.out.print("Marks in Subject 1: ");

int sub1 = sc.nextInt();

System.out.print("Marks in Subject 2: ");

int sub2 = sc.nextInt();

Student student = new Student();

student.initialize(name, rollNo, sub1, sub2);

System.out.println("\nStudent Details:");

student.display();

int totalMarks = student.calculateTotalMarks();

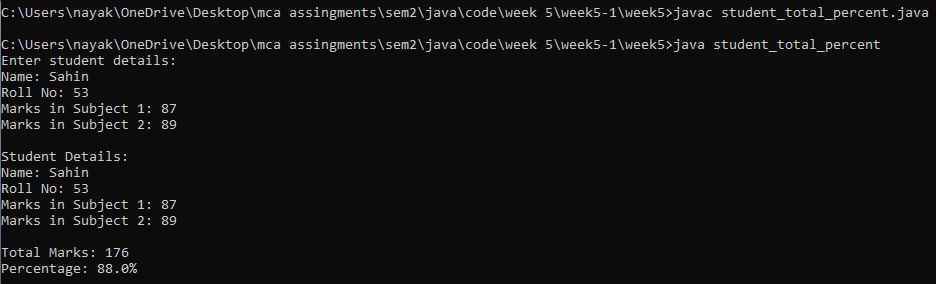
double percentage = student.calculatePercentage();

System.out.println("\nTotal Marks: " + totalMarks);

System.out.println("Percentage: " + percentage + "%");

sc.close(); }}

**Output :**

****

**Question 6 : Write a program to define a class Employee to accept emp\_id, emp \_name, basic\_salary from the user and**

**display the gross\_salary.**

**Source Code :**

import java.util.Scanner;

class Employee {

private int empId;

private String empName;

private double basicSalary;

public void acceptDetails(int empId, String empName, double basicSalary) {

this.empId = empId;

this.empName = empName;

this.basicSalary = basicSalary; }

public double calculateGrossSalary() {

// Assuming 20% of basic salary as allowance

double allowance = 0.2 \* basicSalary;

double grossSalary = basicSalary + allowance;

return grossSalary; }

public void displayGrossSalary() {

double grossSalary = calculateGrossSalary();

System.out.println("Employee ID: " + empId);

System.out.println("Employee Name: " + empName);

System.out.println("Basic Salary: " + basicSalary);

System.out.println("Gross Salary: " + grossSalary); }}

public class employee\_print{

public static void main(String[] args) {

Scanner sc = new Scanner(System.in);

System.out.print("Enter Employee ID: ");

int empId = sc.nextInt();

sc.nextLine(); // Consume newline character

System.out.print("Enter Employee Name: ");

String empName = sc.nextLine();

System.out.print("Enter Basic Salary: ");

double basicSalary = sc.nextDouble();

Employee ob = new Employee();

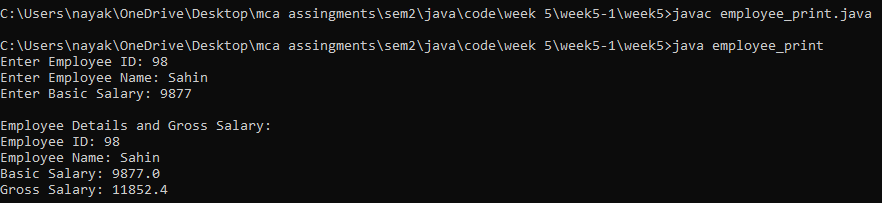
ob.acceptDetails(empId, empName, basicSalary);

System.out.println("\nEmployee Details and Gross Salary:");

ob.displayGrossSalary();

sc.close();}}

**Output :**

****

**Question 7 : Write a program to define a class Fraction having data members numerator and denominator. Initialize three**

**objects using different constructors and display its fractional value.**

**Source Code :**

class Fraction {

private int numerator;

private int denominator;

public Fraction() {

numerator = 1;

denominator = 1;}

public Fraction(int numerator, int denominator) {

this.numerator = numerator;

this.denominator = denominator != 0 ? denominator : 1;}

public Fraction(int numerator) {

this.numerator = numerator;

this.denominator = 1;}

public void displayFraction() {

System.out.println(numerator + "/" + denominator);}}

public class fraction\_{

public static void main(String[] args) {

Fraction ob1 = new Fraction();

Fraction ob2 = new Fraction(3, 4);

Fraction ob3 = new Fraction(5);

System.out.print("Fraction 1: ");

ob1.displayFraction();

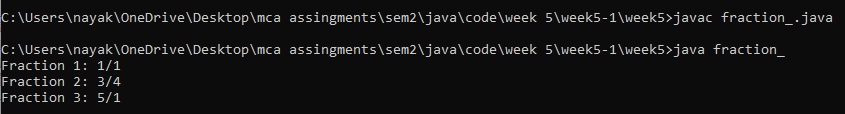
System.out.print("Fraction 2: ");

ob2.displayFraction();

System.out.print("Fraction 3: ");

ob3.displayFraction();}}

**Output :**

****

**Question 8 : Write a program to define a class Item containing code and price. Accept this data for five objects using**

**array of objects. Display code, price in tabular form and also, display total price of all items**.

**Source Code :**

import java.util.Scanner;

class Item {

private String code;

private double price;

public Item(String code, double price) {

this.code = code;

this.price = price;}

public String getCode() {

return code;}

public double getPrice() {

return price;}}

public class item\_code\_price {

public static void main(String[] args) {

Scanner sc = new Scanner(System.in);

Item[] items = new Item[5]; // Create an array of Item objects to store data for five items

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

System.out.println("Enter details for Item " + (i + 1) + ":");

System.out.print("Code: ");

String code = sc.next();

System.out.print("Price: ");

double price = sc.nextDouble();

items[i] = new Item(code, price);}

System.out.println("\nCode\tPrice");

double totalPrice = 0;

for (Item item : items) {

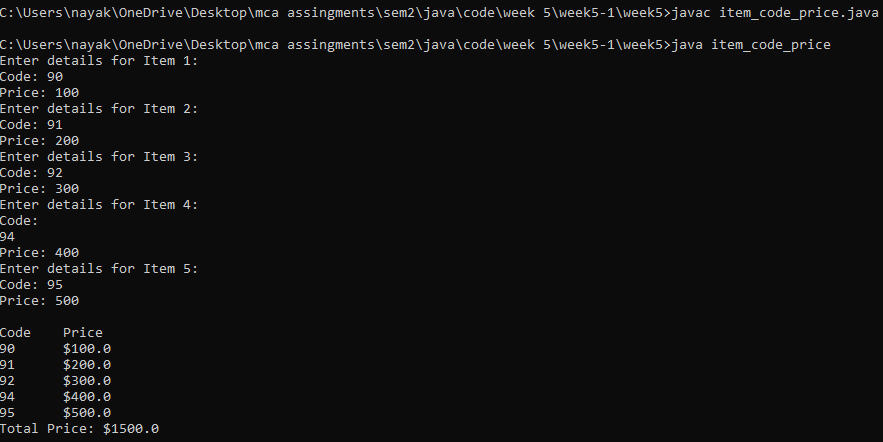
System.out.println(item.getCode() + "\t$" + item.getPrice());

totalPrice += item.getPrice();}

System.out.println("Total Price: $" + totalPrice);

sc.close(); }}

**Output :**

****

**Question 9 : Write a program to define a class Tender containing data members cost and company name. Accept data for**

**five objects and display company name for which cost is minimum.**

**Source Code :**

import java.util.Scanner;

class Tender {

private double cost;

private String companyName;

public Tender(String companyName, double cost) {

this.companyName = companyName;

this.cost = cost;}

public double getCost() {

return cost;}

public String getCompanyName() {

return companyName; }}

public class tender\_cost\_companyname {

public static void main(String[] args) {

Scanner sc = new Scanner(System.in);

Tender[] tenders = new Tender[5];

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

System.out.println("Enter details for Tender " + (i + 1) + ":");

System.out.print("Company Name: ");

String companyName = sc.nextLine();

System.out.print("Cost: ");

double cost = sc.nextDouble();

tenders[i] = new Tender(companyName, cost);

sc.nextLine(); } // Consume newline character

Tender minCostTender = tenders[0];

for (int i = 1; i < tenders.length; i++) {

if (tenders[i].getCost() < minCostTender.getCost()) {

minCostTender = tenders[i];}}

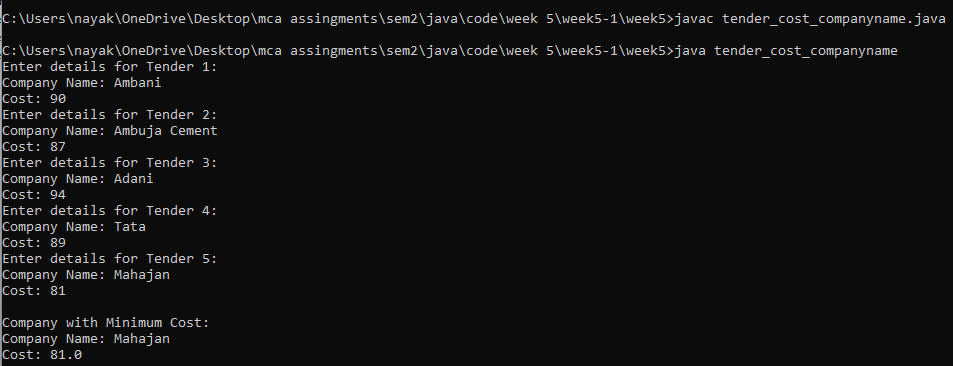
System.out.println("\nCompany with Minimum Cost:");

System.out.println("Company Name: " + minCostTender.getCompanyName());

System.out.println("Cost: " + minCostTender.getCost());

sc.close();}}

**Output :**

****

**Question 10 : Write a program to define a class 'employee' with data members as empid, name and salary. Accept data**

**for 5 objects using Array of objects and print it.**

**Source Code :**

import java.util.Scanner;

class Employee {

private int empId;

private String name;

private double salary;

public Employee(int empId, String name, double salary) {

this.empId = empId;

this.name = name;

this.salary = salary;}

public void display() {

System.out.println("Employee ID: " + empId);

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

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

public class employee\_print1{

public static void main(String[] args) {

Scanner sc = new Scanner(System.in);

Employee[] employees = new Employee[5];

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

System.out.println("Enter details for Employee " + (i + 1) + ":");

System.out.print("Employee ID: ");

int empId = sc.nextInt();

sc.nextLine(); // Consume newline character

System.out.print("Name: ");

String name = sc.nextLine();

System.out.print("Salary: ");

double salary = sc.nextDouble();

employees[i] = new Employee(empId, name, salary);

sc.nextLine(); }// Consume newline character

System.out.println("\nEmployee Details:");

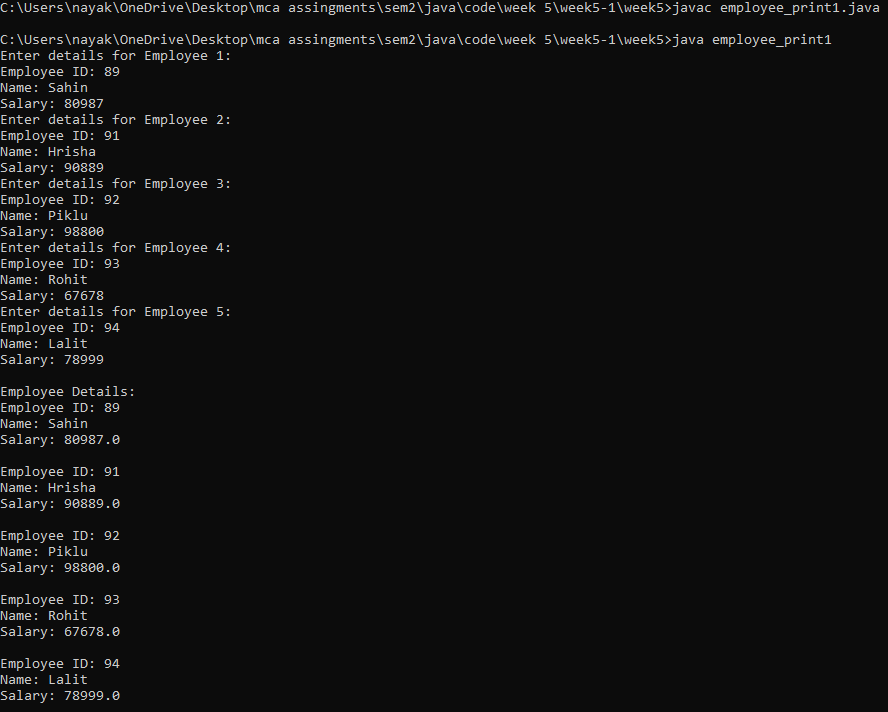
for (Employee employee : employees) {

employee.display();

System.out.println(); }// Add a newline after each employee details

sc.close(); }}

**Output :**

****

**Question 11 : Define a class called circle that contains: • Two private instance variables: radius (of type double) and color**

**(of type String), • Initialize the variables radius and color with default value of 1.0 and "red", respectively**

**using default constructor. • Include a second constructor that will use the default value for color and sets the**

**radius to the value passed as parameter. • Two public methods: getRadius() and getArea() for returning the**

**radius and area of the circle • Invoke the above methods and constructors in the main.**

**Source Code :**

public class q11{

private double radius;

private String color;

public q11(){

this.radius=1.0;

this.color="red";}

public q11(double radius){

this.radius=radius;

this.color="red";}

public double getRadius(){

return this.radius;}

public double getArea(){

return 3.14\*this.radius\*this.radius;}

public static void main(String[] args){

q11 c1=new q11();

q11 c2=new q11(10.4);

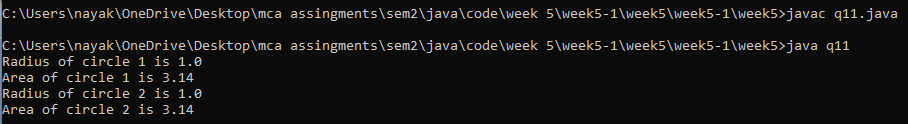
System.out.println("Radius of circle 1 is "+ c1.getRadius());

System.out.println("Area of circle 1 is "+ c1.getArea());

System.out.println("Radius of circle 2 is "+ c1.getRadius());

System.out.println("Area of circle 2 is "+ c1.getArea());}}

**Output :**

****

**Question 12 :** . **Write a program which will accept an integer from the user and pass the value to a method called**

**PrintNumberInWord that will print "ONE", "TWO",... , "NINE", "ZERO" if the integer variable**

**"number" is 1, 2,... , 9, or 0, respectively.**

**Source Code :**

import java.util.Scanner;

public class q12{

public static String printNumberWord(int n){

if(n==0){

return "zero";}

else if(n==1){

return "one";}

else if(n==2){

return "two";}

else if(n==3){

return "three";}

else if(n==4){

return "four";}

else if(n==5){

return "five";}

else if(n==6){

return "six";}

else if(n==7){

return "seven";}

else if(n==8){

return "eight";}

else if(n==9){

return "nine";}

else{

return "Give number between 0 to 9";}}

public static void main(String[] args){

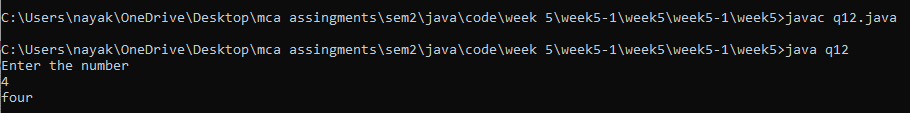
Scanner sc=new Scanner(System.in);

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

int n=sc.nextInt();

System.out.println(printNumberWord(n)); }}

**Output :**

****

**Question 14 : Write a test program that prompts the user to enter the investment amount (e.g., 1000) and the interest**

**rate (e.g., 9%), and print a table that displays future value for the years from 1 to 30, as shown below: The**

**amount invested: 1000 Annual interest rate: 9% Years Future Value 1 1093.8 2 1196.41 ... 29 13467.25 30**

**14730.57**

**Source Code :**

import java.util.Scanner;

public class Investment{

public static void main(String[] args) {

Scanner scanner = new Scanner(System.in);

System.out.print("Enter the investment amount: ");

double investmentAmount = scanner.nextDouble();

System.out.print("Enter the annual interest rate (in percentage): ");

double annualInterestRate = scanner.nextDouble();

annualInterestRate /= 100;

System.out.println("Years\tFuture Value");

for (int years = 1; years <= 30; years++) {

double futureValue = calculateFutureValue(investmentAmount, annualInterestRate, years);

System.out.printf("%d\t%.2f\n", years, futureValue);}

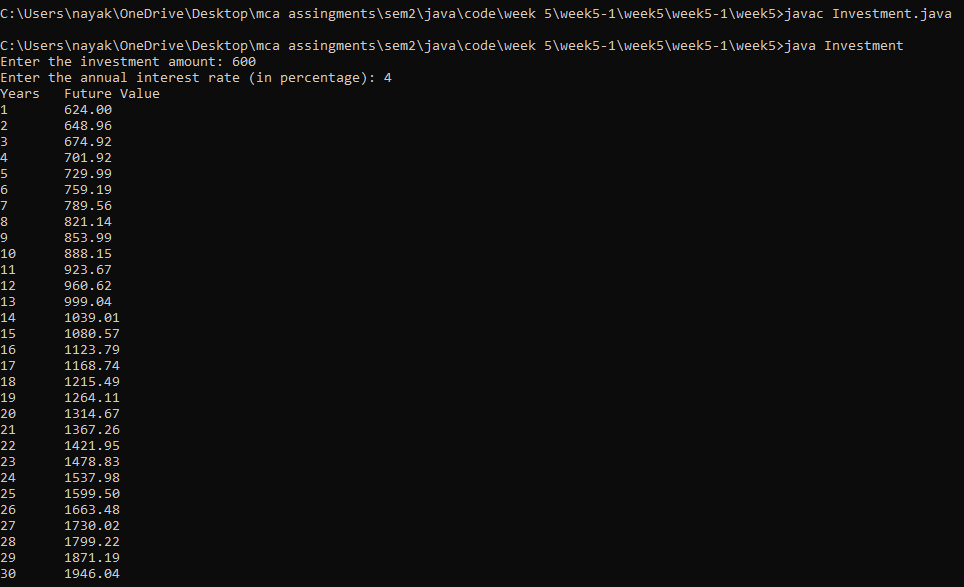
scanner.close();}

private static double calculateFutureValue(double investmentAmount, double annualInterestRate, int years) {

return investmentAmount \* Math.pow(1 + annualInterestRate, years);

}}

**Output :**

****

**Question 15 : Write method headers for the following methods: a. Computing a sales commission, given the sales amount**

**and the commission rate. b. Printing the calendar for a month, given the month and year. c. Computing a**

**square root. d. Testing whether a number is even, and returning true if it is. e. Printing a message a**

**specified number of times. f. Computing the monthly payment, given the loan amount, number of years,**

**and annual interest rate**

**Source Code :**

class Methods{

float commission(float amount, float rate){

return amount \* rate;}

int squareRoot(int num){

return num \* num;}

boolean checkEven(int num){

if(num % 2 == 0)

return true;

else

return false;}

void display(int n, String msg){

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

System.out.println(msg);}}

float emi(float p, float r, float t){

float total = p+(p\*r\*t);

return total/(t\*12); }}

public class q15 {

public static void main(String[] args) {

Methods obj = new Methods();

System.out.println("Commission: " + obj.commission(1000, 0.05f));

System.out.println("Square Root: " + obj.squareRoot(5));

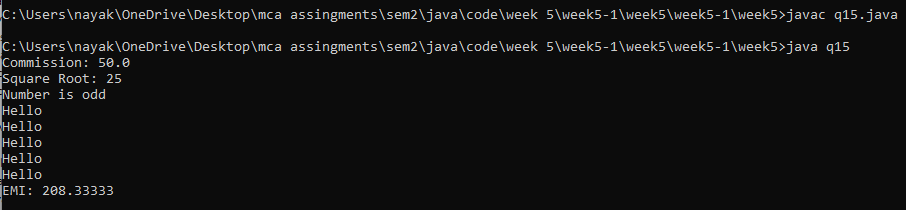
System.out.println(obj.checkEven(97) ? "Number is even" :

"Number is odd");

obj.display(5, "Hello");

System.out.println("EMI: " + obj.emi(10000, 0.05f, 5)); }}

**Output :**

****

**Question 16 : Write a program that reads ten numbers, computes their average, and finds out how many numbers are**

**above the average. [Use this keyword]**

**Source Code :**

import java.util.Scanner;

public class Average {

public static void main(String[] args) {

Scanner scanner = new Scanner(System.in);

double[] numbers = new double[10];

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

for (int i = 0; i < 10; i++) {

System.out.print("Number " + (i + 1) + ": ");

numbers[i] = scanner.nextDouble();}

double sum = 0;

for (double num : numbers) {

sum += num; }

double average = sum / 10;

System.out.println("Average: " + average);

int countAboveAverage = 0;

for (double num : numbers) {

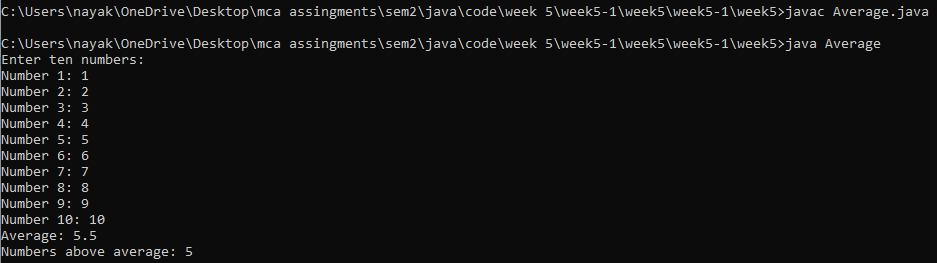
if (num > average) {

countAboveAverage++; }}

System.out.println("Numbers above average: " + countAboveAverage);

scanner.close();}}

**Output :**

****

**Question 17 : Write a program that reads ten integers and displays them in the reverse of the order in which they were**

**read.**

**Source Code :**

import java.util.Scanner;

public class ReverseNumbers {

public static void main(String[] args) {

Scanner scanner = new Scanner(System.in);

int[] numbers = new int[10];

System.out.println("Enter ten integers:");

for (int i = 0; i < 10; i++) {

System.out.print("Enter integer #" + (i + 1) + ": ");

numbers[i] = scanner.nextInt();}

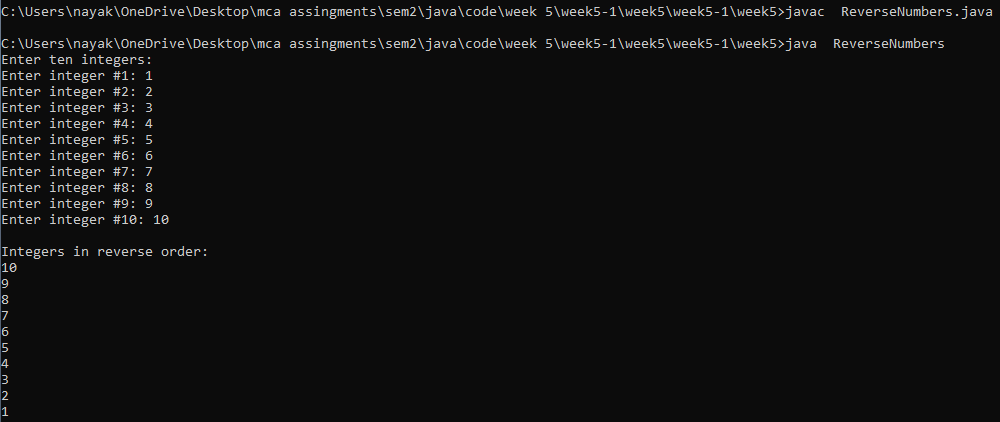
System.out.println("\nIntegers in reverse order:");

for (int i = 9; i >= 0; i--) {

System.out.println(numbers[i]);}

scanner.close(); }}

**Output :**

****

**Question 18 : Write a program to demonstrate use of 'this' keyword.**

**Source Code :**

public class Employee {

private int empid;

private String empname;

private double basic\_salary;

public Employee(int empid, String empname, double basic\_salary) {

this.empid = empid;

this.empname = empname;

this.basic\_salary = basic\_salary;}

public void displayGrossSalary() {

double allowances = 0.2 \* basic\_salary;

double deductions = 0.1 \* basic\_salary;

double gross\_salary = basic\_salary + allowances - deductions;

System.out.println("Employee Details:");

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

System.out.println("Employee Name: " + empname);

System.out.println("Basic Salary: " + basic\_salary);

System.out.println("Allowances: " + allowances);

System.out.println("Deductions: " + deductions);

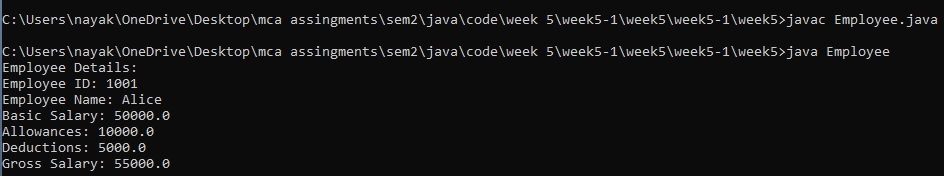
System.out.println("Gross Salary: " + gross\_salary);}

public static void main(String[] args) {

Employee employee1 = new Employee(1001, "Alice", 50000.0);

employee1.displayGrossSalary(); }}

**Output :**

****

**Question 19 : Write a program to demonstrate use of 'static' keyword.**

**Source Code :**

public class NoOfObjects {

public static int count=0;

public NoOfObjects() {

count++;}

public static void main(String[] args) {

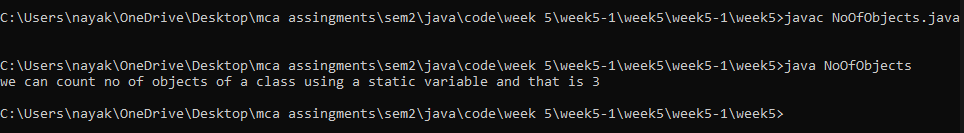
NoOfObjects obj1=new NoOfObjects();

NoOfObjects obj2=new NoOfObjects();

NoOfObjects obj3=new NoOfObjects();

System.out.println("we can count no of objects of a class using a static variable and that is "+ count);}}

**Output :**

****

**Question 20 : Write a program to accept value of apple sales for each day of the week (using array of type float) and then,**

**calculate the average sale of the week.**

**Source Code :**

import java.util.Scanner;

public class AverageAppleSales {

public static void main(String[] args) {

int daysInWeek = 7;

float[] appleSales = new float[daysInWeek];

Scanner scanner = new Scanner(System.in);

for (int day = 0; day < daysInWeek; day++) {

System.out.print("Enter the sales value for day " + (day + 1) + ": ");

appleSales[day] = scanner.nextFloat(); }

float totalSales = 0;

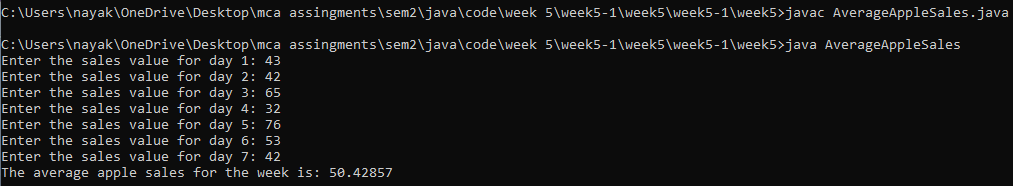
for (float dailySale : appleSales) {

totalSales += dailySale; }

float averageSales = totalSales / daysInWeek;

System.out.println("The average apple sales for the week is: " + averageSales);}}

**Output :**

****

**Question 21 : Write program, which finds the sum of numbers formed by consecutive digits. Input : 2415 output :**

**24+41+15=80.**

**Source Code :**

import java.util.Scanner;

public class Q21 {

public static void main(String[] args) {

Scanner scanner = new Scanner(System.in);

System.out.print("Enter a number: ");

int inputNumber = scanner.nextInt();

int sum = calculateConsecutiveDigitSum(inputNumber);

System.out.println("Output: " + sum);}

private static int calculateConsecutiveDigitSum(int number) {

String strNumber = Integer.toString(number);

int sum = 0;

for (int i = 0; i < strNumber.length() - 1; i++) {

int currentDigit = Character.getNumericValue(strNumber.charAt(i));

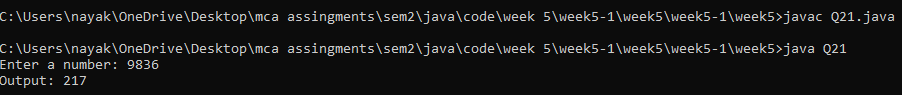
int nextDigit = Character.getNumericValue(strNumber.charAt(i + 1));

int consecutiveNumber = currentDigit \* 10 + nextDigit;

sum += consecutiveNumber;}

return sum; }}

**Output :**

****