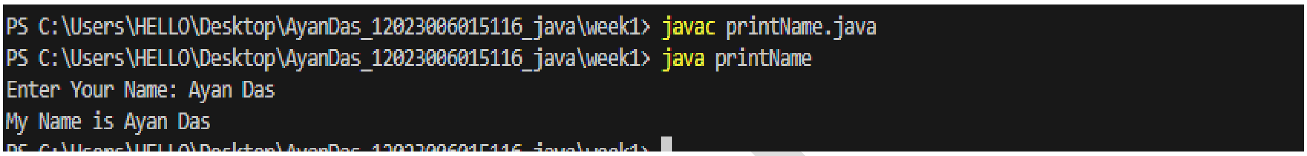
Week 1

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

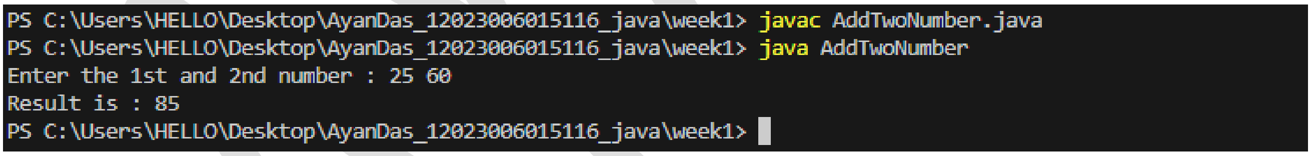
```
import java.util.Scanner;
public class q_1 {
public static void main(String[] args) {
System.out.print("Enter Your Name: ");
Scanner sc = new Scanner(System.in);
String name = sc.nextLine();
sc.close();
System.out.println("My Name is " + name);
}}
```



```
PS C:\Users\HELLO\Desktop\AyanDas_12023006015116_java\week1> javac printName.java
PS C:\Users\HELLO\Desktop\AyanDas_12023006015116_java\week1> java printName
Enter Your Name: Ayan Das
My Name is Ayan Das
PS C:\Users\HELLO\Desktop\AyanDas_12023006015116_java\week1>
```

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

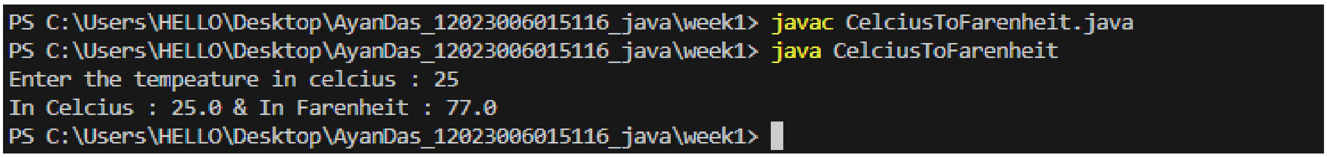
```
import java.util.*;
class AddTwoNumber {
public int add(int n1, int n2) {
int c = n1 + n2;
return c;}
public static void main(String args[]) {
Scanner sc = new Scanner(System.in);
AddTwoNumber add1 = new AddTwoNumber();
System.out.print("Enter the 1st and 2nd number : ");
int n1 = sc.nextInt();
int n2 = sc.nextInt();
int result = add1.add(n1, n2);
System.out.println("Result is : " + result);}}
```



```
PS C:\Users\HELLO\Desktop\AyanDas_12023006015116_java\week1> javac AddTwoNumber.java
PS C:\Users\HELLO\Desktop\AyanDas_12023006015116_java\week1> java AddTwoNumber
Enter the 1st and 2nd number : 25 60
Result is : 85
PS C:\Users\HELLO\Desktop\AyanDas_12023006015116_java\week1>
```

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

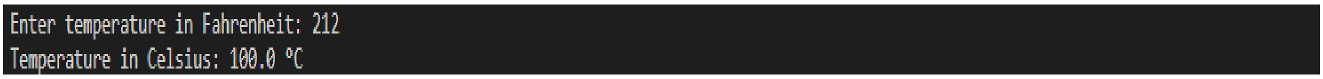
```
import java.util.*;
class CelciusToFarenheit {
public static void main(String args[]) {
Scanner sc = new Scanner(System.in);
System.out.print("Enter the tempeature in celcius : ");
float celcius = sc.nextFloat();
float farenheit = ((9 * celcius) / 5) + 32;
System.out.println("In Celcius : " + celcius + " & In Farenheit : " + farenheit);}}
```



```
PS C:\Users\HELLO\Desktop\AyanDas_12023006015116_java\week1> javac CelciusToFarenheit.java
PS C:\Users\HELLO\Desktop\AyanDas_12023006015116_java\week1> java CelciusToFarenheit
Enter the tempeature in celcius : 25
In Celcius : 25.0 & In Farenheit : 77.0
PS C:\Users\HELLO\Desktop\AyanDas_12023006015116_java\week1>
```

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

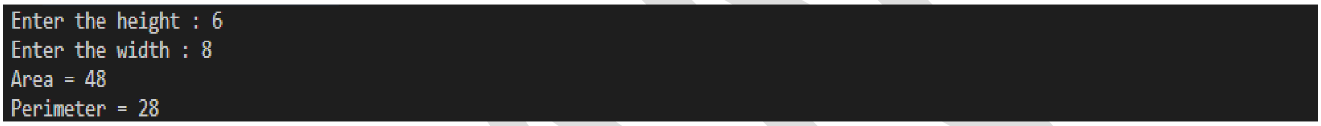
```
import java.util.Scanner;
public class FarenheitToCelcious {
public static void main(String[] args) {
Scanner scanner = new Scanner(System.in);
System.out.print("Enter temperature in Fahrenheit: ");
double fahrenheit = scanner.nextDouble();
double celsius = (fahrenheit - 32) * 5 / 9;
System.out.println("Temperature in Celsius: " + celsius + " °C");
scanner.close();}}
```



```
Enter temperature in Fahrenheit: 212
Temperature in Celsius: 100.0 °C
```

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

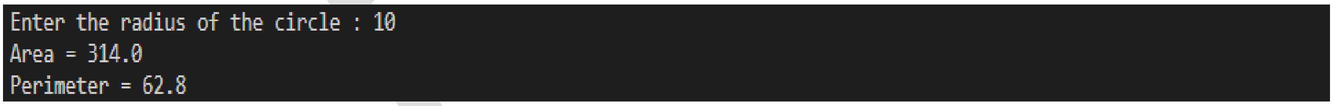
```
import java.util.*;
class Rectangle {
public void area(int height, int width) {
System.out.println("Area = " + (height * width));}
public void perimeter(int height, int width) {
System.out.println("Perimeter = " + (2 * (height + width)));}
public static void main(String[] args) {
Rectangle r1 = new Rectangle();
Scanner sc = new Scanner(System.in);
System.out.print("Enter the height : ");
int height = sc.nextInt();
System.out.print("Enter the width : ");
int width = sc.nextInt();
r1.area(height, width);
r1.perimeter(height, width);}}
```



```
Enter the height : 6
Enter the width : 8
Area = 48
Perimeter = 28
```

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

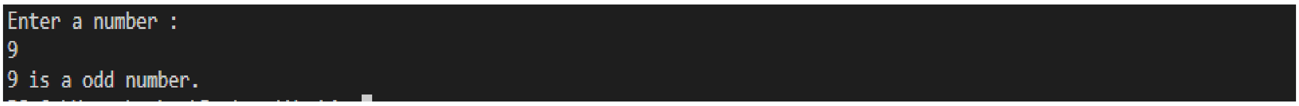
```
import java.util.*;
class Circle {
static final double pi = 3.14;
public void area(double radius) {
System.out.println("Area = " + (pi * radius * radius));}
public void perimeter(double radius) {
System.out.println("Perimeter = " + (float) (2 * pi * radius));}
public static void main(String[] args) {
Circle c1 = new Circle();
Scanner sc = new Scanner(System.in);
System.out.print("Enter the radius of the circle : ");
double radius = sc.nextDouble();
c1.area(radius);
c1.perimeter(radius);}}
```



```
Enter the radius of the circle : 10
Area = 314.0
Perimeter = 62.8
```

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

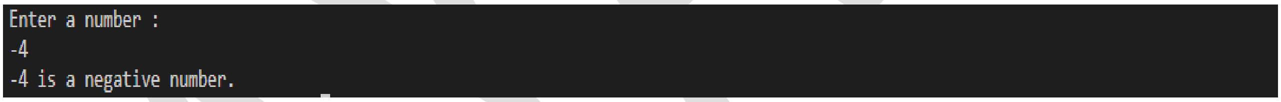
```
import java.util.*;
class Odd_Even {
int number;
public void check(int number) {
if (number > 0) {
if (number % 2 == 0) {
System.out.println(number + " is a even number.");
} else {
System.out.println(number + " is a odd number.");
}
} else {
System.out.println("Please enter a positive number or number should be greater than 0.");
}}
public static void main(String[] args) {
Scanner sc = new Scanner(System.in);
Odd_Even oe = new Odd_Even();
System.out.println("Enter a number : ");
int number = sc.nextInt();
oe.check(number);}}
```



```
Enter a number :
9
9 is a odd number.
```

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

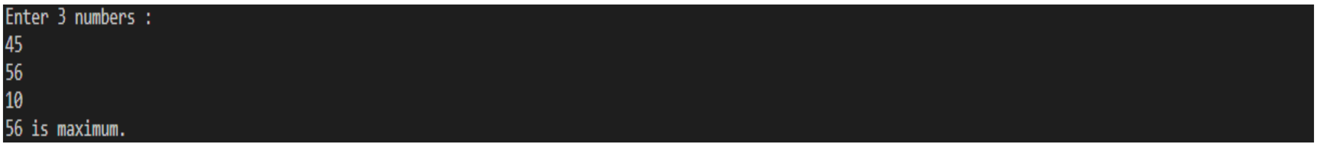
```
import java.util.*;
class Positive_Negative {
int number;
public void check(int number) {
if (number > 0) {
System.out.println(number + " is a positive number.");
} else if (number == 0) {
System.out.println(number + " is neither positive nor negative.");
} else {
System.out.println(number + " is a negative number.");
}}
public static void main(String[] args) {
Scanner sc = new Scanner(System.in);
Positive_Negative pn = new Positive_Negative();
System.out.println("Enter a number : ");
int number = sc.nextInt();
pn.check(number);}}
```



```
Enter a number :
-4
-4 is a negative number.
```

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

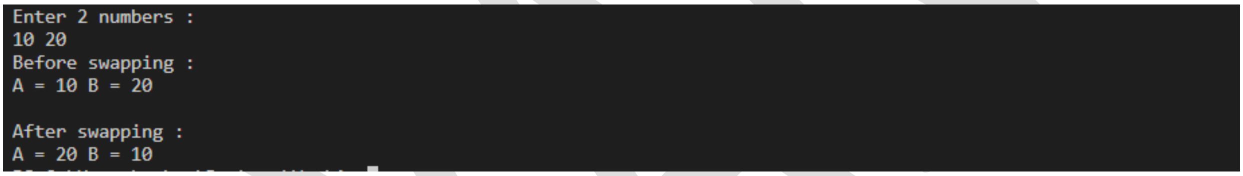
```
import java.util.*;
class FindMaximumBetween3 {
int number;
public void findMaximum(int n1, int n2, int n3) {
if ((n1 > n2) && (n1 > n3)) {
System.out.println(n1 + " is maximum.");
} else if ((n2 > n3) && (n2 > n3)) {
System.out.println(n2 + " is maximum.");
} else {
System.out.println(n3 + " is maximum.");
}}
public static void main(String[] args) {
Scanner sc = new Scanner(System.in);
FindMaximumBetween3 find1 = new FindMaximumBetween3();
System.out.println("Enter 3 numbers : ");
int n1 = sc.nextInt();
int n2 = sc.nextInt();
int n3 = sc.nextInt();
find1.findMaximum(n1, n2, n3);}}
```



```
Enter 3 numbers :
45
56
10
56 is maximum.
```

10. Write a Java program to swap two numbers.

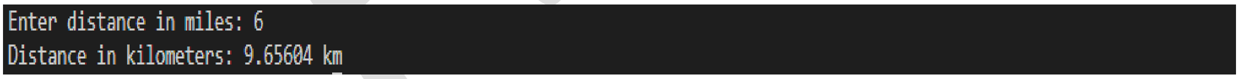
```
import java.util.*;
class Swapping {
int number;
public void swap(int n1, int n2) {
System.out.println("Before swapping : \nA = " + n1 + " B = " + n2);
int temp = n1;
n1 = n2;
n2 = temp;
System.out.println("\nAfter swapping : \nA = " + n1 + " B = " + n2);}
public static void main(String[] args) {
Scanner sc = new Scanner(System.in);
Swapping s1 = new Swapping();
System.out.println("Enter 2 numbers : ");
int n1 = sc.nextInt();
int n2 = sc.nextInt();
s1.swap(n1, n2);}}
```



```
Enter 2 numbers :
10 20
Before swapping :
A = 10 B = 20

After swapping :
A = 20 B = 10
```

11. Write a Java program to convert miles to kilometers.

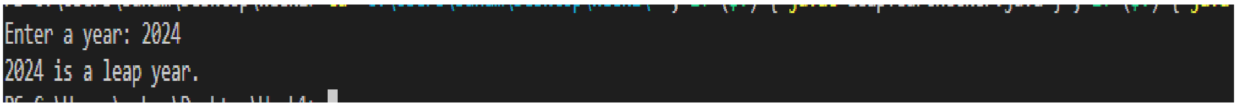
```
import java.util.Scanner;
public class MilesToKilometer {
public static void main(String[] args) {
Scanner scanner = new Scanner(System.in);
System.out.print("Enter distance in miles: ");
double miles = scanner.nextDouble();
double kilometers = miles * 1.60934;
System.out.println("Distance in kilometers: " + kilometers + " km");
scanner.close();}}
```



```
Enter distance in miles: 6
Distance in kilometers: 9.65604 km
```

12. Write a Java program to check whether a year is leapyear or not.

```
import java.util.Scanner;
public class LeapYearChecker {
public static void main(String[] args) {
Scanner scanner = new Scanner(System.in);
System.out.print("Enter a year: ");
int year = scanner.nextInt();
boolean isLeapYear = false;
if ((year % 4 == 0 && year % 100 != 0) || (year % 400 == 0)) {
isLeapYear = true;}
if (isLeapYear) {
System.out.println(year + " is a leap year.");
} else {
System.out.println(year + " is not a leap year.");
}}}
```

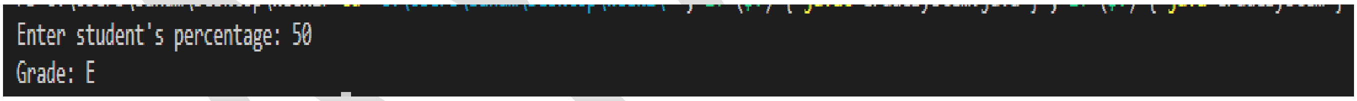


```
Enter a year: 2024
2024 is a leap year.
```

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.

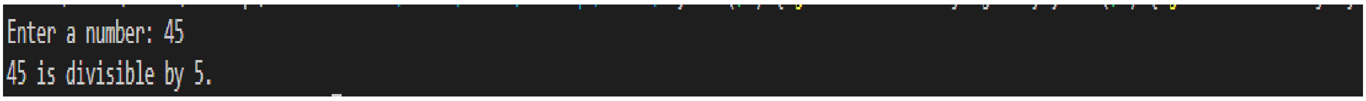
```
import java.util.Scanner;
public class GradeSystem {
public static void main(String[] args) {
Scanner scanner = new Scanner(System.in);
System.out.print("Enter student's percentage: ");
double percentage = scanner.nextDouble();
char grade;
if (percentage >= 90) {
grade = 'A';
} else if (percentage >= 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);
}}
```



```
Enter student's percentage: 50
Grade: E
```

14. Write a Java program to check whether a number is divisible by 5 or not.

```
import java.util.Scanner;
public class DivisibleBy5 {
public static void main(String[] args) {
Scanner scanner = new Scanner(System.in);
System.out.print("Enter a number: ");
int number = scanner.nextInt();
if (number % 5 == 0) {
System.out.println(number + " is divisible by 5.");
} else {
System.out.println(number + " is not divisible by 5.");
}
}}
```

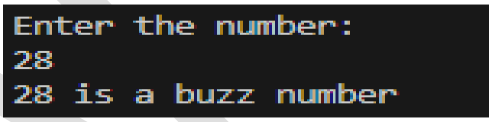


```
Enter a number: 45
45 is divisible by 5.
```

Week 2

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

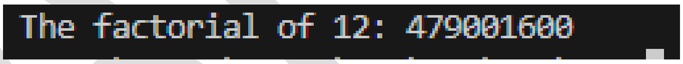
```
import java.util.Scanner;

public class BuzzNumber {

public static void main(String[] args) {

int number;

Scanner sc=new Scanner(System.in);

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

number=sc.nextInt();

if (number%10==7 || number%7==0) {

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

else{

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

}

sc.close();}

}
```



```
Enter the number:
28
28 is a buzz number
```

2. Write a Java program to calculate factorial of 12.

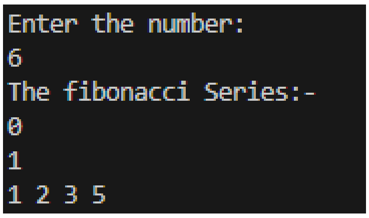
```
public class FactorialofTwelve {
public static void main(String[] args) {
int fact=1;
int num=12;
for(int i=1;i<=num;i++){
fact=fact*i;
}
System.out.println("The factorial of 12: "+fact);
}
}
```



```
The factorial of 12: 479001600
```

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

```
import java.util.Scanner;
public class FibonacciSeri {
public static void main(String[] args) {
int num;
int n1=0,n2=1,n3;
Scanner fb=new Scanner(System.in);
System.out.println("Enter the number: ");
num=fb.nextInt();
System.out.println("The fibonacci Series:-");
System.out.println(n1+"\n"+n2);
for(int i=2;i<num;i++){           n3=n1+n2;
System.err.println(n3);
n1=n2;
n2=n3;
}
fb.close();
}
}
```

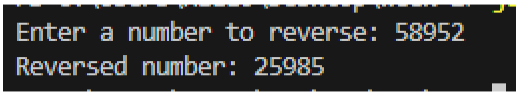


```
Enter the number:
6
The fibonacci Series:-
0
1
1 2 3 5
```

4. **Write a Java program to reverse a number.**

```
import java.util.Scanner;
public class reverseNum {
public static void main(String[] args) {
Scanner scanner = new Scanner(System.in);
System.out.print("Enter a number to reverse: ");
int number = scanner.nextInt();

int reversedNumber = reverseNumber(number);
System.out.println("Reversed number: " + reversedNumber);
scanner.close();
}
public static int reverseNumber(int number) {
int reversedNumber = 0;
while (number != 0) {
int digit = number % 10;
reversedNumber = reversedNumber * 10 + digit;
number /= 10;
}
return reversedNumber;
}
```

```
}
```



```
Enter a number to reverse: 58952
Reversed number: 25985
```

5. Admission to a professional course is subject to the following conditions:

(a) marks in Mathematics >= 60 (b) marks in Physics >=50

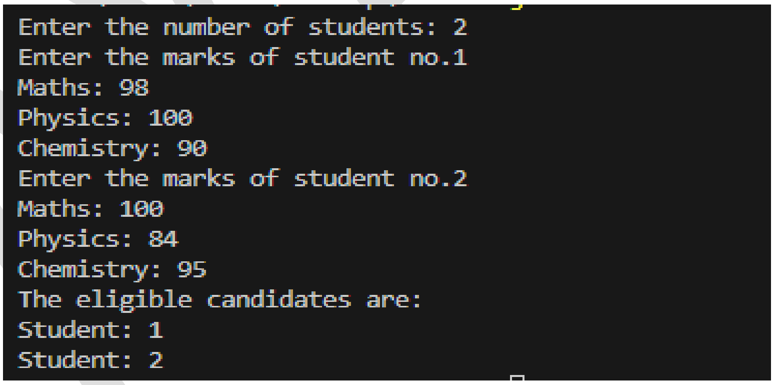
(c) marks in Chemistry >=40 (d) Total in all 3 subjects >=200

(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.

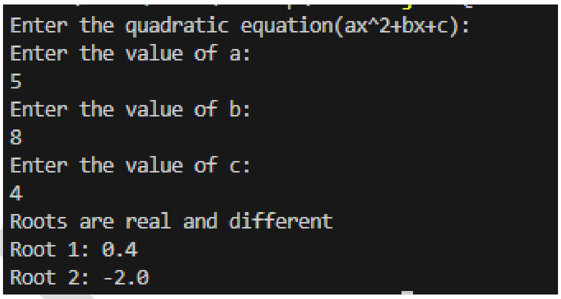
```
import java.util.Scanner;

public class Admission{
static boolean isEligible(int[] marks){
int total=marks[0]+marks[1]+marks[2];
return (marks[0]>=60 && marks[1]>=50 &&
marks[2]>=40 &&
total>=200)||(marks[0]+marks[1]>=150);
}
public static void main(String[] args) {
Scanner sc = new Scanner(System.in);

System.out.print("Enter the number of students:
");
int n = sc.nextInt();
int[][] marks=new int[n][3];
for(int i=0;i<n;i++){
System.out.println("Enter the marks of student
no."+(i+1));
System.out.print("Maths: ");
marks[i][0]=sc.nextInt();
System.out.print("Physics: ");
marks[i][1]=sc.nextInt();
System.out.print("Chemistry: ");
marks[i][2]=sc.nextInt();
}
System.out.println("The eligible candidates are: ");
for(int i=0;i<n;i++){
if (isEligible(marks[i])){
System.out.println("Student: "+(i+1));
}
}
sc.close();   }}
```



```
Enter the number of students: 2
Enter the marks of student no.1
Maths: 98
Physics: 100
Chemistry: 90
Enter the marks of student no.2
Maths: 100
Physics: 84
Chemistry: 95
The eligible candidates are:
Student: 1
Student: 2
```

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

```
import java.util.*;
//find all roots of an quadratic equation
public class QadraticEquationRoot {
public static void main(String[] args) {
Scanner qu=new Scanner(System.in);
System.out.println("Enter the quadratic
equation(ax^2+bx+c): ");
System.out.println("Enter the value of a:");
double a=qu.nextDouble();
System.out.println("Enter the value of b:");
double b=qu.nextDouble();
System.out.println("Enter the value of c:");
double c=qu.nextDouble();
double discriminent=b*b+4*a*c;

if(discriminent>0){
double root1=(-
b+Math.sqrt(discriminent))/(2*a);
double root2=(-b-
Math.sqrt(discriminent))/(2*a);
System.out.println("Roots are real and
different");
System.out.println("Root 1: "+root1);
System.out.println("Root 2: "+root2);
}
else if(discriminent==0){
double root=-b/(2*a);
System.out.println("Roots are real and same");
System.out.println("Root: "+root);
}
else{
double realPart= b/(2*a);
double imaginaryPart=Math.sqrt(-discriminent)/(2*a);
System.out.println("Roots are complex and different");
System.err.println("Root 1:"+realPart+"+"+imaginaryPart+"i");
System.err.println("Root 2:"+realPart+"-"+imaginaryPart+"i");
}
qu.close();
}
}
```

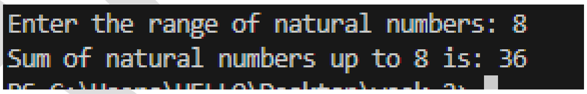


```
Enter the quadratic equation(ax^2+bx+c):
Enter the value of a:
5
Enter the value of b:
8
Enter the value of c:
4
Roots are real and different
Root 1: 0.4
Root 2: -2.0
```

7. Write a Java program to calculate the sum of natural numbers up to a certain range.

Source Code:

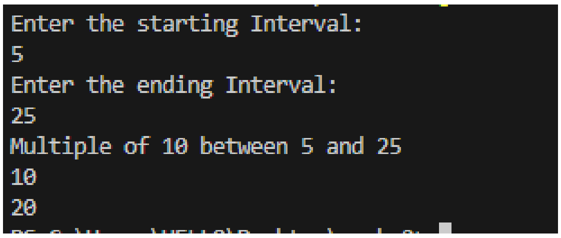
```
import java.util.Scanner;

public class naturalNumSum {
public static void main(String[] args) {
Scanner scanner = new Scanner(System.in);
System.out.print("Enter the range of natural numbers: ");
int range = scanner.nextInt();
int sum = calculateSum(range);
System.out.println("Sum of natural numbers up to " + range + " is: " + sum);
scanner.close(); }
public static int calculateSum(int range) {
int sum = 0;
for (int i = 1; i <= range; i++) {
sum += i;
}
return sum;
}}
```



```
Enter the range of natural numbers: 8
Sum of natural numbers up to 8 is: 36
```

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

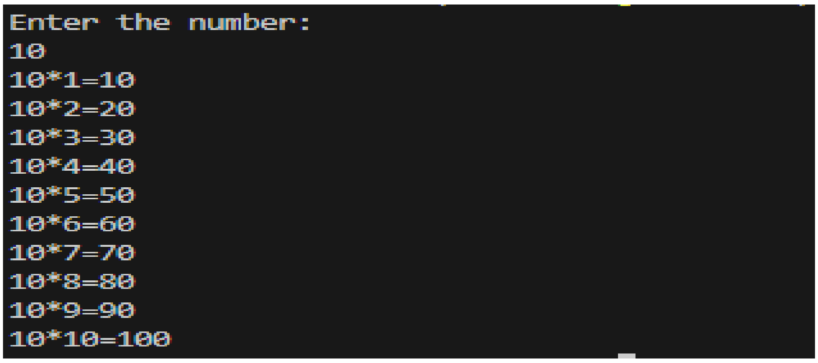
```
import java.util.Scanner;
//multiple of 10 between a given interval
public class MultipleOfTenInterval {
public static void main(String[] args) {
Scanner mul=new Scanner(System.in);
System.out.println("Enter the starting Interval: ");
int start=mul.nextInt();
System.out.println("Enter the ending Interval: ");
int end=mul.nextInt();

System.out.println("Multiple of 10 between "+start+" and "+end);
int firstMultiple= start%10==0 ? start:(start/10+1)*10;

for(int i=firstMultiple;i<=end;i+=10){
System.out.println(i);
}
mul.close();
}
}
```



```
Enter the starting Interval:
5
Enter the ending Interval:
25
Multiple of 10 between 5 and 25
10
20
```

9. Write a Java program to generate multiplication table.

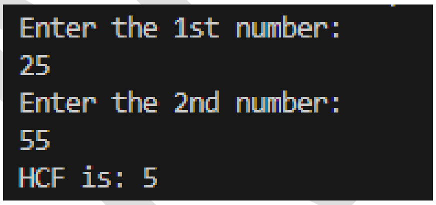
```
import java.util.Scanner;
public class MultiplicationTable{
public static void main(String[] args) {
Scanner ml=new Scanner(System.in);
int num;
System.out.println("Enter the number: ");
num=ml.nextInt();
for(int i=1;i<=num;i++){
System.out.println(num+"*"+i+"="+num*i);
}
ml.close();
} }
```



```
Enter the number:
10
10*1=10
10*2=20
10*3=30
10*4=40
10*5=50
10*6=60
10*7=70
10*8=80
10*9=90
10*10=100
```

10. Write a Java program to find HCF of two Numbers.

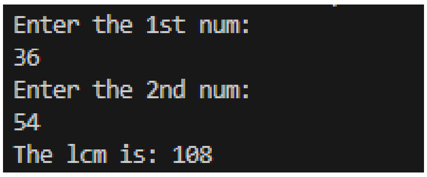
```
import java.util.Scanner;
public class HCFofTwoNum {
public static void main(String[] args) {
Scanner hc=new Scanner(System.in);
int hcf=0;
System.out.println("Enter the 1st number: ");
int num1=hc.nextInt();
System.out.println("Enter the 2nd number:");
int num2=hc.nextInt();
for(int i=1;i<=num1 || i<=num2;i++){

if(num1%i==0 && num2%i==0){
hcf=i;
}
}
System.out.println("HCF is: "+hcf);
hc.close();
}
}
```



```
Enter the 1st number:
25
Enter the 2nd number:
55
HCF is: 5
```

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

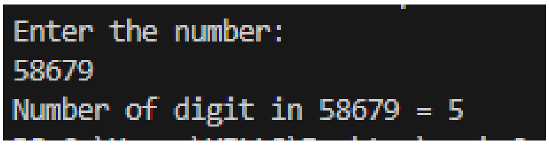
```
import java.util.Scanner;
public class LCMofTwoNum {
public static void main(String[] args) {
Scanner lc=new Scanner(System.in);
int lcm=0;
int hcf=0;
System.out.println("Enter the 1st num: ");
int num1=lc.nextInt();
System.out.println("Enter the 2nd num: ");
int num2=lc.nextInt();
for(int i=1;i<=num1||i<num2;i++){
if(num1%i==0 && num2%i==0){
hcf=i;
}
}
lcm=(num1*num2)/hcf;
System.out.println("The lcm is: "+lcm);
lc.close();
}
}
```



```
Enter the 1st num:
36
Enter the 2nd num:
54
The lcm is: 108
```

12. Write a Java program to count the number of digits of an integer.

```
import java.util.Scanner;
public class NumberOfDigit {
public static void main(String[] args) {
Scanner sc=new Scanner(System.in);
int count=0;
System.out.println("Enter the number: ");
int num=sc.nextInt();
int temp=num;
while (temp>0) {
temp=temp/10;
count++;
}
System.out.println("Number of digit in
"+num+" = "+count);
sc.close();
}
}
```

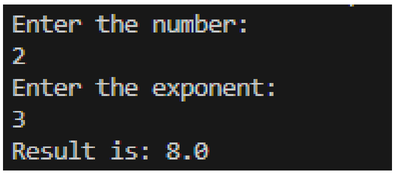


```
Enter the number:
58679
Number of digit in 58679 = 5
```

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

```
import java.util.Scanner;
import java.lang.Math;
```

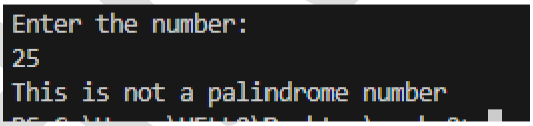
```
public class ExponentialOfNumber {
public static void main(String[] args) {
Scanner ex=new Scanner(System.in);
System.out.println("Enter the number: ");
double num=ex.nextInt();
System.out.println("Enter the exponent:");
double expo=ex.nextInt();
double result=Math.pow(num, expo);
System.out.println("Result is: "+result);

ex.close();
}}
```



```
Enter the number:
2
Enter the exponent:
3
Result is: 8.0
```

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

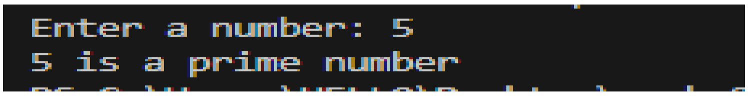
```
import java.util.Scanner;
public class PalindromeNumber {
public static void main(String[] args) {
Scanner pl=new Scanner(System.in);
int rev=0;
System.out.println("Enter the number: ");
int num=pl.nextInt();
int temp=num;
while (num>0) {
int reminder=num%10;
rev=(rev*10)+reminder;
num=num/10;
}
if(rev==temp){
System.out.println("This is a palindrome
number");
}
else{
System.out.println("This is not a palindrome
number");
}
pl.close();
}}
```



```
Enter the number:
25
This is not a palindrome number
```

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

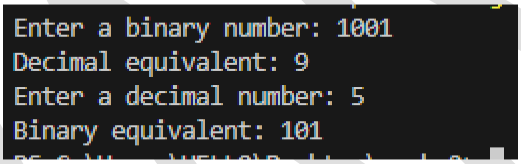
```
import java.util.Scanner;
public class PrimeChecker {
public static void main(String[] args) {
Scanner scanner = new Scanner(System.in);
System.out.print("Enter a number: ");
int number = scanner.nextInt();
if (isPrime(number)) {
System.out.println(number + " is a prime
number");
} else {
System.out.println(number + " is not a prime
number");
}  scanner.close();
}
public static boolean isPrime(int number) {
if (number <= 1) {
return false;
}
for (int i = 2; i <= Math.sqrt(number); i++) {
if (number % i == 0) {
return false;
} }
return true;
}
```



```
Enter a number: 5
5 is a prime number
```

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

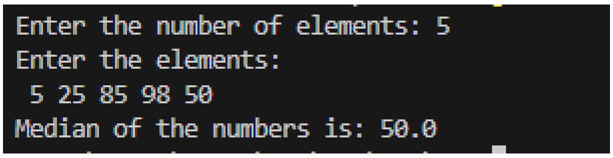
```
import java.util.Scanner;

public class BinaryDecimalConverter {
public static int binaryToDecimal(String
binary) {
int decimal = 0;
int power = 0;
for (int i = binary.length() - 1; i >= 0; i--) {
if (binary.charAt(i) == '1') {
decimal += Math.pow(2, power);
}
power++;
}
return decimal;
}

public static String decimalToBinary(int
decimal) {
StringBuilder binary = new StringBuilder();
if (decimal == 0) {
binary.append(0);
} else {
while (decimal > 0) {
binary.insert(0, decimal % 2);
decimal /= 2;
}   }
return binary.toString();
}
public static void main(String[] args) {
Scanner scanner = new Scanner(System.in);
System.out.print("Enter a binary number: ");
String binaryInput = scanner.nextLine();
int decimalValue =
binaryToDecimal(binaryInput);
System.out.println("Decimal equivalent: " +
decimalValue);

System.out.print("Enter a decimal number: ");
int decimalInput = scanner.nextInt();
String binaryValue =
decimalToBinary(decimalInput);
System.out.println("Binary equivalent: " +
binaryValue);

scanner.close();
}
}
```



```
Enter a binary number: 1001
Decimal equivalent: 9
Enter a decimal number: 5
Binary equivalent: 101
```

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

```
import java.util.Arrays;
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];
System.out.println("Enter the elements:");
for (int i = 0; i < n; i++) {
numbers[i] = scanner.nextInt();
}
double median = findMedian(numbers);
System.out.println("Median of the numbers is:
" + median);
scanner.close();
}
public static double findMedian(int[]
numbers) {
Arrays.sort(numbers);

int length = numbers.length;
if (length % 2 != 0) {
return numbers[length / 2];
} else {
int mid1 = numbers[length / 2 - 1];
int mid2 = numbers[length / 2];
return (double) (mid1 + mid2) / 2;
}
}
}
```

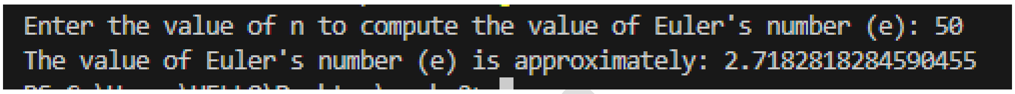


```
Enter the number of elements: 5
Enter the elements:
 5 25 85 98 50
Median of the numbers is: 50.0
```

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!

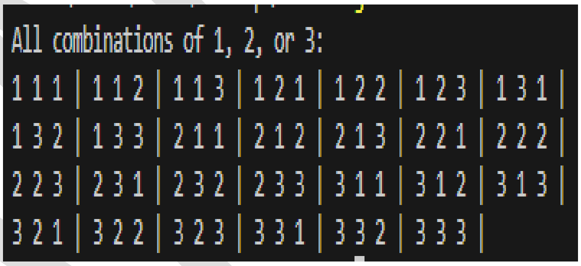
```
import java.util.Scanner;
public class EulerNumber {
public static void main(String[] args) {
Scanner scanner = new Scanner(System.in);
System.out.print("Enter the value of n to
compute the value of Euler's number (e): ");
int n = scanner.nextInt();
double eulerNumber =
computeEulerNumber(n);
System.out.println("The value of Euler's
number (e) is approximately: " +
eulerNumber);
scanner.close();
}
public static double computeEulerNumber(int
n) {
double eulerNumber = 1.0;
double factorial = 1.0;
for (int i = 1; i <= n; i++) {
factorial *= i;
eulerNumber += 1.0 / factorial;
}
return eulerNumber;
}
}
```



```
Enter the value of n to compute the value of Euler's number (e): 50
The value of Euler's number (e) is approximately: 2.7182818284590455
```

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

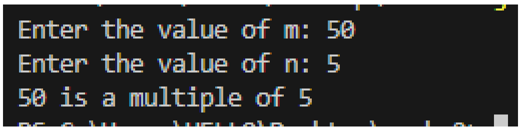
```
public class AllCombination {
public static void main(String[] args) {
generateCombinations();   }
public static void generateCombinations() {
int[] numbers = {1, 2, 3};
System.out.println("All combinations of 1, 2, or
3:");
for (int i = 0; i < numbers.length; i++) {
for (int j = 0; j < numbers.length; j++) {
for (int k = 0; k < numbers.length; k++) {
System.out.print(numbers[i] + " ");
System.out.print(numbers[j] + " ");
System.out.println(numbers[k]);}}}}}
```



```
All combinations of 1, 2, or 3:
1 1 1 | 1 1 2 | 1 1 3 | 1 2 1 | 1 2 2 | 1 2 3 | 1 3 1 |
1 3 2 | 1 3 3 | 2 1 1 | 2 1 2 | 2 1 3 | 2 2 1 | 2 2 2 |
2 2 3 | 2 3 1 | 2 3 2 | 2 3 3 | 3 1 1 | 3 1 2 | 3 1 3 |
3 2 1 | 3 2 2 | 3 2 3 | 3 3 1 | 3 3 2 | 3 3 3 |
```

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

```
import java.util.Scanner;
public class MultipleChecker {
public static void main(String[] args) {
Scanner scanner = new Scanner(System.in);
System.out.print("Enter the value
of m: ");
int m = scanner.nextInt();
System.out.print("Enter the value of n: ");
int n = scanner.nextInt();
if (isMultiple(m, n)) {
System.out.println(m + " is a multiple of " + n);
} else {
System.out.println(m + " is not a multiple of "
+ n);
}
scanner.close();}
public static boolean isMultiple(int m, int n) {
return m % n == 0;
}}
```

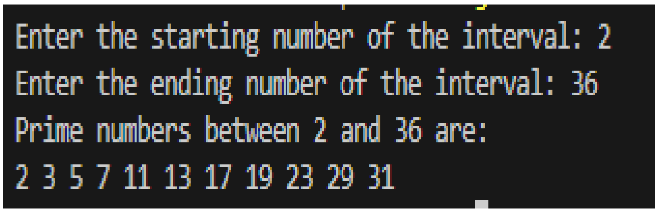


```
Enter the value of m: 50
Enter the value of n: 5
50 is a multiple of 5
```

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

```
import java.util.Scanner;
public class PrimeNumberInInterval {

public static void main(String[] args) {
Scanner scanner = new Scanner(System.in);
System.out.print("Enter the starting number of the
interval: ");
int start = scanner.nextInt();
System.out.print("Enter the ending number of the
interval: ");
int end = scanner.nextInt();
System.out.println("Prime numbers between " +
start + " and " + end + " are:");
displayPrimeNumbers(start, end);
scanner.close();
}
public static boolean isPrime(int num) {
if (num <= 1) {
return false;
}
for (int i = 2; i <= Math.sqrt(num); i++) {
if (num % i == 0) {
```

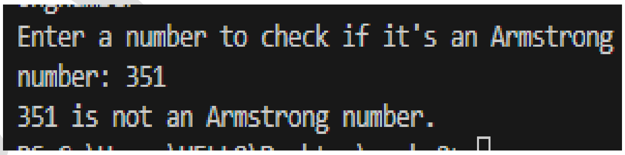
```
return false;
}
}
return true;
}
public static void displayPrimeNumbers(int start, int end) {
        for (int i = start; i <= end; i++) {
        if (isPrime(i)) {
 System.out.print(i + " ");
 } } }}
```



```
Enter the starting number of the interval: 2
Enter the ending number of the interval: 36
Prime numbers between 2 and 36 are:
2 3 5 7 11 13 17 19 23 29 31
```

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

```
import java.util.Scanner;
public class Armstrongnumber {
public static void main(String[] args) {
Scanner scanner = new Scanner(System.in);
System.out.print("Enter a number to check if it's an Armstrong number: ");
int number = scanner.nextInt();
if (isArmstrong(number)) {
System.out.println(number + " is an Armstrong number.");
} else {
System.out.println(number + " is not an Armstrong number.");
}
scanner.close();
}
public static boolean isArmstrong(int number)
{
int originalNumber, remainder, result = 0, n = 0;
originalNumber = number;
while (originalNumber != 0) {
originalNumber /= 10;
++n;
}
originalNumber = number;
while (originalNumber != 0) {
remainder = originalNumber % 10;
result += Math.pow(remainder, n);
originalNumber /= 10;
}
if (result == number) {
return true;
} else {
return false;
}
}
}
```



```
Enter a number to check if it's an Armstrong
number: 351
351 is not an Armstrong number.
```

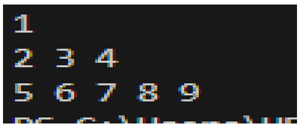
Write Java programs for the patterns given bellow: (23-25)

23. 1

2 3 4

5 6 7 8 9

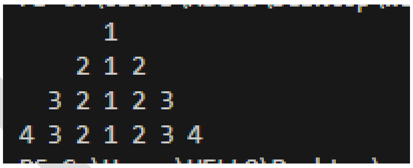
```
public class p1 {
public static void main(String[] args) {
int rows = 3;
int number = 1;
int count=1;
for (int i = 1; i <= rows; i++) {
for (int j = 1; j <= count; j++) {
System.out.print(number + " ");
number++;
}
count = count+2;
System.out.println();
}
}}
```



```
1
2 3 4
5 6 7 8 9
```

24. **1**
2 1 2
3 2 1 2 3
4 3 2 1 2 3 4

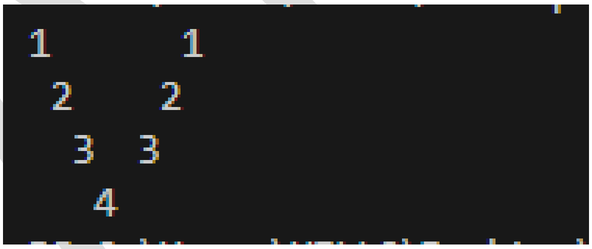
```
public class p2
{
public static void main(String[] args) {
int rows = 4;

for (int i = 1; i <= rows; i++) {
for (int j = 1; j <= rows - i; j++) {
System.out.print("  ");
}
for (int j = i; j >= 2; j--) {
System.out.print(j + " ");
}
for (int j = 1; j <= i; j++) {
System.out.print(j + " ");
}
System.out.println();
}
}
}
```



```
      1
    2 1 2
  3 2 1 2 3
4 3 2 1 2 3 4
```

25. **1 1**
2 2
3 3
4

```
public class p3 {
public static void main(String[] args) {
int rows = 4;
for (int i = 1; i <= rows; i++) {
for (int j = 1; j < i; j++) {
System.out.print(" "); }
System.out.print(i);
for (int j = 1; j <= 2 * (rows - i); j++) {
System.out.print(" ");           }
if (i != rows) {
System.out.print(i);           }
System.out.println(); }    }}
```

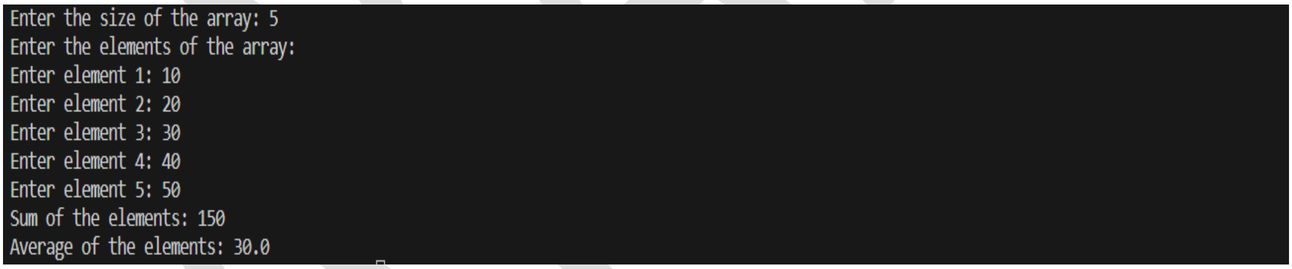


```
1      1
 2    2
  3  3
   4
```

Week 3

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

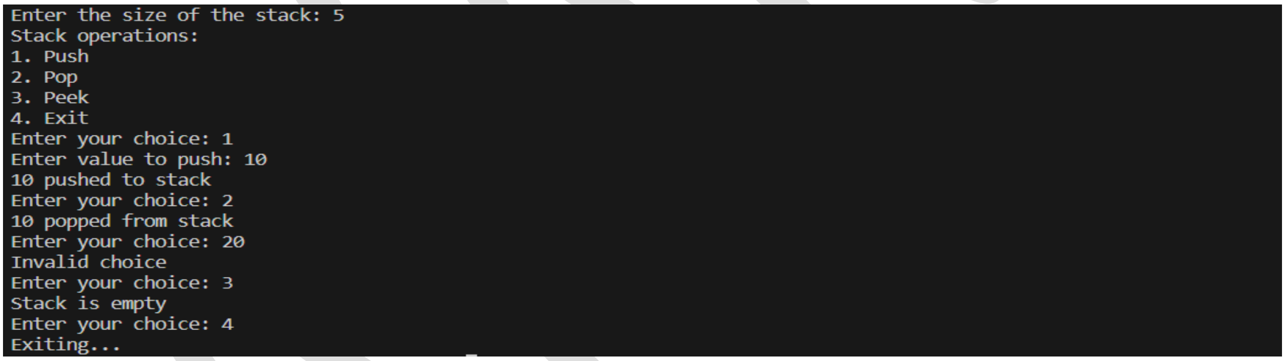
```
import java.util.Scanner;
public class ArraySumAndAverage {
public static void main(String[] args) {
Scanner scanner = new Scanner(System.in);
System.out.print("Enter the size of the array: ");
int size = scanner.nextInt();
int[] numbers = new int[size];
System.out.println("Enter the elements of the array:");
for (int i = 0; i < size; i++) {
System.out.print("Enter element " + (i + 1) + ": ");
numbers[i] = scanner.nextInt();}
int sum = 0;
for (int number : numbers) {
sum += number;
}
double average = (double) sum / size;
System.out.println("Sum of the elements: " + sum);
System.out.println("Average of the elements: " + average);
scanner.close();
}}
```



```
Enter the size of the array: 5
Enter the elements of the array:
Enter element 1: 10
Enter element 2: 20
Enter element 3: 30
Enter element 4: 40
Enter element 5: 50
Sum of the elements: 150
Average of the elements: 30.0
```

2.Write a Java program to implement stack using array.

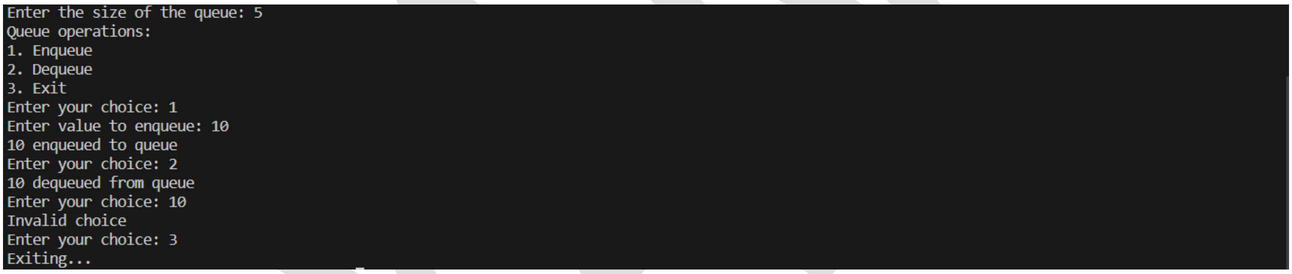
```
import java.util.Scanner;
public class ArrayStack {
private int maxSize;
private int[] stackArray;
private int top;
public ArrayStack(int size) {
maxSize = size;
stackArray = new int[maxSize];
top = -1;
}
public void push(int value) {
if (isFull()) {
System.out.println("Stack is full. Cannot push " +
value);
return;
}
stackArray[++top] = value;
System.out.println(value + " pushed to stack");
}
public int pop() {
if (isEmpty()) {
System.out.println("Stack is empty");
return -1;
}
int value = stackArray[top--];
System.out.println(value + " popped from stack");
return value;
}

public int peek() {
if (isEmpty()) {
System.out.println("Stack is empty");
return -1;
}
return stackArray[top];
}
public boolean isEmpty() {
return (top == -1);
}
public boolean isFull() {
return (top == maxSize - 1);
}
public static void main(String[] args) {
Scanner scanner = new Scanner(System.in);
System.out.print("Enter the size of the stack: ");
int size = scanner.nextInt();
ArrayStack stack = new ArrayStack(size);
System.out.println("Stack operations:");
System.out.println("1. Push");
System.out.println("2. Pop");
System.out.println("3. Peek");
System.out.println("4. Exit");
int choice;
do {
System.out.print("Enter your choice: ");
choice = scanner.nextInt();
switch (choice) {
case 1:
System.out.print("Enter value to push: ");
int value = scanner.nextInt();
stack.push(value);
break;
case 2:
stack.pop();
break;
case 3:
int peekValue = stack.peek();
if (peekValue != -1)
System.out.println("Top element of stack: " +
peekValue);
break;
case 4:
System.out.println("Exiting...");
break;
default:
System.out.println("Invalid choice");
}
} while (choice != 4);
scanner.close();
}}
```



```
Enter the size of the stack: 5
Stack operations:
1. Push
2. Pop
3. Peek
4. Exit
Enter your choice: 1
Enter value to push: 10
10 pushed to stack
Enter your choice: 2
10 popped from stack
Enter your choice: 20
Invalid choice
Enter your choice: 3
Stack is empty
Enter your choice: 4
Exiting...
```

3. Write a Java program to implement Queue using array.

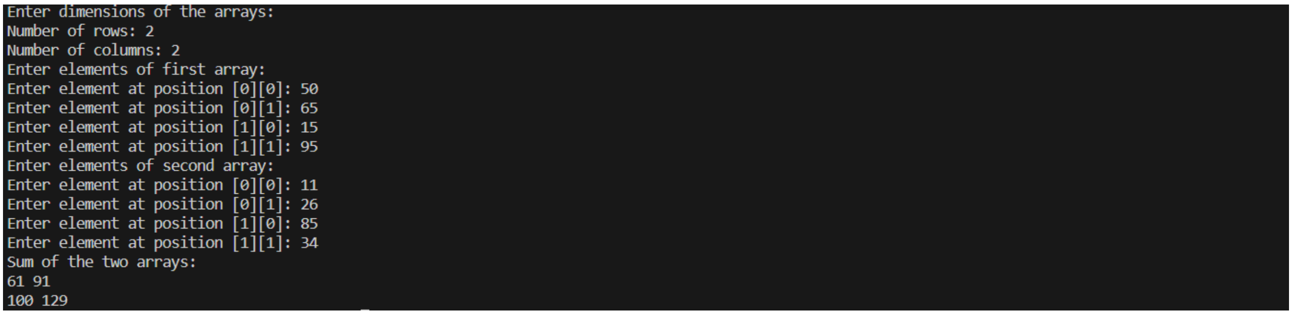
```
import java.util.Scanner;
public class ArrayQueue {
private int maxSize;
private int[] queueArray;
private int front;
private int rear;
private int currentSize;
public ArrayQueue(int size) {
maxSize = size;
queueArray = new int[maxSize];
front = 0;
rear = -1;
currentSize = 0;
}
public void enqueue(int value) {
if (isFull()) {
System.out.println("Queue is full. Cannot enqueue
" + value);
return;
}
rear = (rear + 1) % maxSize;
queueArray[rear] = value;
currentSize++;
System.out.println(value + " enqueued to queue");
}
public int dequeue() {
if (isEmpty()) {
System.out.println("Queue is empty");
return -1;
}
int value = queueArray[front];
front = (front + 1) % maxSize;
currentSize--;
System.out.println(value + " dequeued from
queue");
return value;
}
public boolean isEmpty() {
return (currentSize == 0);
}
public boolean isFull() {
return (currentSize == maxSize);
}
public static void main(String[] args) {
Scanner scanner = new Scanner(System.in);
System.out.print("Enter the size of the queue: ");
int size = scanner.nextInt();
ArrayQueue queue = new ArrayQueue(size);
System.out.println("Queue operations:");
System.out.println("1. Enqueue");
System.out.println("2. Dequeue");
System.out.println("3. Exit");
int choice;
do {
System.out.print("Enter your choice: ");
choice = scanner.nextInt();
switch (choice) {
case 1:
System.out.print("Enter value to enqueue: ");
int value = scanner.nextInt();
queue.enqueue(value);
break;
case 2:
queue.dequeue();
break;
case 3:
System.out.println("Exiting...");
break;
default:
System.out.println("Invalid choice");
}
} while (choice != 3);
scanner.close();
}
}
```



```
Enter the size of the queue: 5
Queue operations:
1. Enqueue
2. Dequeue
3. Exit
Enter your choice: 1
Enter value to enqueue: 10
10 enqueued to queue
Enter your choice: 2
10 dequeued from queue
Enter your choice: 10
Invalid choice
Enter your choice: 3
Exiting...
```

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

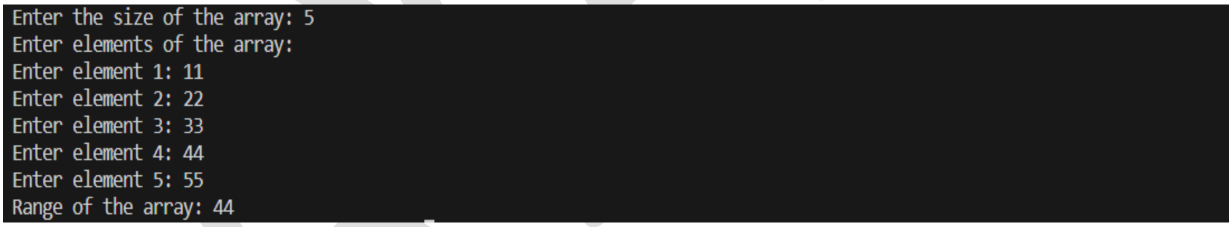
```
import java.util.Scanner;
public class Sum2DArray {
public static void main(String[] args) {
Scanner scanner = new Scanner(System.in);
System.out.println("Enter dimensions of the arrays:");
System.out.print("Number of rows: ");
int rows = scanner.nextInt();
System.out.print("Number of columns: ");
int cols = scanner.nextInt();
int[][] array1 = new int[rows][cols];
int[][] array2 = new int[rows][cols];
System.out.println("Enter elements of first array:");
inputArrayElements(scanner, array1);
System.out.println("Enter elements of second array:");
inputArrayElements(scanner, array2);
int[][] sumArray = new int[rows][cols];
calculateSum(array1, array2, sumArray);
System.out.println("Sum of the two arrays:");
displayArray(sumArray);
scanner.close();
}
public static void inputArrayElements(Scanner scanner, int[][] array) {
for (int i = 0; i < array.length; i++) {
for (int j = 0; j < array[i].length; j++) {
System.out.print("Enter element at position [" + i + "][" + j + "]: ");
array[i][j] = scanner.nextInt();
}
}
}
public static void calculateSum(int[][] array1, int[][] array2, int[][] sumArray) {
for (int i = 0; i < array1.length; i++) {
for (int j = 0; j < array1[i].length; j++) {
sumArray[i][j] = array1[i][j] + array2[i][j];
}
}
}
public static void displayArray(int[][] array) {
for (int i = 0; i < array.length; i++) {
for (int j = 0; j < array[i].length; j++) {
System.out.print(array[i][j] + " ");
}
System.out.println();
}
}
}
```



```
Enter dimensions of the arrays:
Number of rows: 2
Number of columns: 2
Enter elements of first array:
Enter element at position [0][0]: 50
Enter element at position [0][1]: 65
Enter element at position [1][0]: 15
Enter element at position [1][1]: 95
Enter elements of second array:
Enter element at position [0][0]: 11
Enter element at position [0][1]: 26
Enter element at position [1][0]: 85
Enter element at position [1][1]: 34
Sum of the two arrays:
61 91
100 129
```

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

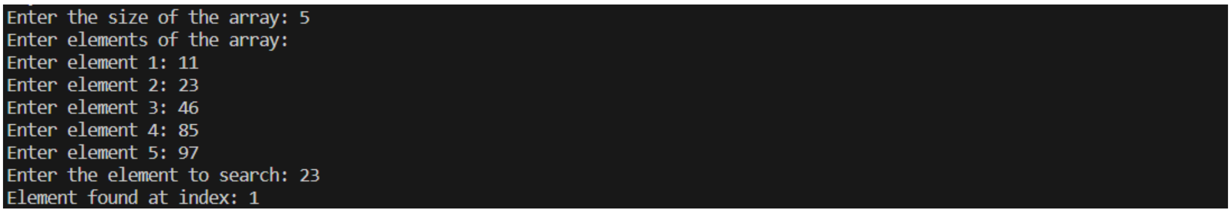
```
import java.util.Scanner;
public class ArrayRange {
public static void main(String[] args) {
Scanner scanner = new Scanner(System.in);
System.out.print("Enter the size of the array: ");
int size = scanner.nextInt();
int[] array = new int[size];
System.out.println("Enter elements of the array:");
for (int i = 0; i < size; i++) {
System.out.print("Enter element " + (i + 1) + ": ");
array[i] = scanner.nextInt();
}
int range = findRange(array);
System.out.println("Range of the array: " + range);
scanner.close();
}
public static int findRange(int[] array) {
int min = array[0];
int max = array[0];
for (int i = 1; i < array.length; i++) {
if (array[i] < min) {
min = array[i];
}
if (array[i] > max) {
max = array[i];
}
}
return max - min;
}
}
```



```
Enter the size of the array: 5
Enter elements of the array:
Enter element 1: 11
Enter element 2: 22
Enter element 3: 33
Enter element 4: 44
Enter element 5: 55
Range of the array: 44
```

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

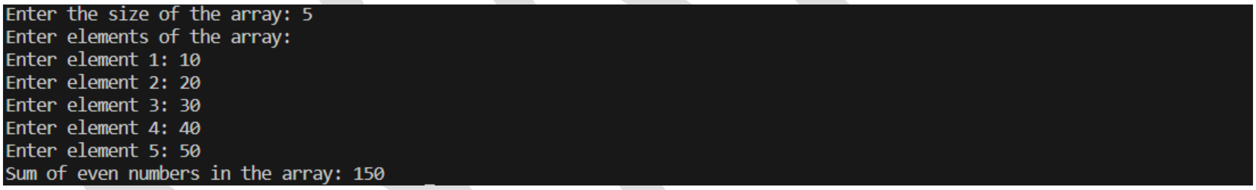
```
import java.util.Scanner;

public class ArraySearch {
public static void main(String[] args) {
Scanner scanner = new Scanner(System.in);
System.out.print("Enter the size of the array: ");
int size = scanner.nextInt();
int[] array = new int[size];
System.out.println("Enter elements of the array:");
for (int i = 0; i < size; i++) {
System.out.print("Enter element " + (i + 1) + ": ");
array[i] = scanner.nextInt();
}
System.out.print("Enter the element to search: ");
int target = scanner.nextInt();
int index = searchElement(array, target);
if (index != -1) {
System.out.println("Element found at index: " + index);
} else {
System.out.println("Element not found in the array.");
}

scanner.close();
}
public static int searchElement(int[] array, int target)
{
for (int i = 0; i < array.length; i++) {
if (array[i] == target) {
return i;
}
}
return -1;
}
}
```



```
Enter the size of the array: 5
Enter elements of the array:
Enter element 1: 11
Enter element 2: 23
Enter element 3: 46
Enter element 4: 85
Enter element 5: 97
Enter the element to search: 23
Element found at index: 1
```

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

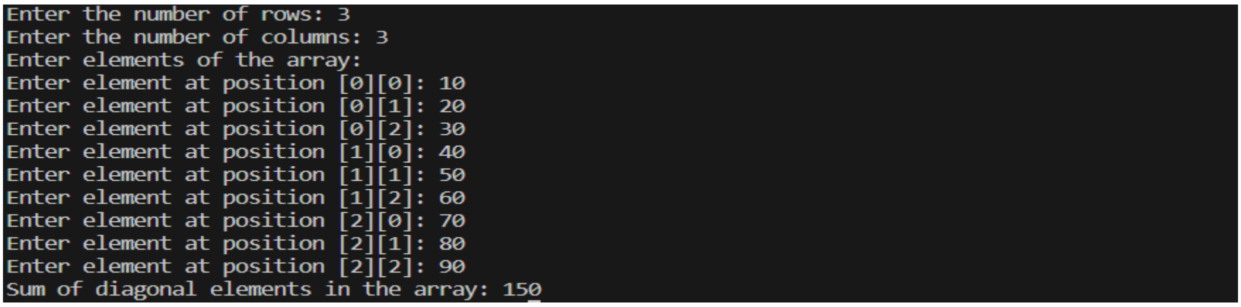
```
import java.util.Scanner;
public class SumOfEvenNumbers {
public static void main(String[] args) {
Scanner scanner = new Scanner(System.in);
System.out.print("Enter the size of the array: ");
int size = scanner.nextInt();
int[] array = new int[size];
System.out.println("Enter elements of the array:");
for (int i = 0; i < size; i++) {
System.out.print("Enter element " + (i + 1) + ": ");
array[i] = scanner.nextInt();
}
int sum = sumOfEvenNumbers(array);
System.out.println("Sum of even numbers in the array: " + sum);
scanner.close();
}
public static int sumOfEvenNumbers(int[] array) {
int sum = 0;
for (int num : array) {
if (num % 2 == 0) {
sum += num;
}
}
return sum;
}
}
```



```
Enter the size of the array: 5
Enter elements of the array:
Enter element 1: 10
Enter element 2: 20
Enter element 3: 30
Enter element 4: 40
Enter element 5: 50
Sum of even numbers in the array: 150
```

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

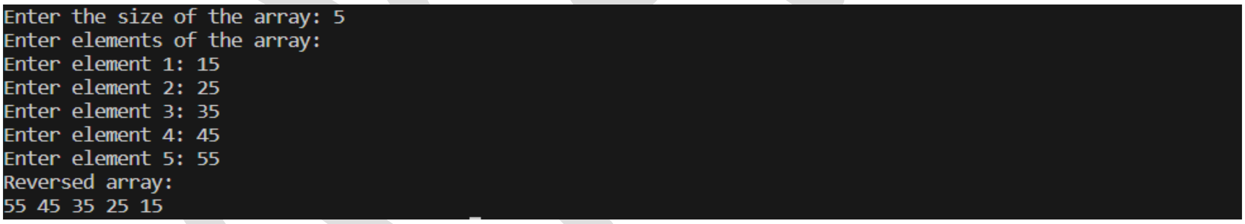
```
import java.util.Scanner;
public class SumOfDiagonalElements {
public static void main(String[] args) {
Scanner scanner = new Scanner(System.in);
System.out.print("Enter the number of rows: ");
int rows = scanner.nextInt();
System.out.print("Enter the number of columns: ");
int cols = scanner.nextInt();
int[][] array = new int[rows][cols];
System.out.println("Enter elements of the array:");
for (int i = 0; i < rows; i++) {
for (int j = 0; j < cols; j++) {
System.out.print("Enter element at position [" + i +
"][" + j + "]: ");
array[i][j] = scanner.nextInt();
}
}
int sum = sumOfDiagonalElements(array);
System.out.println("Sum of diagonal elements in the array: " + sum);
scanner.close();
}
public static int sumOfDiagonalElements(int[][] array) {
int sum = 0;
int rows = array.length;
int cols = array[0].length;
for (int i = 0; i < rows && i < cols; i++) {
sum += array[i][i];
}
return sum;
}
}
```



```
Enter the number of rows: 3
Enter the number of columns: 3
Enter elements of the array:
Enter element at position [0][0]: 10
Enter element at position [0][1]: 20
Enter element at position [0][2]: 30
Enter element at position [1][0]: 40
Enter element at position [1][1]: 50
Enter element at position [1][2]: 60
Enter element at position [2][0]: 70
Enter element at position [2][1]: 80
Enter element at position [2][2]: 90
Sum of diagonal elements in the array: 150
```

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

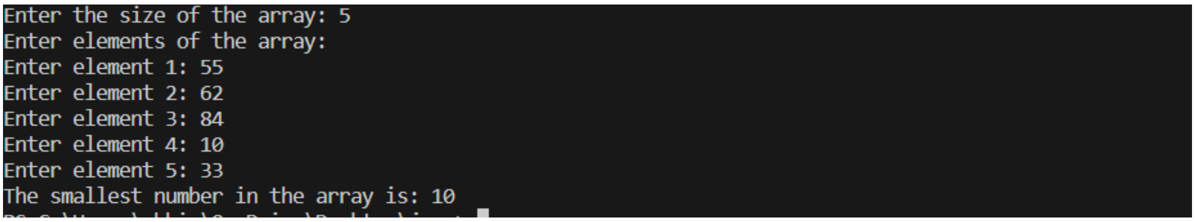
```
import java.util.Scanner;
public class ReverseArray {
public static void main(String[] args) {
Scanner scanner = new Scanner(System.in);
System.out.print("Enter the size of the array: ");
int size = scanner.nextInt();
int[] array = new int[size];
System.out.println("Enter elements of the array:");
for (int i = 0; i < size; i++) {
System.out.print("Enter element " + (i + 1) + ": ");
array[i] = scanner.nextInt();
}
reverseArray(array);
System.out.println("Reversed array:");
for (int num : array) {
System.out.print(num + " ");
}
scanner.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--;
}
}
}
```



```
Enter the size of the array: 5
Enter elements of the array:
Enter element 1: 15
Enter element 2: 25
Enter element 3: 35
Enter element 4: 45
Enter element 5: 55
Reversed array:
55 45 35 25 15
```

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

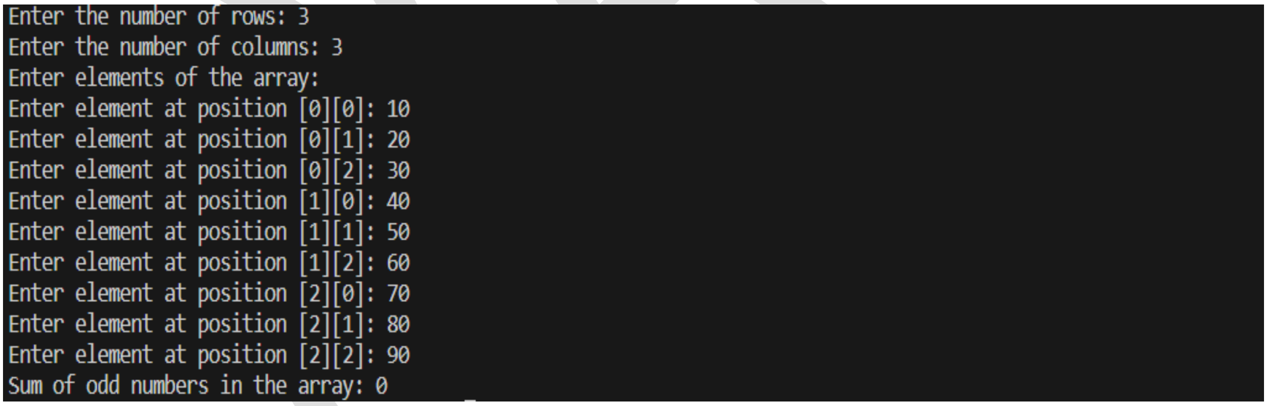
```
import java.util.Scanner;
public class SmallestNumberInArray {
public static void main(String[] args) {
Scanner scanner = new Scanner(System.in);
System.out.print("Enter the size of the array: ");
int size = scanner.nextInt();
int[] array = new int[size];
System.out.println("Enter elements of the array:");
for (int i = 0; i < size; i++) {
System.out.print("Enter element " + (i + 1) + ": ");
array[i] = scanner.nextInt();
}
int smallest = findSmallestNumber(array);
System.out.println("The smallest number in the
array is: " + smallest);
scanner.close();
}
public static int findSmallestNumber(int[] array) {
if (array.length == 0) {
// Handle the case when the array is empty
return Integer.MIN_VALUE;
}
int smallest = array[0];
for (int i = 1; i < array.length; i++) {
if (array[i] < smallest) {
smallest = array[i];
}
}
return smallest;
}
}
```



```
Enter the size of the array: 5
Enter elements of the array:
Enter element 1: 55
Enter element 2: 62
Enter element 3: 84
Enter element 4: 10
Enter element 5: 33
The smallest number in the array is: 10
```

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

```
import java.util.Scanner;
public class SumOfOddNumbers2D {
public static void main(String[] args) {
Scanner scanner = new Scanner(System.in);
System.out.print("Enter the number of rows: ");
int rows = scanner.nextInt();
System.out.print("Enter the number of columns: ");
int cols = scanner.nextInt();
int[][] array = new int[rows][cols];
System.out.println("Enter elements of the array:");
for (int i = 0; i < rows; i++) {
for (int j = 0; j < cols; j++) {
System.out.print("Enter element at position [" + i +
"][" + j + "]: ");
array[i][j] = scanner.nextInt();
}
}
int sum = sumOfOddNumbers(array);
System.out.println("Sum of odd numbers in the
array: " + sum);
scanner.close();
}
public static int sumOfOddNumbers(int[][] array) {
int sum = 0;
for (int[] row : array) {
for (int num : row) {
if (num % 2 != 0) {
sum += num;
}
}
}
return sum;
}
}
```

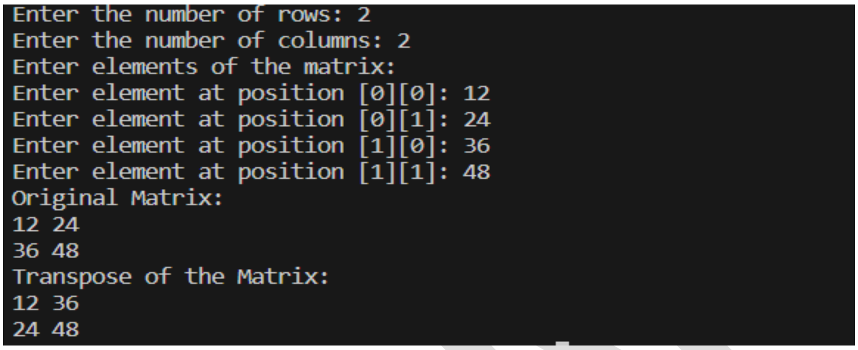


```
Enter the number of rows: 3
Enter the number of columns: 3
Enter elements of the array:
Enter element at position [0][0]: 10
Enter element at position [0][1]: 20
Enter element at position [0][2]: 30
Enter element at position [1][0]: 40
Enter element at position [1][1]: 50
Enter element at position [1][2]: 60
Enter element at position [2][0]: 70
Enter element at position [2][1]: 80
Enter element at position [2][2]: 90
Sum of odd numbers in the array: 0
```

12. Write a Java program to print transpose of matrix.

```
import java.util.Scanner;
public class TransposeOfMatrix {
public static void main(String[] args) {
Scanner scanner = new Scanner(System.in);
System.out.print("Enter the number of rows: ");
int rows = scanner.nextInt();
System.out.print("Enter the number of columns: ");
int cols = scanner.nextInt();
int[][] matrix = new int[rows][cols];
System.out.println("Enter elements of the
matrix:");
for (int i = 0; i < rows; i++) {
for (int j = 0; j < cols; j++) {
System.out.print("Enter element at position [" + i +
"][" + j + "]: ");
matrix[i][j] = scanner.nextInt();
}
}
System.out.println("Original Matrix:");
printMatrix(matrix);
System.out.println("Transpose of the Matrix:");
printTranspose(matrix);
scanner.close();
}
```

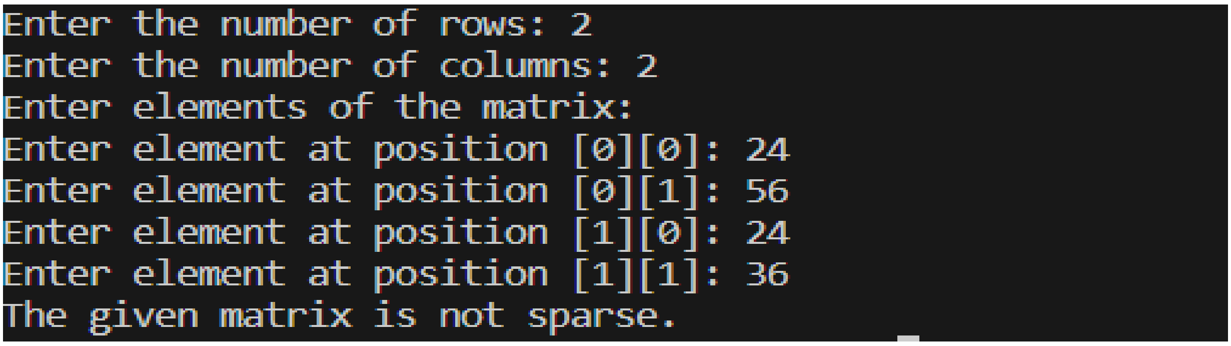
```
public static void printMatrix(int[][] matrix) {
for (int[] row : matrix) {
for (int num : row) {
System.out.print(num + " ");
}
System.out.println();
}
}
public static void printTranspose(int[][] matrix) {
int rows = matrix.length;
int cols = matrix[0].length;
for (int j = 0; j < cols; j++) {
for (int i = 0; i < rows; i++) {
System.out.print(matrix[i][j] + " ");
}
System.out.println();
}
}
```



```
Enter the number of rows: 2
Enter the number of columns: 2
Enter elements of the matrix:
Enter element at position [0][0]: 12
Enter element at position [0][1]: 24
Enter element at position [1][0]: 36
Enter element at position [1][1]: 48
Original Matrix:
12 24
36 48
Transpose of the Matrix:
12 36
24 48
```

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

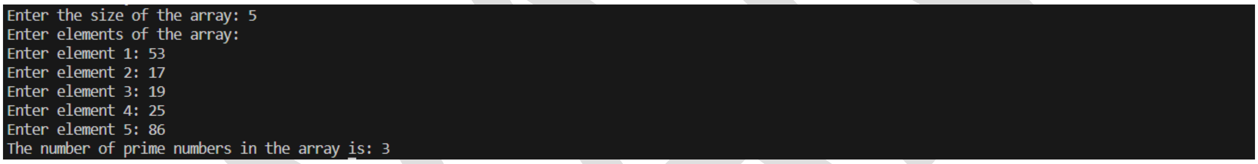
```
import java.util.Scanner;
public class SparseMatrixChecker {
public static void main(String[] args) {
Scanner scanner = new Scanner(System.in);
System.out.print("Enter the number of rows: ");
int rows = scanner.nextInt();
System.out.print("Enter the number of columns: ");
int cols = scanner.nextInt();
int[][] matrix = new int[rows][cols];
System.out.println("Enter elements of the matrix:");
for (int i = 0; i < rows; i++) {
for (int j = 0; j < cols; j++) {
System.out.print("Enter element at position [" + i + "][" + j + "]: ");
matrix[i][j] = scanner.nextInt();
}
}
boolean isSparse = isSparseMatrix(matrix);
if (isSparse) {
System.out.println("The given matrix is sparse.");
} else {
System.out.println("The given matrix is not sparse.");
}
scanner.close();
}
public static boolean isSparseMatrix(int[][] matrix) {
int zeroCount = 0;
int totalElements = matrix.length * matrix[0].length;
for (int[] row : matrix) {
for (int num : row) {
if (num == 0) {
zeroCount++;
}
}
}
return zeroCount > (totalElements / 2);
}
```



```
Enter the number of rows: 2
Enter the number of columns: 2
Enter elements of the matrix:
Enter element at position [0][0]: 24
Enter element at position [0][1]: 56
Enter element at position [1][0]: 24
Enter element at position [1][1]: 36
The given matrix is not sparse.
```

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

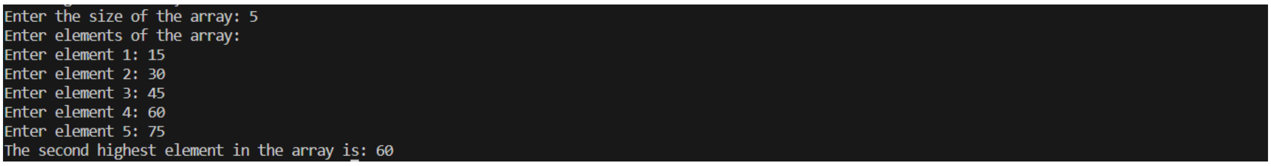
```
import java.util.Scanner;
public class CountPrimeNumbers {
public static void main(String[] args) {
Scanner scanner = new Scanner(System.in);
System.out.print("Enter the size of the array: ");
int size = scanner.nextInt();
int[] array = new int[size];
System.out.println("Enter elements of the array:");
for (int i = 0; i < size; i++) {
System.out.print("Enter element " + (i + 1) + ": ");
array[i] = scanner.nextInt();
}
int primeCount = countPrimeNumbers(array);
System.out.println("The number of prime numbers in the array is: " + primeCount);
scanner.close();
}
public static boolean isPrime(int num) {
if (num <= 1) {
return false;
}
for (int i = 2; i <= Math.sqrt(num); i++) {
if (num % i == 0) {
return false;
}
}
return true;
}
public static int countPrimeNumbers(int[] array) {
int count = 0;
for (int num : array) {
if (isPrime(num)) {
count++;
}
}
return count;
}
}
```



```
Enter the size of the array: 5
Enter elements of the array:
Enter element 1: 53
Enter element 2: 17
Enter element 3: 19
Enter element 4: 25
Enter element 5: 86
The number of prime numbers in the array is: 3
```

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

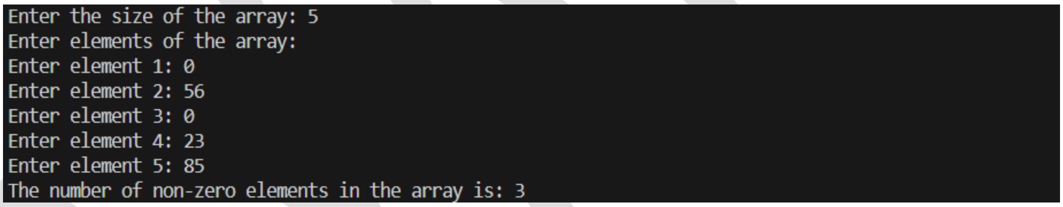
```
import java.util.Scanner;
public class SecondHighestElement {
public static void main(String[] args) {
Scanner scanner = new Scanner(System.in);
System.out.print("Enter the size of the array: ");
int size = scanner.nextInt();
int[] array = new int[size];
System.out.println("Enter elements of the array:");
for (int i = 0; i < size; i++) {
System.out.print("Enter element " + (i + 1) + ": ");
array[i] = scanner.nextInt();
}
int secondHighest = findSecondHighest(array);
if (secondHighest != Integer.MIN_VALUE) {
System.out.println("The second highest element in the array is: " + secondHighest);
} else {
System.out.println("The second highest element does not exist in the array.");
}
scanner.close();
}
public static int findSecondHighest(int[] array) {
if (array.length < 2) {
return Integer.MIN_VALUE;
}
int firstMax = Integer.MIN_VALUE;
int secondMax = Integer.MIN_VALUE;

for (int num : array) {
if (num > firstMax) {
secondMax = firstMax;
firstMax = num;
} else if (num > secondMax && num != firstMax) {
secondMax = num;
}
}
if (secondMax == Integer.MIN_VALUE) {
return Integer.MIN_VALUE;
}
return secondMax;
}
}
```



```
Enter the size of the array: 5
Enter elements of the array:
Enter element 1: 15
Enter element 2: 30
Enter element 3: 45
Enter element 4: 60
Enter element 5: 75
The second highest element in the array is: 60
```

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

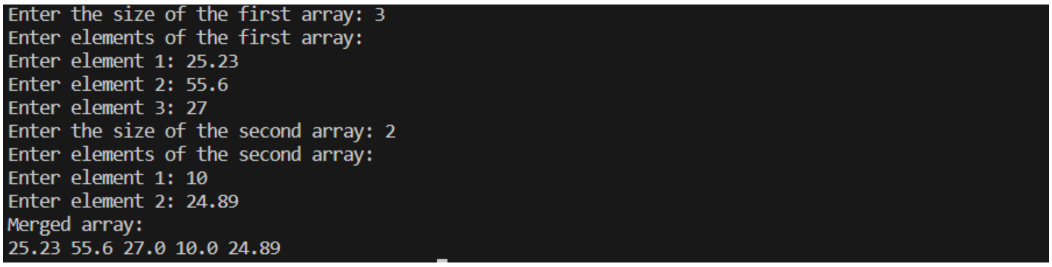
```
import java.util.Scanner;
public class NonZeroElementCounter {
public static void main(String[] args) {
Scanner scanner = new Scanner(System.in);
System.out.print("Enter the size of the array: ");
int size = scanner.nextInt();
int[] array = new int[size];
System.out.println("Enter elements of the array:");
for (int i = 0; i < size; i++) {
System.out.print("Enter element " + (i + 1) + ": ");
array[i] = scanner.nextInt();
}
int nonZeroCount = countNonZeroElements(array);
System.out.println("The number of non-zero elements in the array is: " + nonZeroCount);
scanner.close();
}
public static int countNonZeroElements(int[] array) {
int count = 0;
for (int num : array) {
if (num != 0) {
count++;
}
}
return count;
}
}
```



```
Enter the size of the array: 5
Enter elements of the array:
Enter element 1: 0
Enter element 2: 56
Enter element 3: 0
Enter element 4: 23
Enter element 5: 85
The number of non-zero elements in the array is: 3
```

17. Write a Java program to merge two float arrays.

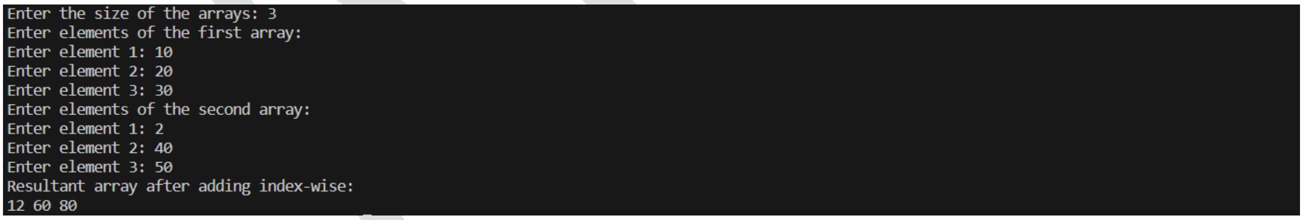
```
import java.util.Scanner;
public class MergeFloatArrays {
public static void main(String[] args) {
Scanner scanner = new Scanner(System.in);
System.out.print("Enter the size of the first array: ");
int size1 = scanner.nextInt();
float[] array1 = new float[size1];
System.out.println("Enter elements of the first array:");
for (int i = 0; i < size1; i++) {
System.out.print("Enter element " + (i + 1) + ": ");
array1[i] = scanner.nextFloat();
}
System.out.print("Enter the size of the second array: ");
int size2 = scanner.nextInt();
float[] array2 = new float[size2];
System.out.println("Enter elements of the second array:");
for (int i = 0; i < size2; i++) {
System.out.print("Enter element " + (i + 1) + ": ");
array2[i] = scanner.nextFloat();
}
float[] mergedArray = mergeArrays(array1, array2);
System.out.println("Merged array:");
for (float num : mergedArray) {
System.out.print(num + " ");
}
scanner.close();
}

public static float[] mergeArrays(float[] array1, float[] array2) {
int size1 = array1.length;
int size2 = array2.length;
float[] mergedArray = new float[size1 + size2];
// Copy elements of the first array
for (int i = 0; i < size1; i++) {
mergedArray[i] = array1[i];
}
for (int i = 0; i < size2; i++) {
mergedArray[size1 + i] = array2[i];
}
return mergedArray;
}
}
```



```
Enter the size of the first array: 3
Enter elements of the first array:
Enter element 1: 25.23
Enter element 2: 55.6
Enter element 3: 27
Enter the size of the second array: 2
Enter elements of the second array:
Enter element 1: 10
Enter element 2: 24.89
Merged array:
25.23 55.6 27.0 10.0 24.89
```

18. Write a Java program where elements of two integer arrays get added index wise and get stored into a third array.

```
import java.util.Scanner;
public class AddArraysIndexWise {
public static void main(String[] args) {
Scanner scanner = new Scanner(System.in);
System.out.print("Enter the size of the arrays: ");
int size = scanner.nextInt();
int[] array1 = new int[size];
int[] array2 = new int[size];
System.out.println("Enter elements of the first array:");
for (int i = 0; i < size; i++) {
System.out.print("Enter element " + (i + 1) + ": ");
array1[i] = scanner.nextInt();
}
System.out.println("Enter elements of the second array:");
for (int i = 0; i < size; i++) {
System.out.print("Enter element " + (i + 1) + ": ");
array2[i] = scanner.nextInt();

}
int[] sumArray = addArraysIndexWise(array1, array2);
System.out.println("Resultant array after adding index-wise:");
for (int num : sumArray) {
System.out.print(num + " ");
}

scanner.close();
}
public static int[] addArraysIndexWise(int[] array1, int[] array2) {
int size = array1.length;
int[] sumArray = new int[size];
for (int i = 0; i < size; i++) {
sumArray[i] = array1[i] + array2[i];
}
return sumArray;
}
}
```

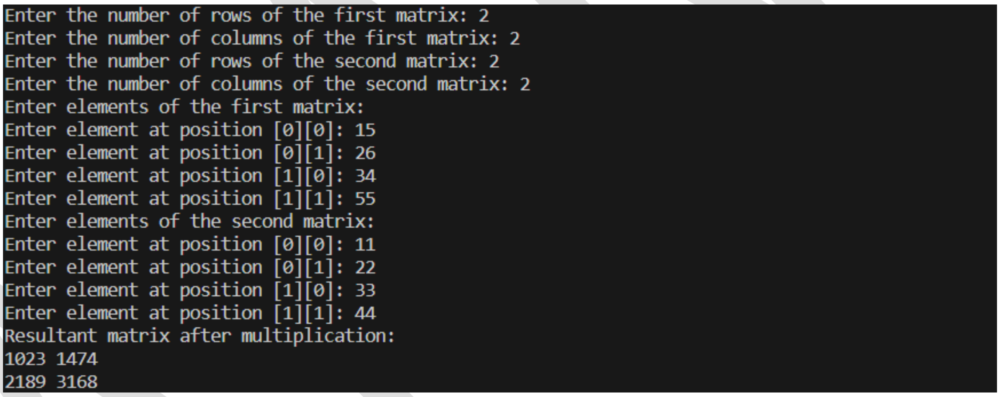


```
Enter the size of the arrays: 3
Enter elements of the first array:
Enter element 1: 10
Enter element 2: 20
Enter element 3: 30
Enter elements of the second array:
Enter element 1: 2
Enter element 2: 40
Enter element 3: 50
Resultant array after adding index-wise:
12 60 80
```

19. Write a Java program to multiply two matrices.

```
import java.util.Scanner;
public class MatrixMultiplication {
public static void main(String[] args) {
Scanner scanner = new Scanner(System.in);
System.out.print("Enter the number of rows of the first matrix: ");
int rows1 = scanner.nextInt();
System.out.print("Enter the number of columns of the first matrix: ");
int cols1 = scanner.nextInt();
System.out.print("Enter the number of rows of the second matrix: ");
int rows2 = scanner.nextInt();
System.out.print("Enter the number of columns of the second matrix: ");
int cols2 = scanner.nextInt();
if (cols1 != rows2) {
System.out.println("Matrix multiplication is not possible.");
scanner.close();
return;
}
int[][] matrix1 = new int[rows1][cols1];
System.out.println("Enter elements of the first matrix:");
for (int i = 0; i < rows1; i++) {
for (int j = 0; j < cols1; j++) {
```

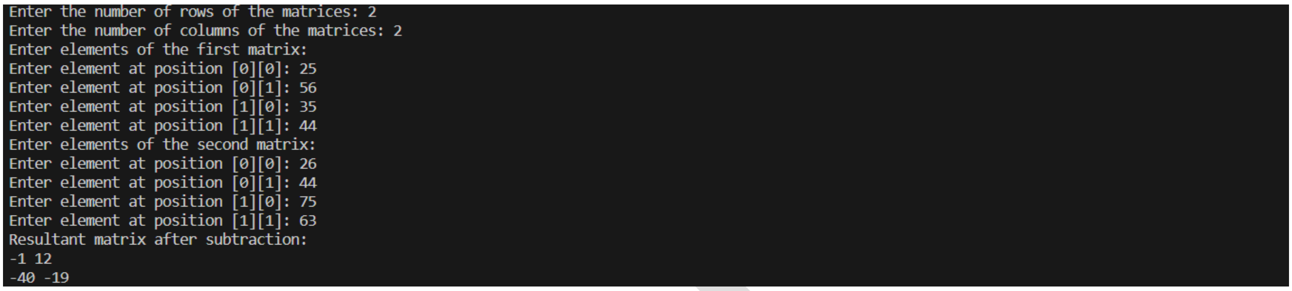
```
System.out.print("Enter element at position [" + i + "][" + j + "]: ");
matrix1[i][j] = scanner.nextInt();
}
}
int[][] matrix2 = new int[rows2][cols2];
System.out.println("Enter elements of the second matrix:");
for (int i = 0; i < rows2; i++) {
for (int j = 0; j < cols2; j++) {
System.out.print("Enter element at position [" + i + "][" + j + "]: ");
matrix2[i][j] = scanner.nextInt();
}
}
int[][] resultMatrix = multiplyMatrices(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(resultMatrix[i][j] + " ");
}
System.out.println();
}
scanner.close();
}
public static int[][] multiplyMatrices(int[][] matrix1, int[][] matrix2) {
int rows1 = matrix1.length;
int cols1 = matrix1[0].length;
int cols2 = matrix2[0].length;
int[][] resultMatrix = 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++) {
resultMatrix[i][j] += matrix1[i][k] * matrix2[k][j];
}
}
}
return resultMatrix;
}
}
```



```
Enter the number of rows of the first matrix: 2
Enter the number of columns of the first matrix: 2
Enter the number of rows of the second matrix: 2
Enter the number of columns of the second matrix: 2
Enter elements of the first matrix:
Enter element at position [0][0]: 15
Enter element at position [0][1]: 26
Enter element at position [1][0]: 34
Enter element at position [1][1]: 55
Enter elements of the second matrix:
Enter element at position [0][0]: 11
Enter element at position [0][1]: 22
Enter element at position [1][0]: 33
Enter element at position [1][1]: 44
Resultant matrix after multiplication:
1023 1474
2189 3168
```

20. Write a Java program to subtract two matrices.

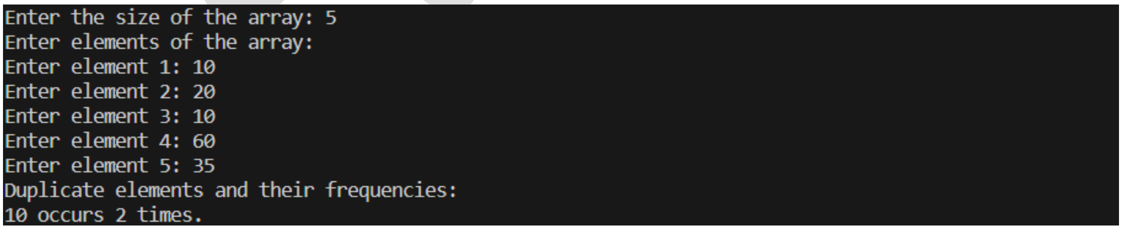
```
import java.util.Scanner;
public class MatrixSubtraction {
public static void main(String[] args) {
Scanner scanner = new Scanner(System.in);
System.out.print("Enter the number of rows of the matrices: ");
int rows = scanner.nextInt();
System.out.print("Enter the number of columns of the matrices: ");
int cols = scanner.nextInt();
int[][] matrix1 = new int[rows][cols];
int[][] matrix2 = new int[rows][cols];
System.out.println("Enter elements of the first matrix:");
for (int i = 0; i < rows; i++) {
for (int j = 0; j < cols; j++) {
System.out.print("Enter element at position [" + i + "][" + j + "]: ");
matrix1[i][j] = scanner.nextInt();
}
}
System.out.println("Enter elements of the second matrix:");
for (int i = 0; i < rows; i++) {
for (int j = 0; j < cols; j++) {
System.out.print("Enter element at position [" + i + "][" + j + "]: ");
matrix2[i][j] = scanner.nextInt();
}
}
int[][] resultMatrix = subtractMatrices(matrix1, matrix2);
System.out.println("Resultant matrix after subtraction:");
for (int i = 0; i < rows; i++) {
for (int j = 0; j < cols; j++) {
System.out.print(resultMatrix[i][j] + " ");
}
System.out.println();
}
scanner.close();
}
public static int[][] subtractMatrices(int[][] matrix1, int[][] matrix2) {
int rows = matrix1.length;
int cols = matrix1[0].length;
int[][] resultMatrix = new int[rows][cols];
for (int i = 0; i < rows; i++) {
for (int j = 0; j < cols; j++) {
resultMatrix[i][j] = matrix1[i][j] - matrix2[i][j];
}
}
return resultMatrix;
}
```



```
Enter the number of rows of the matrices: 2
Enter the number of columns of the matrices: 2
Enter elements of the first matrix:
Enter element at position [0][0]: 25
Enter element at position [0][1]: 56
Enter element at position [1][0]: 35
Enter element at position [1][1]: 44
Enter elements of the second matrix:
Enter element at position [0][0]: 26
Enter element at position [0][1]: 44
Enter element at position [1][0]: 75
Enter element at position [1][1]: 63
Resultant matrix after subtraction:
-1 12
-40 -19
```

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

```
import java.util.HashMap;
import java.util.Map;
import java.util.Scanner;
public class DuplicateElementsFrequency {
public static void main(String[] args) {
Scanner scanner = new Scanner(System.in);
System.out.print("Enter the size of the array: ");
int size = scanner.nextInt();
int[] array = new int[size];
System.out.println("Enter elements of the array:");
for (int i = 0; i < size; i++) {
System.out.print("Enter element " + (i + 1) + ": ");
array[i] = scanner.nextInt();
}
Map<Integer, Integer> frequencyMap = findDuplicateElements(array);
System.out.println("Duplicate elements and their frequencies:");
for (Map.Entry<Integer, Integer> entry : frequencyMap.entrySet()) {
System.out.println(entry.getKey() + " occurs " + entry.getValue() + " times.");
}
scanner.close();
}
public static Map<Integer, Integer> findDuplicateElements(int[] array) {
Map<Integer, Integer> frequencyMap = new HashMap<>();
for (int num : array) {
if (frequencyMap.containsKey(num)) {
frequencyMap.put(num, frequencyMap.get(num) + 1);
} else {
frequencyMap.put(num, 1);
}
}
frequencyMap.entrySet().removeIf(entry -> entry.getValue() == 1);
return frequencyMap;
}
}
```

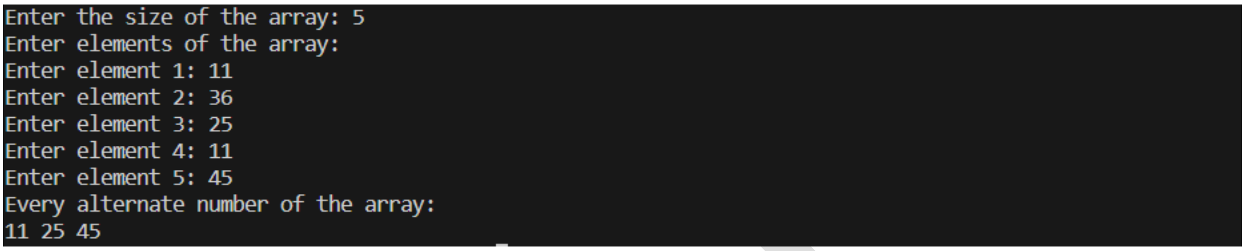


```
Enter the size of the array: 5
Enter elements of the array:
Enter element 1: 10
Enter element 2: 20
Enter element 3: 10
Enter element 4: 60
Enter element 5: 35
Duplicate elements and their frequencies:
10 occurs 2 times.
```

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

```
import java.util.Scanner;
public class AlternateNumbers {
public static void main(String[] args) {
Scanner scanner = new Scanner(System.in);
System.out.print("Enter the size of the array: ");
int size = scanner.nextInt();
int[] array = new int[size];
System.out.println("Enter elements of the array:");
for (int i = 0; i < size; i++) {
```

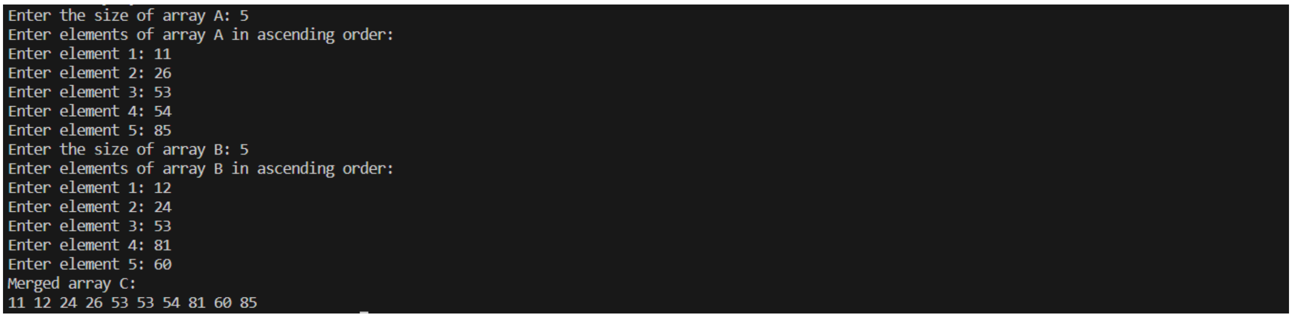
```
System.out.print("Enter element " + (i + 1) + ": ");
array[i] = scanner.nextInt();
}
System.out.println("Every alternate number of the array:");
for (int i = 0; i < size; i += 2) {
System.out.print(array[i] + " ");
}
scanner.close();
}
}
```



```
Enter the size of the array: 5
Enter elements of the array:
Enter element 1: 11
Enter element 2: 36
Enter element 3: 25
Enter element 4: 11
Enter element 5: 45
Every alternate number of the array:
11 25 45
```

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.

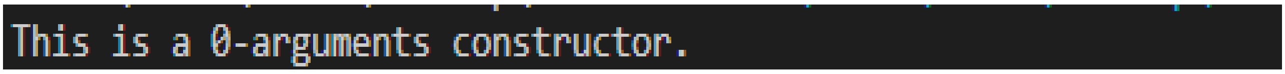
```
import java.util.Scanner;
public class MergeSortedArrays {
public static void main(String[] args) {
Scanner scanner = new Scanner(System.in);
System.out.print("Enter the size of array A: ");
int sizeA = scanner.nextInt();
int[] A = new int[sizeA];
System.out.println("Enter elements of array A in ascending order:");
for (int i = 0; i < sizeA; i++) {
System.out.print("Enter element " + (i + 1) + ": ");
A[i] = scanner.nextInt();
}
System.out.print("Enter the size of array B: ");
int sizeB = scanner.nextInt();
int[] B = new int[sizeB];
System.out.println("Enter elements of array B in ascending order:");
for (int i = 0; i < sizeB; i++) {
System.out.print("Enter element " + (i + 1) + ": ");
B[i] = scanner.nextInt();
}
int[] C = mergeSortedArrays(A, B);
System.out.println("Merged array C:");
for (int num : C) {
System.out.print(num + " ");
}
scanner.close();
}
public static int[] mergeSortedArrays(int[] A, int[] B) {
int sizeA = A.length;
int sizeB = B.length;
int sizeC = sizeA + sizeB;
int[] C = new int[sizeC];
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;
}
}
```



```
Enter the size of array A: 5
Enter elements of array A in ascending order:
Enter element 1: 11
Enter element 2: 26
Enter element 3: 53
Enter element 4: 54
Enter element 5: 85
Enter the size of array B: 5
Enter elements of array B in ascending order:
Enter element 1: 12
Enter element 2: 24
Enter element 3: 53
Enter element 4: 81
Enter element 5: 60
Merged array C:
11 12 24 26 53 53 54 81 60 85
```

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

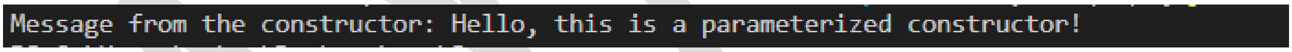
```
public class ZeroArgumentsConstructor {

public ZeroArgumentsConstructor() {                                    // 0-arguments constructor
System.out.println("This is a 0-arguments constructor.");
}

public static void main(String[] args) {
ZeroArgumentsConstructor example = new ZeroArgumentsConstructor();
}
}
```



```
This is a 0-arguments constructor.
```

25.Write a Java program to show parameterized constructor.

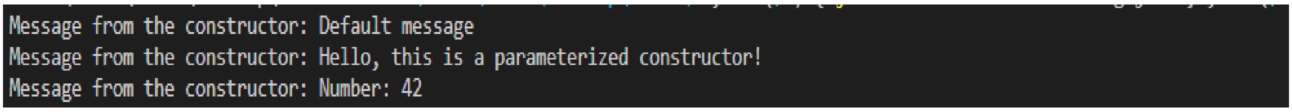
```
public class ParameterizedConstructor {
private String message;
public ParameterizedConstructor(String msg) {
this.message = msg;
}
public void displayMessage() {
System.out.println("Message from the constructor: " + message);
}
public static void main(String[] args) {
ParameterizedConstructor example = new ParameterizedConstructor("Hello, this is a parameterized constructor!");
example.displayMessage();
}
}
```



```
Message from the constructor: Hello, this is a parameterized constructor!
```

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

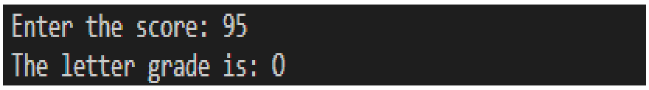
```
public class ConstructorOverloading {

private String message;

public ConstructorOverloading() {
this.message = "Default message";
}
public ConstructorOverloading(String msg) {
this.message = msg;
}
public ConstructorOverloading(int number) {
this.message = "Number: " + number;
}
public void displayMessage() {
System.out.println("Message from the constructor: " + message);
}
public static void main(String[] args) {
ConstructorOverloading example1 = new ConstructorOverloading();
ConstructorOverloading example2 = new ConstructorOverloading("Hello, this is a parameterized constructor!");
ConstructorOverloading example3 = new ConstructorOverloading(42);
example1.displayMessage();
example2.displayMessage();
example3.displayMessage();
}
}
```



```
Message from the constructor: Default message
Message from the constructor: Hello, this is a parameterized constructor!
Message from the constructor: Number: 42
```

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.

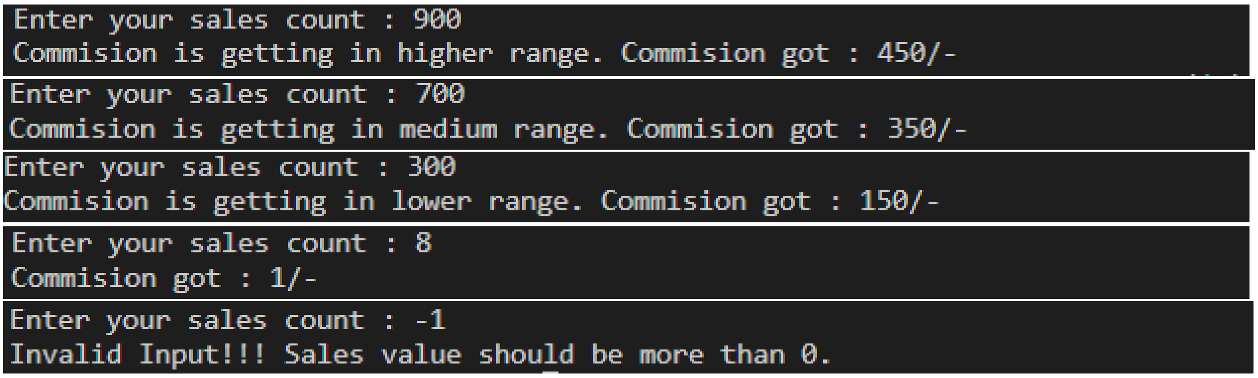
```
import java.util.Scanner;

public class Grader {
private int score;

// Constructor
public Grader(int score) {
this.score = score;
}

// Method to get the letter grade based on the score
public String letterGrade() {
if (score >= 90 && score <= 100) {
return "O";
} else if (score >= 80 && score < 90) {
return "E";
} else if (score >= 70 && score < 80) {
return "A";
} else if (score >= 60 && score < 70) {
return "B";
} else if (score >= 50 && score < 60) {
return "C";
} else {
return "F";
}
}

public static void main(String[] args) {
Scanner scanner = new Scanner(System.in);
System.out.print("Enter the score: ");
int userScore = scanner.nextInt();
if (userScore >= 0 && userScore <= 100) {
Grader grader = new Grader(userScore);
System.out.println("The letter grade is: " + grader.letterGrade());
} else {
System.out.println("Invalid score. Please enter a score between 0 and 100.");
}
scanner.close();
}
}
```



```
Enter the score: 95
The letter grade is: O
```

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".

```
import java.util.*;

class Commision {

int sales;
int commision;

Commision(int s) {
this.sales = s;
}

public void calculateCommision() {

if (sales < 1000 && sales > 800) {
commision = (sales * 50) / 100;
System.out.println("Commision is getting in higher range. Commision got : " + commision + "/-");
} else if (sales < 800 && sales > 400) {
commision = (sales * 50) / 100;
System.out.println("Commision is getting in medium range. Commision got : " + commision + "/-");
} else if (sales < 400 && sales > 10) {
commision = (sales * 50) / 100;
System.out.println("Commision is getting in lower range. Commision got : " + commision + "/-");
} else if (sales < 10 && sales > 0) {
commision = (sales * 20) / 100;
System.out.println("Commision got : " + commision + "/-");
} else {
System.out.println("Invalid Input!!! Sales value should be more than 0.");
}
}
}
public class CommissionDetails {
public static void main(String[] args) {
Scanner sc = new Scanner(System.in);
System.out.print("Enter your sales count : ");
int s = sc.nextInt();
Commision commision1 = new Commision(s);
commision1.calculateCommision();}}
```



```
Enter your sales count : 900
Commision is getting in higher range. Commision got : 450/-
```

```
Enter your sales count : 700
Commision is getting in medium range. Commision got : 350/-
```

```
Enter your sales count : 300
Commision is getting in lower range. Commision got : 150/-
```

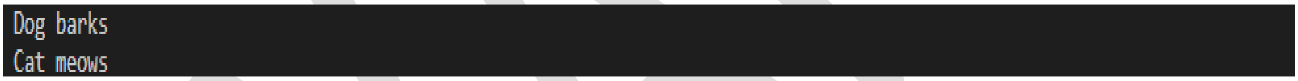
```
Enter your sales count : 8
Commision got : 1/-
```

```
Enter your sales count : -1
Invalid Input!!! Sales value should be more than 0.
```

Week 4

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

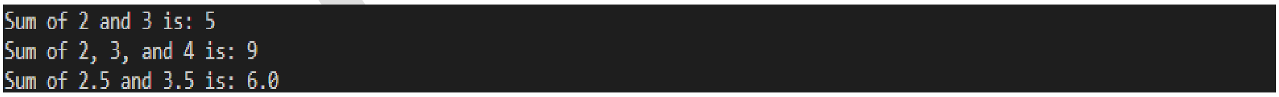
```
class Animal {
void sound() {
System.out.println("Animal makes a sound");
}}
class Dog extends Animal {
@Override
void sound() {
System.out.println("Dog barks");
}}
class Cat extends Animal {
@Override
void sound() {
System.out.println("Cat meows");
}}
public class One {
public static void main(String[] args) {
Dog dog = new Dog();
Cat cat = new Cat();
dog.sound();
cat.sound();
}}
```



```
Dog barks
Cat meows
```

2. Write a Java program to show method overloading.

```
public class Two {
static int add(int a, int b) {
return a + b;}
static int add(int a, int b, int c) {
return a + b + c;}
static double add(double a, double b) {
return a + b;}
public static void main(String[] args) {
System.out.println("Sum of 2 and 3 is: " + add(2, 3));
System.out.println("Sum of 2, 3, and 4 is: " + add(2, 3, 4));
System.out.println("Sum of 2.5 and 3.5 is: " + add(2.5, 3.5));
}}
```

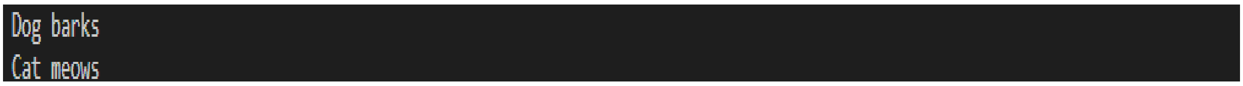


```
Sum of 2 and 3 is: 5
Sum of 2, 3, and 4 is: 9
Sum of 2.5 and 3.5 is: 6.0
```

3. Write a Java program to show method overriding.

```
class Animal {
void sound() {
System.out.println("Animal makes a sound");}}
class Dog extends Animal {
@Override
void sound() {
System.out.println("Dog barks");}}
class Cat extends Animal {
@Override
void sound() {
```

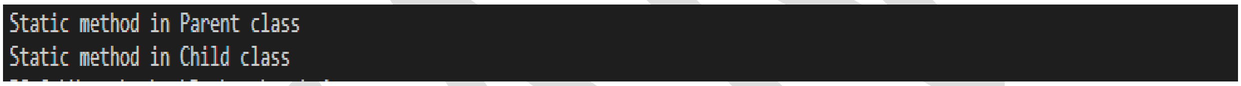
```
System.out.println("Cat meows");}}
public class Three {
public static void main(String[] args) {
Animal dog = new Dog();
Animal cat = new Cat();
dog.sound();
cat.sound();}}
```



```
Dog barks
Cat meows
```

4. Write a Java program to show method hiding.

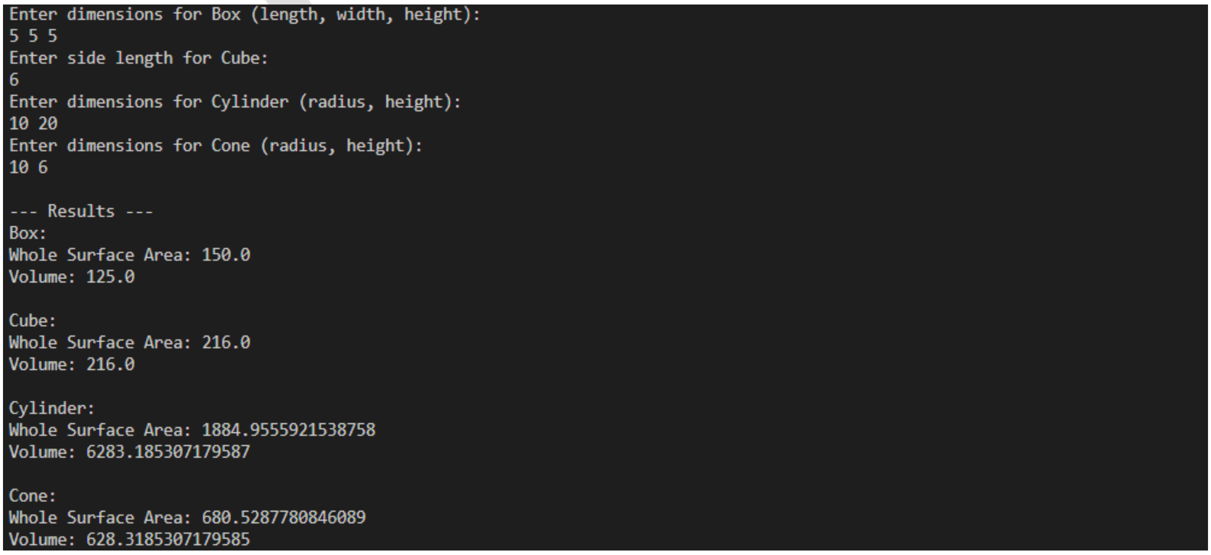
```
class Parent {
static void display() {
System.out.println("Static method in Parent class");}}
class Child extends Parent {
static void display() {
System.out.println("Static method in Child class");}}
public class Four {
public static void main(String[] args) {
Parent.display();
Child.display();}}
```



```
Static method in Parent class
Static method in Child class
```

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.

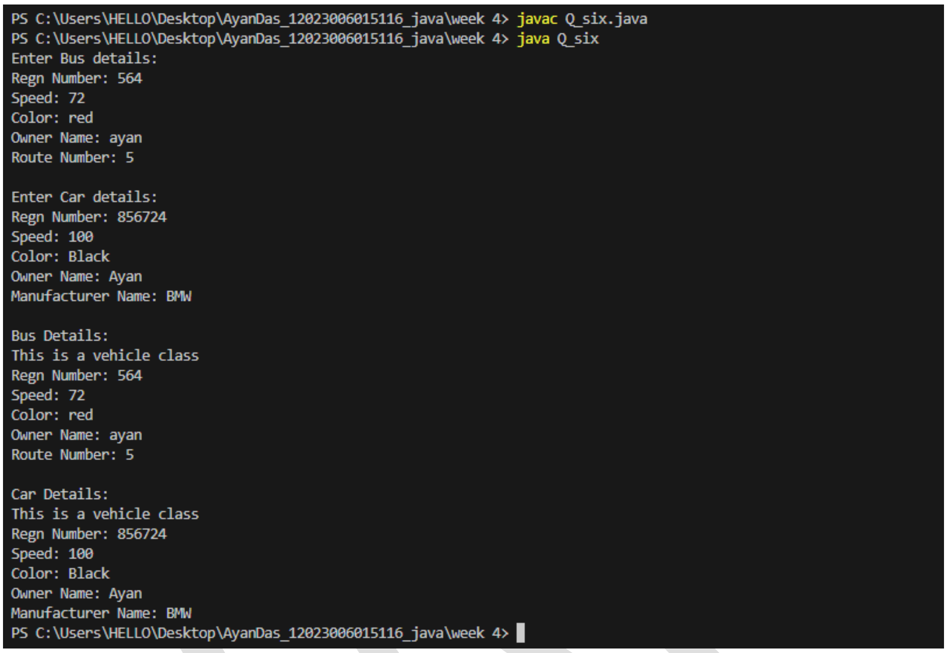
```
import java.util.Scanner;
class ThreeDObject {
ThreeDObject() {}
double wholeSurfaceArea() {
return 0.0;
}
double volume() {
return 0.0;
}
}
class Box extends ThreeDObject {
double length, width, height;

Box(double length, double width, double
height) {
this.length = length;
this.width = width;
this.height = height;
}
@Override
double wholeSurfaceArea() {
return 2 * (length * width + length * height +
width * height);
}
@Override
double volume() {
return length * width * height;
}
}
class Cube extends ThreeDObject {
double side;

Cube(double side) {
this.side = side;
}
@Override
double wholeSurfaceArea() {
return 6 * side * side;
}
@Override
double volume() {
return side * side * side;
}
}
class Cylinder extends ThreeDObject {
double radius, height;

Cylinder(double radius, double height) {
this.radius = radius;
this.height = height;
}
@Override
double wholeSurfaceArea() {
return 2 * Math.PI * radius * (radius + height);
}
@Override
double volume() {
return Math.PI * radius * radius * height;
}
}
class Cone extends ThreeDObject {
double radius, height;

Cone(double radius, double height) {
this.radius = radius;
this.height = height;
}
@Override
double wholeSurfaceArea() {
double slant_height = Math.sqrt(radius *
radius + height * height);
return Math.PI * radius * (radius +
slant_height);
}
@Override
double volume() {
return (1.0 / 3.0) * Math.PI * radius * radius *
height;
}
}
public class five {
public static void main(String[] args) {
Scanner scanner = new Scanner(System.in);
System.out.println("Enter dimensions for Box
(length, width, height):");
double length = scanner.nextDouble();
double width = scanner.nextDouble();
double height = scanner.nextDouble();
Box box = new Box(length, width, height);
System.out.println("Enter side length for
Cube:");
double side = scanner.nextDouble();
Cube cube = new Cube(side);
System.out.println("Enter dimensions for
Cylinder (radius, height):");
double radius = scanner.nextDouble();
height = scanner.nextDouble();
Cylinder cylinder = new Cylinder(radius,
height);
System.out.println("Enter dimensions for
Cone (radius, height):");
radius = scanner.nextDouble();
height = scanner.nextDouble();
Cone cone = new Cone(radius, height);
System.out.println("\n--- Results ---");
System.out.println("Box:");
System.out.println("Whole Surface Area: " +
box.wholeSurfaceArea());
System.out.println("Volume: " +
box.volume());
System.out.println("\nCube:");
System.out.println("Whole Surface Area: " +
cube.wholeSurfaceArea());
System.out.println("Volume: " +
cube.volume());
System.out.println("\nCylinder:");
System.out.println("Whole Surface Area: " +
cylinder.wholeSurfaceArea());
System.out.println("Volume: " +
cylinder.volume());
System.out.println("\nCone:");
System.out.println("Whole Surface Area: " +
cone.wholeSurfaceArea());
System.out.println("Volume: " +
cone.volume());

scanner.close();
}
}
```



```
Enter dimensions for Box (length, width, height):
5 5 5
Enter side length for Cube:
6
Enter dimensions for Cylinder (radius, height):
10 20
Enter dimensions for Cone (radius, height):
10 6

--- Results ---
Box:
Whole Surface Area: 150.0
Volume: 125.0

Cube:
Whole Surface Area: 216.0
Volume: 216.0

Cylinder:
Whole Surface Area: 1884.9555921538758
Volume: 6283.185307179587

Cone:
Whole Surface Area: 680.5287780846089
Volume: 628.3185307179585
```

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.

```
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 int routeNumber;
public Bus(String regnNumber, int speed,
String color, String ownerName, int
routeNumber) {
super(regnNumber, speed, color,
ownerName);
this.routeNumber = routeNumber;
}
@Override
protected void showData() {
super.showData();
System.out.println("Regn 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;
}

@Override
protected void showData() {
super.showData();
System.out.println("Regn 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 six {
public static void main(String[] args) {
Scanner scanner = new Scanner(System.in);

System.out.println("Enter Bus details:");
System.out.print("Regn Number: ");
String busRegnNumber = scanner.nextLine();
System.out.print("Speed: ");
int busSpeed = scanner.nextInt();
scanner.nextLine();
System.out.print("Color: ");
String busColor = scanner.nextLine();
System.out.print("Owner Name: ");
String busOwnerName = scanner.nextLine();
System.out.print("Route Number: ");
int routeNumber = scanner.nextInt();
Bus myBus = new Bus(busRegnNumber,
busSpeed, busColor, busOwnerName,
routeNumber);


System.out.println("\nEnter Car details:");
System.out.print("Regn Number: ");
String carRegnNumber = scanner.next();
System.out.print("Speed: ");
int carSpeed = scanner.nextInt();
scanner.nextLine();
System.out.print("Color: ");
String carColor = scanner.nextLine();
System.out.print("Owner Name: ");
String carOwnerName = scanner.nextLine();
System.out.print("Manufacturer Name: ");
String manufacturerName = scanner.nextLine();

Car myCar = new Car(carRegnNumber, carSpeed, carColor, carOwnerName, manufacturerName);

System.out.println("\nBus Details:");
myBus.showData();

System.out.println("\nCar Details:");
myCar.showData();

scanner.close();
}
}
```

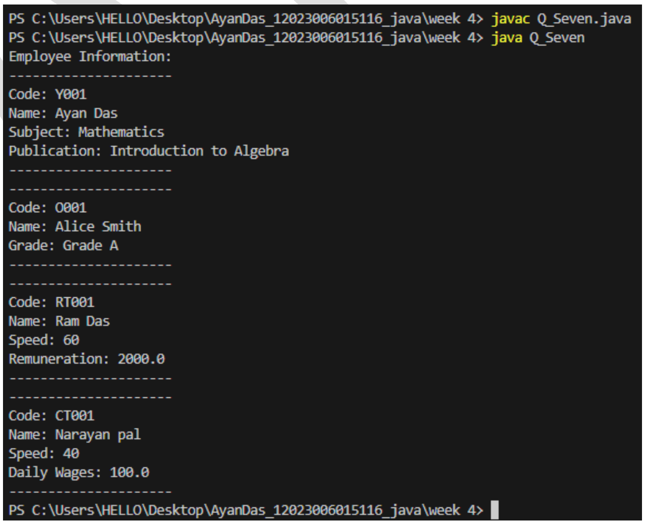


```
PS C:\Users\HELLO\Desktop\AyanDas_12023006015116_java\week 4> javac Q_six.java
PS C:\Users\HELLO\Desktop\AyanDas_12023006015116_java\week 4> java Q_six
Enter Bus details:
Regn Number: 564
Speed: 72
Color: red
Owner Name: ayan
Route Number: 5

Enter Car details:
Regn Number: 856724
Speed: 100
Color: Black
Owner Name: Ayan
Manufacturer Name: BMW

Bus Details:
This is a vehicle class
Regn Number: 564
Speed: 72
Color: red
Owner Name: ayan
Route Number: 5

Car Details:
This is a vehicle class
Regn Number: 856724
Speed: 100
Color: Black
Owner Name: Ayan
Manufacturer Name: BMW
PS C:\Users\HELLO\Desktop\AyanDas_12023006015116_java\week 4>
```

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. Staff (code, name) Teacher (subject, publication) is a Staff Officer (grade) is a Staff Typist (speed) is a Staff RegularTypist (remuneration) is a Typist CasualTypist (daily wages) is a Typist.

```
import java.util.ArrayList;
import java.util.Scanner;

class Staff {
protected String code;
protected String name;
public Staff(String code, String name) {
this.code = code;
this.name = name;
}
public void displayInfo() {
System.out.println("Code: " + code);
System.out.println("Name: " + name);
}
}
class Teacher extends Staff {
private String subject;
private String publication;
public Teacher(String code, String name, String subject, String publication) {
super(code, name);
this.subject = subject;
this.publication = publication;
}
@Override
public void displayInfo() {
super.displayInfo();
System.out.println("Subject: " + subject);
System.out.println("Publication: " + publication);
}
}

class Officer extends Staff {
private String grade;
public Officer(String code, String name, String grade) {
super(code, name);
this.grade = grade;
}
@Override
public void displayInfo() {
super.displayInfo();
System.out.println("Grade: " + grade);
}
}
```

```
class Typist extends Staff {
private int speed;

public Typist(String code, String name, int
speed) {
super(code, name);
this.speed = speed;
}
@Override
public void displayInfo() {
super.displayInfo();
System.out.println("Speed: " + speed);
}
}
class RegularTypist extends Typist {
private double remuneration;

public RegularTypist(String code, String name,
int speed, double remuneration) {
super(code, name, speed);
this.remuneration = remuneration;
}
@Override
public void displayInfo() {
super.displayInfo();
System.out.println("Remuneration: " +
remuneration);
}
}
class CasualTypist extends Typist {
private double dailyWages;
public CasualTypist(String code, String name,
int speed, double dailyWages) {
super(code, name, speed);
this.dailyWages = dailyWages;
}
@Override
public void displayInfo() {
super.displayInfo();
System.out.println("Daily Wages: " +
dailyWages);
}
}
public class seven {
public static void main(String[] args) {
Scanner scanner = new Scanner(System.in);
ArrayList<Staff> employees = new
ArrayList<>();
employees.add(new Teacher("T001", "John
Doe", "Mathematics", "Introduction to
Algebra"));
employees.add(new Officer("O001", "Alice
Smith", "Grade A"));
employees.add(new RegularTypist("RT001",
"Emma Johnson", 60, 2000.0));
employees.add(new CasualTypist("CT001",
"Michael Brown", 40, 100.0));
System.out.println("Employee Information:");
for (Staff employee : employees) {
System.out.println("--------------------");
employee.displayInfo();
System.out.println("--------------------");
}
scanner.close();
}
}
```

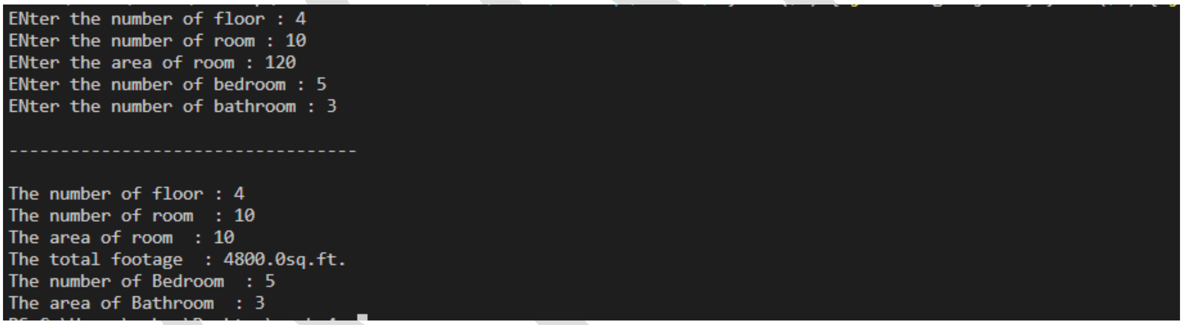


```
PS C:\Users\HELLO\Desktop\AyanDas_12023006015116_java\week 4> javac Q_Seven.java
PS C:\Users\HELLO\Desktop\AyanDas_12023006015116_java\week 4> java Q_Seven
Employee Information:
--------------------
Code: Y001
Name: Ayan Das
Subject: Mathematics
Publication: Introduction to Algebra
--------------------
--------------------
Code: O001
Name: Alice Smith
Grade: Grade A
--------------------
--------------------
Code: RT001
Name: Ram Das
Speed: 60
Remuneration: 2000.0
--------------------
--------------------
Code: CT001
Name: Narayan pal
Speed: 40
Daily Wages: 100.0
--------------------
PS C:\Users\HELLO\Desktop\AyanDas_12023006015116_java\week 4>
```

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.**

```
import java.util.*;
class Building {
```

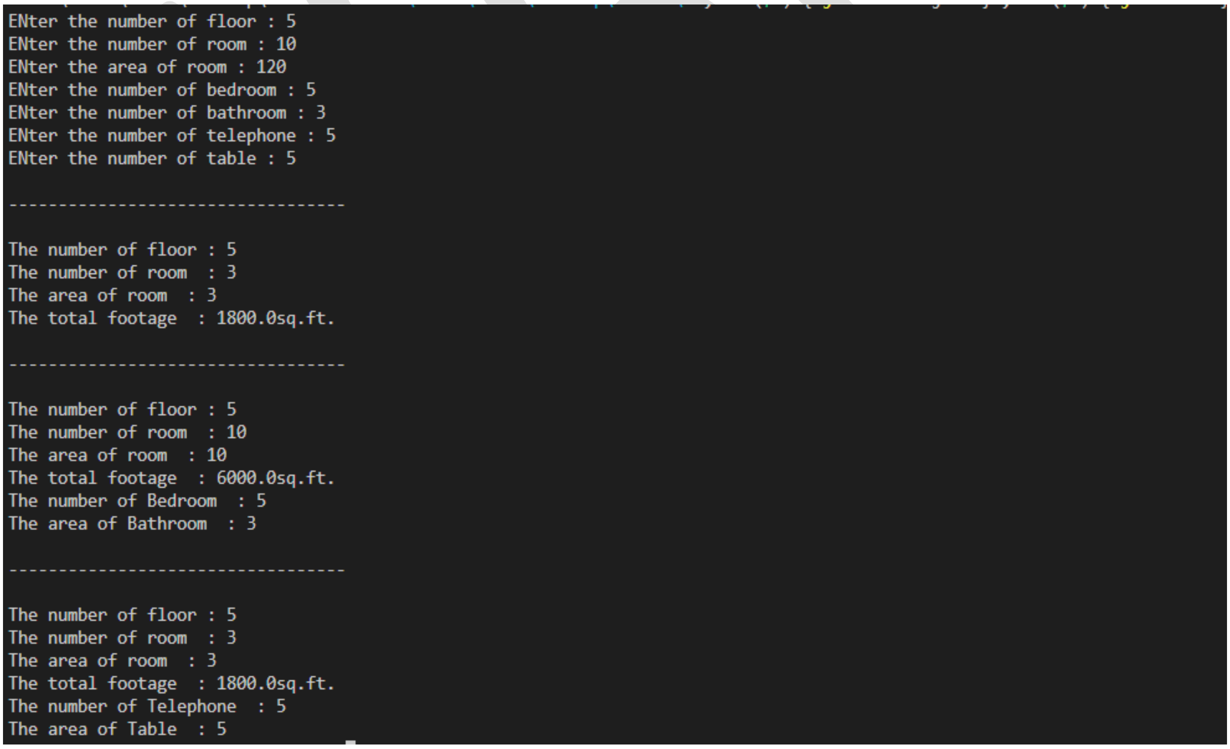
```
public void demonstrate(int floorNo, int
roomNo, double roomArea, double footage) {
System.out.println("The number of floor : " +
floorNo);
System.out.println("The number of room  : " +
roomNo);
System.out.println("The area of room   : " +
roomNo);
System.out.println("The total footage   : " +
(floorNo * roomNo * roomArea) + "sq.ft.");
}
}
class House extends Building {

public void demonstrate(int floorNo, int
roomNo, double roomArea, double footage,
int bedroomNo, int bathroomNo) {
super.demonstrate(floorNo, roomNo,
roomArea, footage);
System.out.println("The number of Bedroom  :
" + bedroomNo);
System.out.println("The area of Bathroom  : "
+ bathroomNo);
}
}
public class Eight {
public static void main(String[] args) {
House h = new House();
Scanner sc = new Scanner(System.in);
int floorNo, roomNo, bedroomNo,
bathroomNo;
double roomArea, footage = 0;

System.out.print("ENter the number of floor :
");
floorNo = sc.nextInt();
System.out.print("ENter the number of room :
");
roomNo = sc.nextInt();
System.out.print("ENter the area of room : ");
roomArea = sc.nextDouble();
System.out.print("ENter the number of
bedroom : ");
bedroomNo = sc.nextInt();
System.out.print("ENter the number of
bathroom : ");
bathroomNo = sc.nextInt();
System.out.println("\n------------------------------
---\n");
h.demonstrate(floorNo, roomNo, roomArea,
footage, bedroomNo, bathroomNo);
}}
```



```
ENter the number of floor : 4
ENter the number of room : 10
ENter the area of room : 120
ENter the number of bedroom : 5
ENter the number of bathroom : 3

---------------------------------

The number of floor : 4
The number of room  : 10
The area of room  : 10
The total footage  : 4800.0sq.ft.
The number of Bedroom  : 5
The area of Bathroom  : 3
```

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.

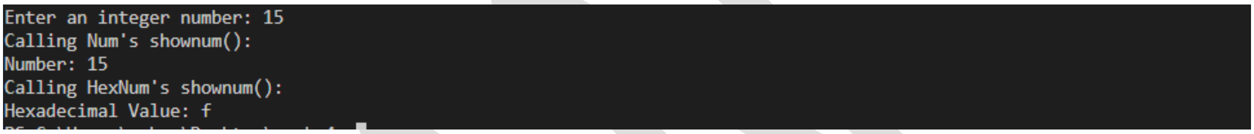
```
import java.util.*;
import java.util.Spliterator.OfPrimitive;

class Building {
public void demonstrate(int floorNo, int
roomNo, double roomArea, double footage) {
System.out.println("The number of floor : " +
floorNo);
System.out.println("The number of room  : " +
roomNo);
System.out.println("The area of room   : " +
roomNo);
System.out.println("The total footage   : " +
(floorNo * roomNo * roomArea) + "sq.ft.");
}
}
class House extends Building {
public void demonstrate(int floorNo, int
roomNo, double roomArea, double footage, int
bedroomNo, int bathroomNo) {
super.demonstrate(floorNo, roomNo,
roomArea, footage);
System.out.println("The number of Bedroom  : "
+ bedroomNo);
System.out.println("The area of Bathroom  : " +
bathroomNo);
}
}
class Office extends Building {
public void demonstrate(int floorNo, int
roomNo, double roomArea, double footage, int
telephoneNo, int tableNo) {
super.demonstrate(floorNo, roomNo,
roomArea, footage);
System.out.println("The number of Telephone :
" + telephoneNo);
System.out.println("The area of Table : " +
tableNo);
}
}
public class Nine {
public static void main(String[] args) {
Building b = new Building();
House h = new House();
Office o = new Office();
Scanner sc = new Scanner(System.in);
int floorNo, roomNo, bedroomNo, bathroomNo,
telephoneNo, tableNo;
double roomArea, footage = 0;

System.out.print("ENter the number of floor : ");
floorNo = sc.nextInt();
System.out.print("ENter the number of room :
");
roomNo = sc.nextInt();
System.out.print("ENter the area of room : ");
roomArea = sc.nextDouble();
System.out.print("ENter the number of
bedroom : ");
bedroomNo = sc.nextInt();
System.out.print("ENter the number of
bathroom : ");
bathroomNo = sc.nextInt();
System.out.print("ENter the number of
telephone : ");
telephoneNo = sc.nextInt();
System.out.print("ENter the number of table :
");
tableNo = sc.nextInt();
System.out.println("\n--------------------------------
-\n");

b.demonstrate(floorNo, bathroomNo,
roomArea, footage);
System.out.println("\n--------------------------------
-\n");
h.demonstrate(floorNo, roomNo, roomArea,
footage, bedroomNo, bathroomNo);
System.out.println("\n--------------------------------
-\n");
o.demonstrate(floorNo, bathroomNo,
roomArea, footage, telephoneNo, tableNo);

}
}
```



```
ENter the number of floor : 5
ENter the number of room : 10
ENter the area of room : 120
ENter the number of bedroom : 5
ENter the number of bathroom : 3
ENter the number of telephone : 5
ENter the number of table : 5

---------------------------------

The number of floor : 5
The number of room  : 3
The area of room  : 3
The total footage  : 1800.0sq.ft.

---------------------------------

The number of floor : 5
The number of room  : 10
The area of room  : 10
The total footage  : 6000.0sq.ft.
The number of Bedroom  : 5
The area of Bathroom  : 3

---------------------------------

The number of floor : 5
The number of room  : 3
The area of room  : 3
The total footage  : 1800.0sq.ft.
The number of Telephone  : 5
The area of Table  : 5
```

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.

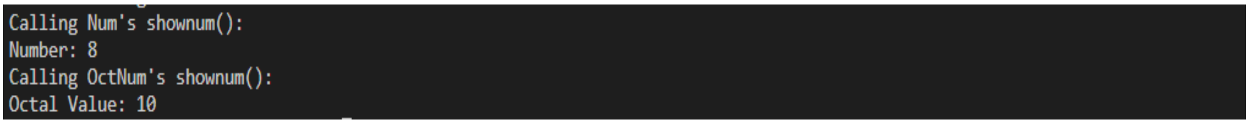
```
import java.util.Scanner;

class Num {
protected int number;
public Num(int number) {
this.number = number;
}
public void shownum() {
System.out.println("Number: " + number);
}
}
class HexNum extends Num {
public HexNum(int number) {
super(number);
}
@Override
public void shownum() {
System.out.println("Hexadecimal Value: " +
Integer.toHexString(number));
}
}
public class Ten {
public static void main(String[] args) {
Scanner scanner = new Scanner(System.in);
System.out.print("Enter an integer number: ");
int userInput = scanner.nextInt();
Num numObject = new Num(userInput);
System.out.println("Calling Num's
shownum():");
numObject.shownum();
HexNum hexNumObject = new
HexNum(userInput);
System.out.println("Calling HexNum's
shownum():");
hexNumObject.shownum();
scanner.close();
}}
```



```
Enter an integer number: 15
Calling Num's shownum():
Number: 15
Calling HexNum's shownum():
Hexadecimal Value: f
```

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.

```
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 {
public OctNum(int number) {
super(number);
}
@Override
public void shownum() {
System.out.println("Octal Value: " +
Integer.toOctalString(number));
}
}
public class Eleven {
public static void main(String[] args) {
Scanner scanner = new Scanner(System.in);
System.out.print("Enter an integer number: ");
int userInput = scanner.nextInt();
Num numObject = new Num(userInput);
System.out.println("Calling Num's
shownum():");
numObject.shownum();
OctNum octNumObject = new
OctNum(userInput);
System.out.println("Calling OctNum's
shownum():");
octNumObject.shownum();
scanner.close();
}
}
```

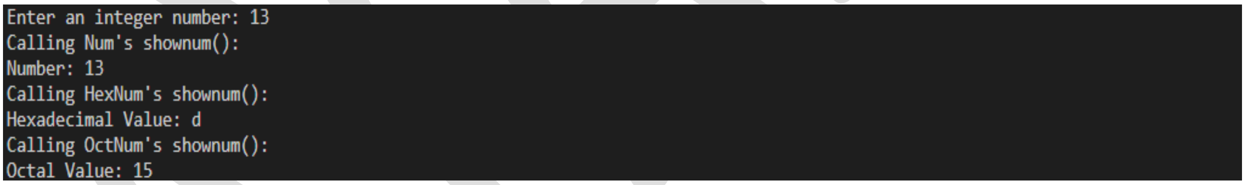


```
Calling Num's shownum():
Number: 8
Calling OctNum's shownum():
Octal Value: 10
```

12. Combine Question number 10 and 11 and have all the three classes together. Now describe the working of all classes.

```
import java.util.Scanner;

class Num {
```

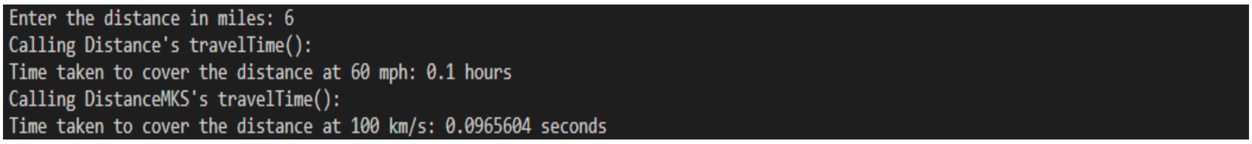
```
protected int number;
public Num(int number) {
this.number = number;
}
public void shownum() {
System.out.println("Number: " + number);
}
}
class HexNum extends Num {
public HexNum(int number) {
super(number);
}

@Override
public void shownum() {
System.out.println("Hexadecimal Value: " +
Integer.toHexString(number));
}
}
class OctNum extends Num {
public OctNum(int number) {
super(number);
}
@Override
public void shownum() {
System.out.println("Octal Value: " +
Integer.toOctalString(number));
}
}
}
public class Twelve {
public static void main(String[] args) {
Scanner scanner = new Scanner(System.in);
System.out.print("Enter an integer number: ");
int userInput = scanner.nextInt();
Num numObject = new Num(userInput);
System.out.println("Calling Num's
shownum():");
numObject.shownum();
HexNum hexNumObject = new
HexNum(userInput);
System.out.println("Calling HexNum's
shownum():");
hexNumObject.shownum();
OctNum octNumObject = new
OctNum(userInput);
System.out.println("Calling OctNum's
shownum():");
octNumObject.shownum();

scanner.close();
}
}
```



```
Enter an integer number: 13
Calling Num's shownum():
Number: 13
Calling HexNum's shownum():
Hexadecimal Value: d
Calling OctNum's shownum():
Octal Value: 15
```

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.**

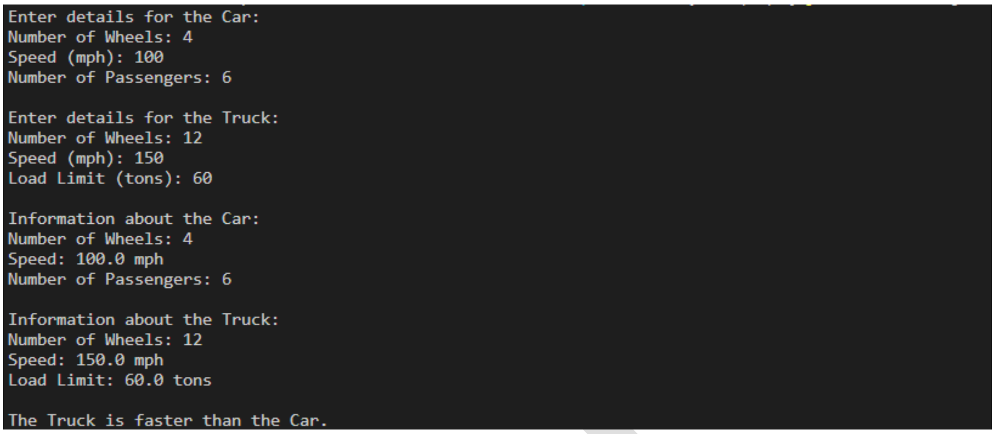
```
import java.util.Scanner;

class Distance {
protected double miles;
public Distance(double miles) {
this.miles = miles;
}
public void travelTime() {
double speedInMph = 60.0;
double timeInHours = miles / speedInMph;
System.out.println("Time taken to cover the
distance at 60 mph: " + timeInHours + " hours");
}
}
class DistanceMKS extends Distance {
public DistanceMKS(double miles) {
super(miles);
}
@Override
public void travelTime() {
double speedInKmps = 100.0;
double timeInSeconds = (miles * 1.60934) /
speedInKmps; // converting miles to kilometers
System.out.println("Time taken to cover the
distance at 100 km/s: " + timeInSeconds + "
seconds");
}
}
public class Thirteen {
public static void main(String[] args) {
Scanner scanner = new Scanner(System.in);
System.out.print("Enter the distance in miles: ");
double userInput = scanner.nextDouble();
Distance distanceObject = new Distance(userInput);
System.out.println("Calling Distance's travelTime():");
distanceObject.travelTime();
DistanceMKS distanceMKSObject = new DistanceMKS(userInput);
System.out.println("Calling DistanceMKS's travelTime():");
distanceMKSObject.travelTime();
scanner.close();
}
}
```



```
Enter the distance in miles: 6
Calling Distance's travelTime():
Time taken to cover the distance at 60 mph: 0.1 hours
Calling DistanceMKS's travelTime():
Time taken to cover the distance at 100 km/s: 0.0965604 seconds
```

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.**

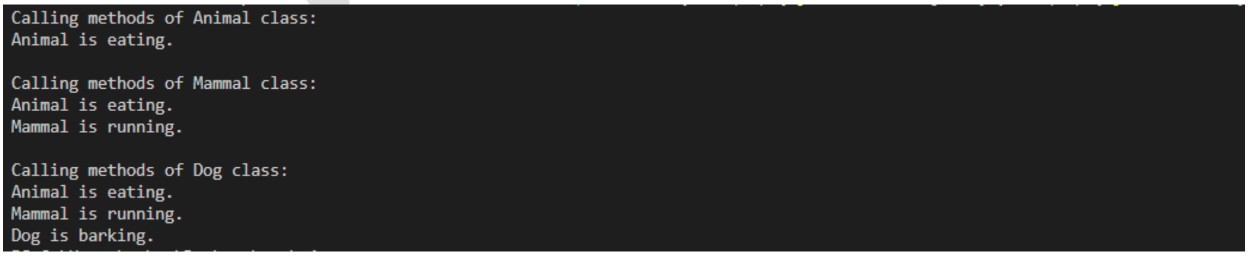
```
import java.util.Scanner;

class Vehicle {
protected int wheels;
protected double speed;

public Vehicle(int wheels, double speed) {
this.wheels = wheels;
this.speed = speed;
}
public void displayInfo() {
System.out.println("Number of Wheels: " + wheels);
System.out.println("Speed: " + speed + " mph");
}
}
class Car extends Vehicle {
private int passengers;
public Car(int wheels, double speed, int passengers) {
super(wheels, speed);
this.passengers = passengers;
}
@Override
public void displayInfo() {
super.displayInfo();
System.out.println("Number of Passengers: " + passengers);
}
}
class Truck extends Vehicle {
private double loadLimit;
public Truck(int wheels, double speed, double loadLimit) {
super(wheels, speed);
this.loadLimit = loadLimit;
}
@Override
public void displayInfo() {
super.displayInfo();
System.out.println("Load Limit: " + loadLimit + " tons");
}
}
public class Fourteen {
public static void main(String[] args) {
Scanner scanner = new Scanner(System.in);
System.out.println("Enter details for the Car:");
System.out.print("Number of Wheels: ");
int carWheels = scanner.nextInt();
System.out.print("Speed (mph): ");
double carSpeed = scanner.nextDouble();
System.out.print("Number of Passengers: ");
int carPassengers = scanner.nextInt();
Car car = new Car(carWheels, carSpeed, carPassengers);
System.out.println("\nEnter details for the Truck:");
System.out.print("Number of Wheels: ");
int truckWheels = scanner.nextInt();
System.out.print("Speed (mph): ");
double truckSpeed = scanner.nextDouble();
System.out.print("Load Limit (tons): ");
double truckLoadLimit = scanner.nextDouble();
Truck truck = new Truck(truckWheels, truckSpeed, truckLoadLimit);
System.out.println("\nInformation about the Car:");
car.displayInfo();
System.out.println("\nInformation about the Truck:");
truck.displayInfo();
if (car.speed > truck.speed) {
System.out.println("\nThe Car is faster than the Truck.");
} else if (car.speed < truck.speed) {
System.out.println("\nThe Truck is faster than the Car.");
} else {
System.out.println("\nThe Car and Truck have the same speed.");
}
scanner.close();
}
}
```



```
Enter details for the Car:
Number of Wheels: 4
Speed (mph): 100
Number of Passengers: 6

Enter details for the Truck:
Number of Wheels: 12
Speed (mph): 150
Load Limit (tons): 60

Information about the Car:
Number of Wheels: 4
Speed: 100.0 mph
Number of Passengers: 6

Information about the Truck:
Number of Wheels: 12
Speed: 150.0 mph
Load Limit: 60.0 tons

The Truck is faster than the Car.
```

15. Write a Java program to explain “multilevel inheritance.

```
// Base class
class Animal {
   void eat() {
      System.out.println("Animal is eating.");
   }
}

// First level derived class
class Mammal extends Animal {
   void run() {
      System.out.println("Mammal is running.");
   }
}

// Second level derived class
class Dog extends Mammal {
   void bark() {
      System.out.println("Dog is barking.");
   }
}

public class Fifteen {
   public static void main(String[] args) {
      Animal animal = new Animal();
      System.out.println("Calling methods of Animal class:");
      animal.eat();
      System.out.println();

      Mammal mammal = new Mammal();
      System.out.println("Calling methods of Mammal class:");
      mammal.eat();
      mammal.run();
      System.out.println();

      Dog dog = new Dog();
      System.out.println("Calling methods of Dog class:");
      dog.eat();
      dog.run();
      dog.bark();
   }
}
```

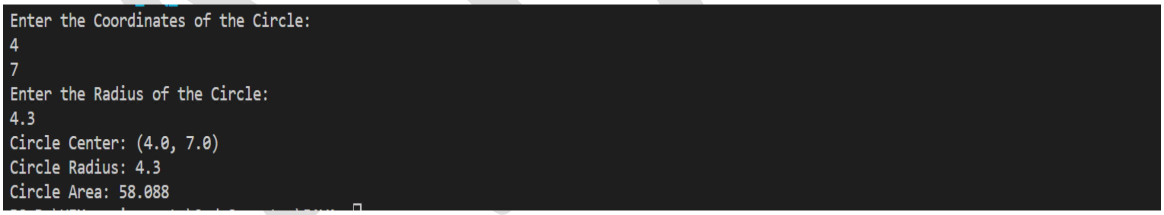


```
Calling methods of Animal class:
Animal is eating.

Calling methods of Mammal class:
Animal is eating.
Mammal is running.

Calling methods of Dog class:
Animal is eating.
Mammal is running.
Dog is barking.
```

Week 5

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

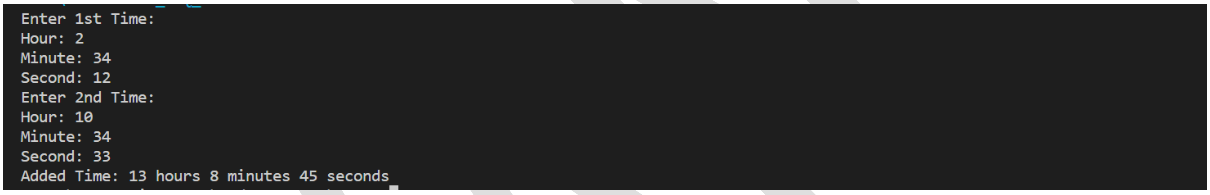
```
import java.lang.Math;
import java.util.Scanner;
class Point {
private double x;
private double y;
public Point(double x, double y) {
this.x = x;
this.y = y;}
public double getX() {
return x;
}
public double getY() {
return y;
}}
class Circle {
private Point center;
private double radius;
public Circle(Point center, double radius) {
this.center = center;
this.radius = radius;
}
public double calculateArea() {
return Math.PI * radius * radius;
}
public Point getCenter() {
return center;
}
public double getRadius() {
return radius;
}}
public class Q_1 {
public static void main(String[] args) {
Scanner sc = new Scanner(System.in);
System.out.println("Enter the Coordinates of the Circle: ");
double x = sc.nextDouble();
double y = sc.nextDouble();
Point center = new Point(x, y);
System.out.println("Enter the Radius of the Circle: ");
double radius = sc.nextDouble();
sc.close();
Circle circle = new Circle(center, radius);
System.out.println("Circle Center: (" + circle.getCenter().getX() + ", " + circle.getCenter().getY() + ")");
System.out.println("Circle Radius: " + circle.getRadius());
System.out.print("Circle Area: ");
System.out.printf("%.3f", circle.calculateArea());
}}
```



```
Enter the Coordinates of the Circle:
4
7
Enter the Radius of the Circle:
4.3
Circle Center: (4.0, 7.0)
Circle Radius: 4.3
Circle Area: 58.088
```

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.

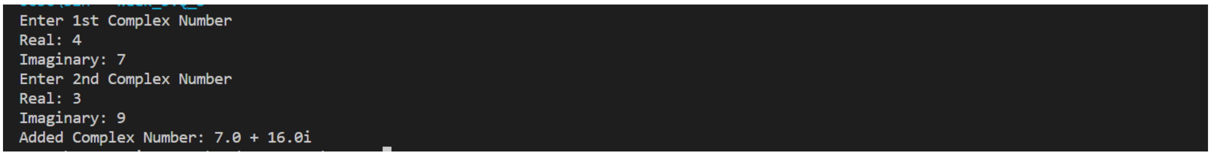
```
import java.util.Scanner;
class Time {
private int hour;
private int min;
private int sec;
public Time(int hour, int min, int sec) {
this.hour = hour;
this.min = min;
this.sec = sec;
}
public void add(Time other) {
int newHour = this.hour + other.hour;
int newMin = this.min + other.min;
int newSec = this.sec + other.sec;
if (newSec >= 60) {
newMin += newSec / 60;
newSec %= 60;
}
if (newMin >= 60) {
newHour += newMin / 60;
newMin %= 60;
}
if (newHour >= 24) {
newHour %= 24;
}
System.out.println("Added Time: " + newHour + " hours " + newMin + " minutes " + newSec + " seconds");
}
}
public class Q_2 {
public static void main(String[] args) {
Scanner sc = new Scanner(System.in);
System.out.println("Enter 1st Time:");
System.out.print("Hour: ");
int hour = sc.nextInt();
System.out.print("Minute: ");
int min = sc.nextInt();
System.out.print("Second: ");
int sec = sc.nextInt();
Time time1 = new Time(hour, min, sec);
System.out.println("Enter 2nd Time:");
System.out.print("Hour: ");
hour = sc.nextInt();
System.out.print("Minute: ");
min = sc.nextInt();
System.out.print("Second: ");
sec = sc.nextInt();
sc.close();
Time time2 = new Time(hour, min, sec);
time1.add(time2);
}
}
```



```
Enter 1st Time:
Hour: 2
Minute: 34
Second: 12
Enter 2nd Time:
Hour: 10
Minute: 34
Second: 33
Added Time: 13 hours 8 minutes 45 seconds
```

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.

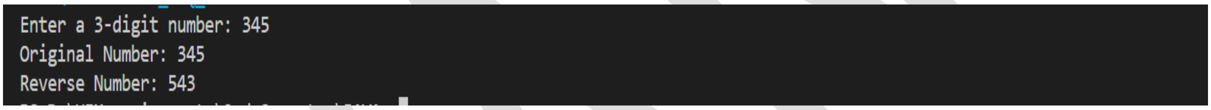
```
import java.util.Scanner;
class Complex {
private double real;
private double imaginary;
public Complex(double real, double imaginary) {
this.real = real;
this.imaginary = imaginary;
}
public void add(Complex other) {
double newReal = this.real + other.real;
double newImaginary = this.imaginary + other.imaginary;
System.out.println("Added Complex Number: " + newReal + " + " + newImaginary + "i");
}
}
public class Q_3 {
public static void main(String[] args) {
Scanner sc = new Scanner(System.in);
System.out.println("Enter 1st Complex Number");
System.out.print("Real: ");
double real = sc.nextInt();
System.out.print("Imaginary: ");
double imaginary = sc.nextInt();
Complex c1 = new Complex(real, imaginary);
System.out.println("Enter 2nd Complex Number");
System.out.print("Real: ");
real = sc.nextInt();
System.out.print("Imaginary: ");
imaginary = sc.nextInt();
Complex c2 = new Complex(real, imaginary);
sc.close();
c1.add(c2);
}
}
```



```
Enter 1st Complex Number
Real: 4
Imaginary: 7
Enter 2nd Complex Number
Real: 3
Imaginary: 9
Added Complex Number: 7.0 + 16.0i
```

4. Write a program to define a class having one 3-digit number, num as data member. Initialize and display reverse of that number.

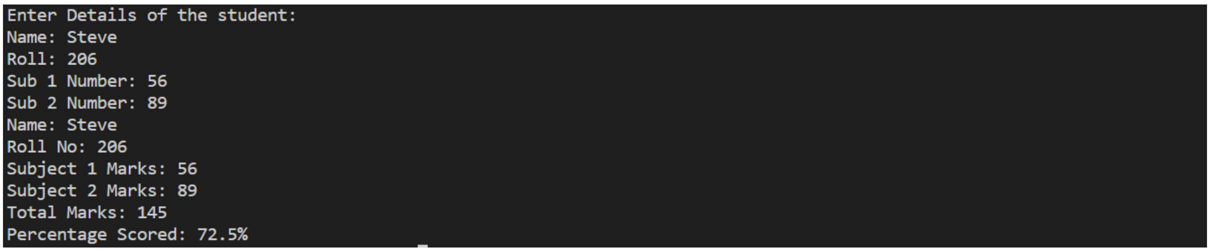
```
import java.util.Scanner;
class ThreeDigitNumber {
private int num;
public ThreeDigitNumber(int num) {
if (num >= 100 && num <= 999) {
this.num = num;
} else {
System.out.println("Error: The number must be a
3-digit number.");
System.exit(0);
}
}
public void displayReverse() {
int originalNum = num;
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 Number: " + reverse);
}
}
public class Q_4 {
public static void main(String[] args) {
Scanner sc = new Scanner(System.in);
System.out.print("Enter a 3-digit number: ");
int num = sc.nextInt();
sc.close();
ThreeDigitNumber number = new
ThreeDigitNumber(num);
number.displayReverse();
}
}
```



```
Enter a 3-digit number: 345
Original Number: 345
Reverse Number: 543
```

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.

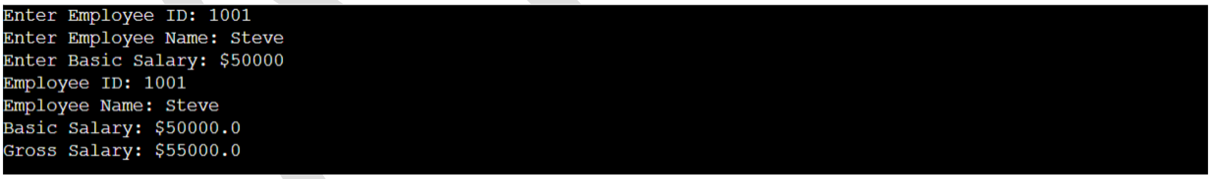
```
import java.util.Scanner;
class Student {
private String name;
private int rollNo;
private int sub1;
private int sub2;
public Student(String name, int rollNo, int sub1, int
sub2) {
this.name = name;
this.rollNo = rollNo;
this.sub1 = sub1;
this.sub2 = sub2;
}
public void displayStudentDetails() {
System.out.println("Name: " + name);
System.out.println("Roll No: " + rollNo);
System.out.println("Subject 1 Marks: " + sub1);
System.out.println("Subject 2 Marks: " + sub2);
}
public int calculateTotalMarks() {
return sub1 + sub2;
}
public double calculatePercentage() {
int totalMarks = calculateTotalMarks();
return (totalMarks / 2.0); // Assuming each subject
is out of 100
}}
public class Q_5 {
public static void main(String[] args) {
Scanner sc = new Scanner(System.in);
System.out.println("Enter Details of the student:");
System.out.print("Name: ");
String name = sc.nextLine();
System.out.print("Roll: ");
int roll = sc.nextInt();
System.out.print("Sub 1 Number: ");
int sub1 = sc.nextInt();
System.out.print("Sub 2 Number: ");
int sub2 = sc.nextInt();
sc.close();
Student student = new Student(name, roll, sub1,
sub2);
student.displayStudentDetails();
System.out.println("Total Marks: " +
student.calculateTotalMarks());
System.out.println("Percentage Scored: " +
student.calculatePercentage() + "%");
}
}
```



```
Enter Details of the student:
Name: Steve
Roll: 206
Sub 1 Number: 56
Sub 2 Number: 89
Name: Steve
Roll No: 206
Subject 1 Marks: 56
Subject 2 Marks: 89
Total Marks: 145
Percentage Scored: 72.5%
```

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.

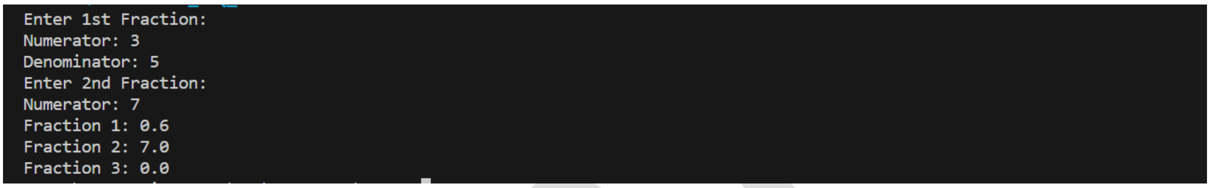
```
import java.util.Scanner;
class Employee {
private int empId;
private String empName;
private double basicSalary;
public Employee(int empId, String empName,
double basicSalary) {
this.empId = empId;
this.empName = empName;
this.basicSalary = basicSalary;
}
public double calculateGrossSalary() {
double allowance = 0.10 * basicSalary;
double grossSalary = basicSalary + allowance;
return grossSalary;
}
public void displayEmployeeDetails() {
System.out.println("Employee ID: " + empId);
System.out.println("Employee Name: " +
empName);
System.out.println("Basic Salary: $" + basicSalary);
System.out.println("Gross Salary: ₹" +
calculateGrossSalary());
}
}
public class Q_6 {
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
System.out.print("Enter Employee Name: ");
String empName = sc.nextLine();
System.out.print("Enter Basic Salary: ₹");
double basicSalary = sc.nextDouble();

Employee employee = new Employee(empId,
empName, basicSalary);
employee.displayEmployeeDetails();

sc.close();
}
}
```



```
Enter Employee ID: 1001
Enter Employee Name: Steve
Enter Basic Salary: $50000
Employee ID: 1001
Employee Name: Steve
Basic Salary: $50000.0
Gross Salary: $55000.0
```

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.

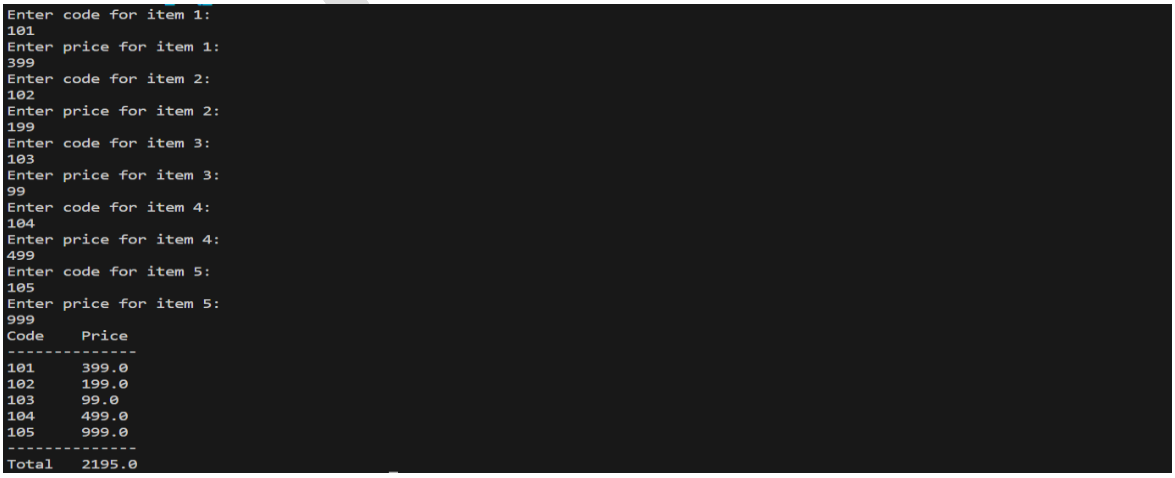
```
import java.util.Scanner;
class Fraction {
private int numerator;
private int denominator;
public Fraction(int numerator, int denominator) {
this.numerator = numerator;
this.denominator = denominator;
}
public Fraction(int numerator) {
this.numerator = numerator;
this.denominator = 1;
}
public Fraction() {
this.numerator = 0;
this.denominator = 1;
}
public double getFractionalValue() {
return (double) numerator / denominator;
}
}
public class Q_7 {
public static void main(String[] args) {
Scanner sc = new Scanner(System.in);
System.out.println("Enter 1st Fraction: ");
System.out.print("Numerator: ");
int numerator = sc.nextInt();
System.out.print("Denominator: ");
int denominator = sc.nextInt();
Fraction fraction1 = new Fraction(numerator,
denominator);
System.out.println("Enter 2nd Fraction: ");
System.out.print("Numerator: ");
numerator = sc.nextInt();
sc.close();
Fraction fraction2 = new Fraction(numerator);
Fraction fraction3 = new Fraction();
System.out.println("Fraction 1: " +
fraction1.getFractionalValue());
System.out.println("Fraction 2: " +
fraction2.getFractionalValue());
System.out.println("Fraction 3: " +
fraction3.getFractionalValue());
}
}
```



```
Enter 1st Fraction:
Numerator: 3
Denominator: 5
Enter 2nd Fraction:
Numerator: 7
Fraction 1: 0.6
Fraction 2: 7.0
Fraction 3: 0.0
```

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.

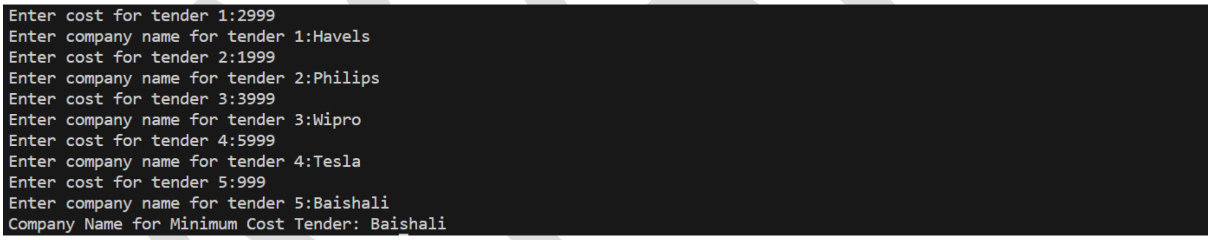
```
import java.util.Scanner;
class Item {
private String code;
private double price;
public Item(String code, double price) {
this.code = code;
this.price = price;}
public String getCode() {
return code;}
public double getPrice() {
return price;}}
public class Q_8 {
public static void main(String[] args) {
Scanner scanner = new Scanner(System.in);
Item[] items = new Item[5];
for (int i = 0; i < 5; i++) {
System.out.println("Enter code for item " + (i + 1) +
":");
String code = scanner.nextLine();
System.out.println("Enter price for item " + (i + 1) +
":");
double price = scanner.nextDouble();
scanner.nextLine(); // Consume newline
items[i] = new Item(code, price);
}
System.out.println("Code\tPrice");
System.out.println("--------------");
double total = 0;
for (Item item : items) {
System.out.println(item.getCode() + "\t" +
item.getPrice());
total += item.getPrice();
}
System.out.println("--------------");
System.out.println("Total\t" + total);
scanner.close();
}}
```



```
Enter code for item 1:
101
Enter price for item 1:
399
Enter code for item 2:
102
Enter price for item 2:
199
Enter code for item 3:
103
Enter price for item 3:
99
Enter code for item 4:
104
Enter price for item 4:
499
Enter code for item 5:
105
Enter price for item 5:
999
Code    Price
--------------
101     399.0
102     199.0
103     99.0
104     499.0
105     999.0
--------------
Total   2195.0
```

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.

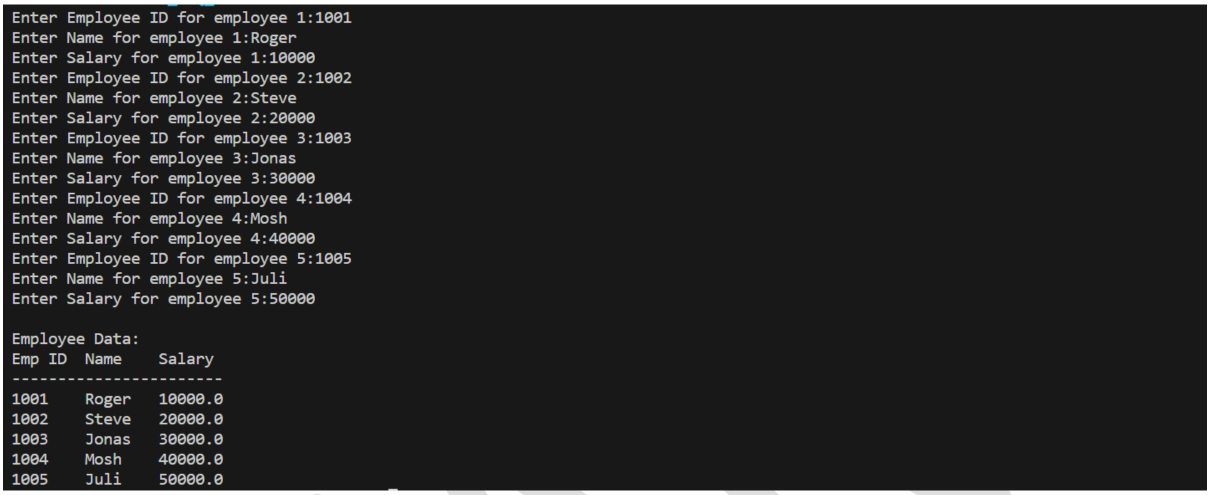
```
import java.util.Scanner;
class Tender {
private double cost;
private String companyName;
public Tender(double cost, String companyName) {
this.cost = cost;
this.companyName = companyName;
}
public double getCost() {
return cost;
}
public String getCompanyName() {
return companyName;
}
}
public class Q_9 {
public static void main(String[] args) {
Scanner scanner = new Scanner(System.in);
Tender[] tenders = new Tender[5];
// Accepting data for five objects
for (int i = 0; i < 5; i++) {
System.out.println("Enter cost for tender " + (i + 1) + ":");
double cost = scanner.nextDouble();
scanner.nextLine(); // Consume newline
System.out.println("Enter company name for tender " + (i + 1) + ":");
String companyName = scanner.nextLine();
tenders[i] = new Tender(cost, companyName);
}
Tender minCostTender = tenders[0];
for (int i = 1; i < tenders.length; i++) {
if (tenders[i].getCost() < minCostTender.getCost())
{
minCostTender = tenders[i];
}
}
System.out.println("Company Name for Minimum Cost Tender: " + minCostTender.getCompanyName());
scanner.close();
}
}
```



```
Enter cost for tender 1:2999
Enter company name for tender 1:Havels
Enter cost for tender 2:1999
Enter company name for tender 2:Philips
Enter cost for tender 3:3999
Enter company name for tender 3:Wipro
Enter cost for tender 4:5999
Enter company name for tender 4:Tesla
Enter cost for tender 5:999
Enter company name for tender 5:Baishali
Company Name for Minimum Cost Tender: Baishali
```

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.

```
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 int getEmpId() {
return empId;
}
public String getName() {
return name;
}
public double getSalary() {
return salary;
}
}
public class Q_10 {
public static void main(String[] args) {
Scanner scanner = new Scanner(System.in);
Employee[] employees = new Employee[5];
for (int i = 0; i < 5; i++) {
System.out.println("Enter Employee ID for employee " + (i + 1) + ":");
int empId = scanner.nextInt();
scanner.nextLine(); // Consume newline
System.out.println("Enter Name for employee " + (i + 1) + ":");
String name = scanner.nextLine();
System.out.println("Enter Salary for employee " + (i + 1) + ":");
double salary = scanner.nextDouble();
scanner.nextLine(); // Consume newline
employees[i] = new Employee(empId, name,
salary);
}
System.out.println("\nEmployee Data:");
System.out.println("Emp ID\tName\tSalary");
System.out.println("----------------------");
for (Employee employee : employees) {
System.out.println(employee.getEmpId() + "\t" +
employee.getName() + "\t" +
employee.getSalary());
}
scanner.close();
}
}
```

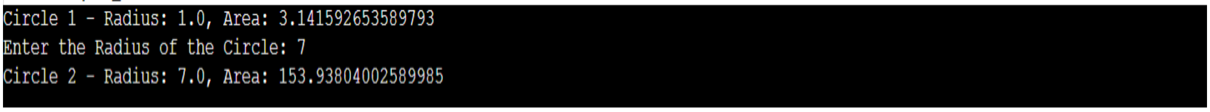


```
Enter Employee ID for employee 1:1001
Enter Name for employee 1:Roger
Enter Salary for employee 1:10000
Enter Employee ID for employee 2:1002
Enter Name for employee 2:Steve
Enter Salary for employee 2:20000
Enter Employee ID for employee 3:1003
Enter Name for employee 3:Jonas
Enter Salary for employee 3:30000
Enter Employee ID for employee 4:1004
Enter Name for employee 4:Mosh
Enter Salary for employee 4:40000
Enter Employee ID for employee 5:1005
Enter Name for employee 5:Juli
Enter Salary for employee 5:50000

Employee Data:
Emp ID  Name    Salary
----------------------
1001    Roger   10000.0
1002    Steve   20000.0
1003    Jonas   30000.0
1004    Mosh    40000.0
1005    Juli    50000.0
```

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.**

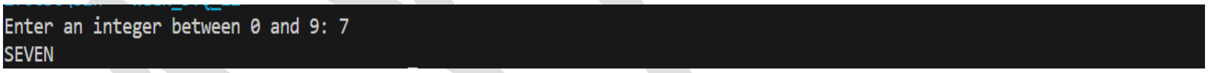
```
import java.lang.Math;
import java.util.Scanner;
class Circle {
private double radius;
private String color;
Circle() {
radius = 1.0;
color = "red";
}
Circle(double radius) {
this.radius = radius;
color = "red";
}
double getRadius() {
return radius;
}
double getArea() {
return Math.PI * radius * radius;
}
}
public class Q_11 {
public static void main(String[] args) {
Circle circle1 = new Circle();
System.out.println("Circle 1 - Radius: " +
circle1.getRadius() + ", Area: " + circle1.getArea());
Scanner sc = new Scanner(System.in);
System.out.print("Enter the Radius of the Circle: ");
double radius = sc.nextDouble();
sc.close();
Circle circle2 = new Circle(radius);
System.out.println("Circle 2 - Radius: " +
circle2.getRadius() + ", Area: " + circle2.getArea());
}
}
```



```
Circle 1 - Radius: 1.0, Area: 3.141592653589793
Enter the Radius of the Circle: 7
Circle 2 - Radius: 7.0, Area: 153.93804002589985
```

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.

```
import java.util.Scanner;
public class Q_12 {
public static void main(String[] args) {
Scanner scanner = new Scanner(System.in);
System.out.print("Enter an integer between 0 and
9: ");
int number = scanner.nextInt();
printNumberInWord(number);
scanner.close();
}
public static void printNumberInWord(int number)
{
switch (number) {
case 0:
System.out.println("ZERO");
break;
case 1:
System.out.println("ONE");
break;
case 2:
System.out.println("TWO");
break;
case 3:
System.out.println("THREE");
break;
case 4:
System.out.println("FOUR");
break;
case 5:
System.out.println("FIVE");
break;
case 6:
System.out.println("SIX");
break;
case 7:
System.out.println("SEVEN");
break;
case 8:
System.out.println("EIGHT");
break;
case 9:
System.out.println("NINE");
break;
default:
System.out.println("Invalid number. Please enter a
number between 0 and 9.");
}
}
}
```



```
Enter an integer between 0 and 9: 7
SEVEN
```

13. Design a class named Account that contains:

I. A private int data field named id for the account (default 0).

II. A private double data field named balance for the account (default 0).

III. A private double data field named annualInterestRate that stores the cur-rent interest rate (default 0). Assume all accounts have the same interest rate.

IV. A private Date data field named dateCreated that stores the date when the account was created. V. A no-arg constructor that creates a default account.

VI. A constructor that creates an account with the specified id and initial balance.

VII. The accessor and mutator methods for id,balance, and annualInterestRate.

VIII. The accessor method for dateCreated.

IX. A method named getMonthlyInterestRate() that returns the monthly interest rate.

X. A method named getMonthlyInterest() that returns the monthly interest.

XI. A method named withdraw that withdraws a specified amount from the account.

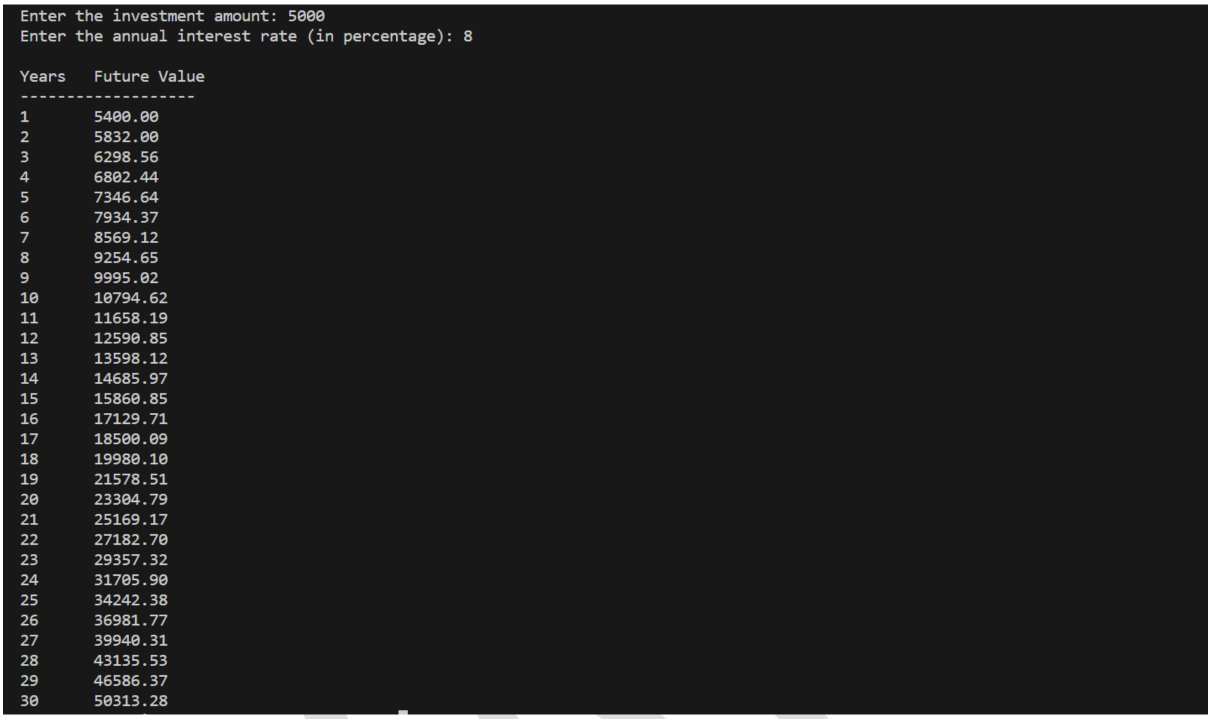
XII. A method named deposit that deposits a specified amount to the account.

```
import java.util.Date;
class Account {
private int id;
private double balance;
private static double annualInterestRate = 3; //
Assume all accounts have the same interest rate
private Date dateCreated;
// No-arg constructor that creates a default
account
public Account() {
id = 0;
balance = 0;
dateCreated = new Date();
}
// Constructor that creates an account with the specified id and initial balance
public Account(int id, double balance) {
this.id = id;
this.balance = balance;
dateCreated = new Date();
}
// Accessor method for id
public int getId() {
return id;
}
// Mutator method for id
public void setId(int id) {
this.id = id;
}
// Accessor method for balance
public double getBalance() {
return balance;
}
// Mutator method for balance
public void setBalance(double balance) {
this.balance = balance;
}
// Accessor method for annualInterestRate
public static double getAnnualInterestRate() {
return annualInterestRate;
}
// Method to deposit a specified amount to the account
public void deposit(double amount) {
balance += amount;
}}
// Mutator method for annualInterestRate
public static void setAnnualInterestRate(double annualInterestRate) {
Account.annualInterestRate = annualInterestRate;
}
// Accessor method for dateCreated
public Date getDateCreated() {
return dateCreated;
}
// Method to calculate and return the monthly interest rate
public double getMonthlyInterestRate() {
return annualInterestRate / 12;
}
// Method to calculate and return the monthly interest
public double getMonthlyInterest() {
return balance * (getMonthlyInterestRate() / 100);
}
// Method to withdraw a specified amount from the account
public void withdraw(double amount) {
if (amount <= balance) {
balance -= amount;
} else {
System.out.println("Insufficient funds!");
}
}
```

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

```
import java.util.Scanner;
public class Q_14 {
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() / 100; // Convert percentage to decimal
System.out.println("\nYears\tFuture Value");
System.out.println("-------------------");
for (int years = 1; years <= 30; years++) {
double futureValue = calculateFutureValue(investmentAmount, annualInterestRate, years);
System.out.printf("%-5d\t%.2f%n", years, futureValue);}
scanner.close();}
```

public static double calculateFutureValue(double investmentAmount, double annualInterestRate, int years) {
return investmentAmount * Math.pow(1 + annualInterestRate, years);}}



```
Enter the investment amount: 5000
Enter the annual interest rate (in percentage): 8

Years   Future Value
-------------------
1       5400.00
2       5832.00
3       6298.56
4       6802.44
5       7346.64
6       7934.37
7       8569.12
8       9254.65
9       9995.02
10      10794.62
11      11658.19
12      12590.85
13      13598.12
14      14685.97
15      15860.85
16      17129.71
17      18500.09
18      19980.10
19      21578.51
20      23304.79
21      25169.17
22      27182.70
23      29357.32
24      31705.90
25      34242.38
26      36981.77
27      39940.31
28      43135.53
29      46586.37
30      50313.28
```

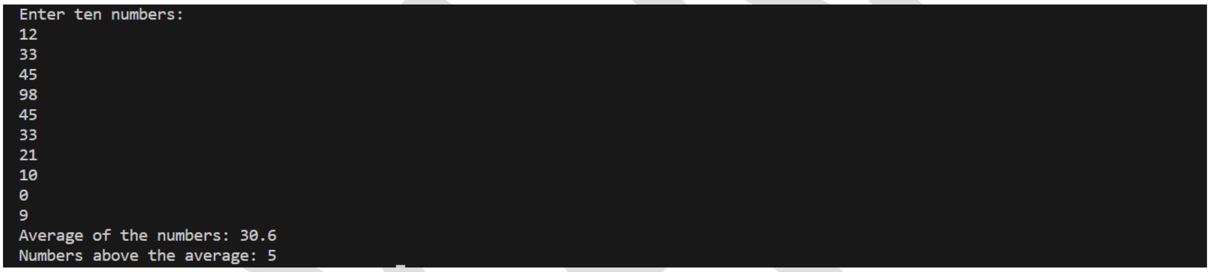
15. Write method headers for the following methods:
a. Computing a sales commission, given the sales amount and the commission rate.
b. Printing the calendar for a month, given the month and year.
c. Computing a square root.
d. Testing whether a number is even, and returning true if it is.
e. Printing a message a specified number of times.
f. Computing the monthly payment, given the loan amount, number of years, and annual interest rate.

```
public class Q_15 {
public double computeSalesCommission(double salesAmount, double commissionRate) {
// Method body
}

public void printCalendar(int month, int year) {
// Method body
}
public double computeSquareRoot(double number) {
// Method body
}
public boolean isEven(int number) {
// Method body
}

public void printMessage(String message, int times) {
// Method body
}
public double computeMonthlyPayment(double loanAmount, int numberOfYears,
double annualInterestRate) {
// Method body
}}
```

16. Write a program that reads ten numbers, computes their average, and finds out how many numbers are above the average. [Use this keyword]

```
import java.util.Scanner;
public class Q_16 {
private double[] numbers;
public Q_16() {
```

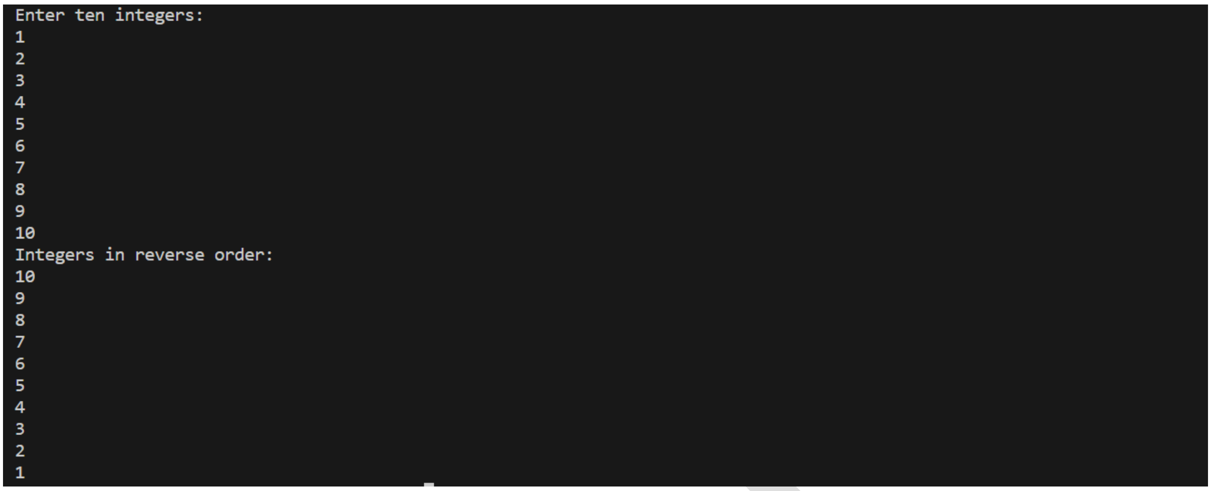
```
numbers = new double[10];
}
public void readNumbers() {
Scanner scanner = new Scanner(System.in);
System.out.println("Enter ten numbers:");
for (int i = 0; i < 10; i++) {
numbers[i] = scanner.nextDouble();
}
scanner.close();
}

public double calculateAverage() {
double sum = 0;
for (double number : this.numbers) {
sum += number;
}
return sum / this.numbers.length;
}
public int countAboveAverage() {
double average = this.calculateAverage();
int count = 0;
for (double number : this.numbers) {
if (number > average) {
count++;
}
}
return count;
}
public static void main(String[] args) {
Q_16 averageAndAbove = new Q_16();
averageAndAbove.readNumbers();
double average =
averageAndAbove.calculateAverage();
int countAboveAverage =
averageAndAbove.countAboveAverage();
System.out.println("Average of the numbers: " +
average);
System.out.println("Numbers above the average: "
+ countAboveAverage);
}
}
```



```
Enter ten numbers:
12
33
45
98
45
33
21
10
0
9
Average of the numbers: 30.6
Numbers above the average: 5
```

17. Write a program that reads ten integers and displays them in the reverse of the order in which they were read.

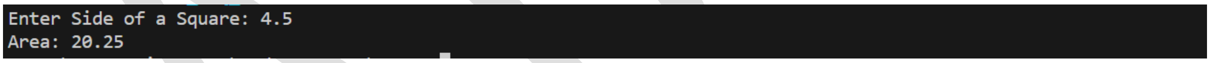
```
import java.util.Scanner;
public class Q_17 {
public static void main(String[] args) {
Scanner scanner = new Scanner(System.in);
int[] numbers = new int[10];
System.out.println("Enter ten integers:");
for (int i = 0; i < 10; i++) {
numbers[i] = scanner.nextInt();}
System.out.println("Integers in reverse order:");
for (int i = 9; i >= 0; i--) {
System.out.println(numbers[i]);}
scanner.close();}}
```



```
Enter ten integers:
1
2
3
4
5
6
7
8
9
10
Integers in reverse order:
10
9
8
7
6
5
4
3
2
1
```

18. Write a program to demonstrate use of 'this' keyword.

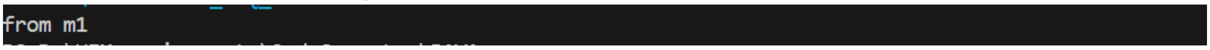
```
import java.util.Scanner;
class Square {
private double side;
Square(double side) {
this.side = side;}
double area() {
return side * side;}}
public class Q_18 {
public static void main(String[] args) {
Scanner sc = new Scanner(System.in);
System.out.print("Enter Side of a Square: ");
int side = sc.nextInt();
sc.close();
Square sq = new Square(side);
System.out.println("Area: " + sq.area());}}
```



```
Enter Side of a Square: 4.5
Area: 20.25
```

19. Write a program to demonstrate use of 'static' keyword.

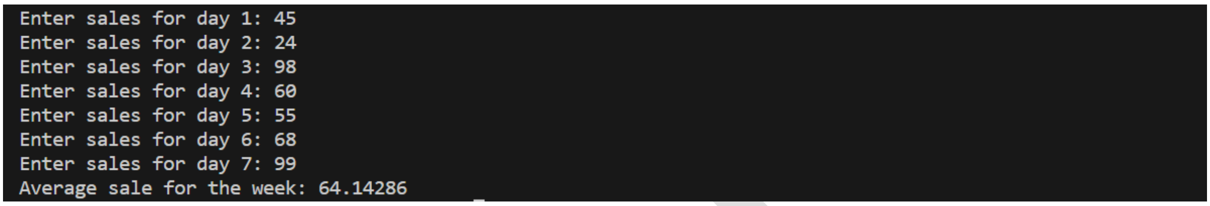
```
class Q_19 {
static void m1() {
System.out.println("from m1");
}
public static void main(String[] args) {
m1();
}}
```



```
from m1
```

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.

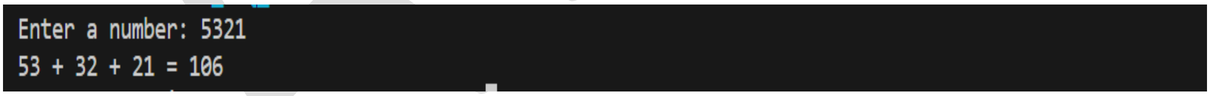
```
import java.util.Scanner;
public class Q_20 {
public static void main(String[] args) {
float[] sales = new float[7];
Scanner scanner = new Scanner(System.in);
for (int i = 0; i < 7; i++) {
System.out.print("Enter sales for day " + (i + 1) + ": ");
sales[i] = scanner.nextFloat();
}
scanner.close();
float totalSales = 0;
for (float sale : sales) {
totalSales += sale;
}
float averageSale = totalSales / 7;
System.out.println("Average sale for the week: " + averageSale);
}}
```



```
Enter sales for day 1: 45
Enter sales for day 2: 24
Enter sales for day 3: 98
Enter sales for day 4: 60
Enter sales for day 5: 55
Enter sales for day 6: 68
Enter sales for day 7: 99
Average sale for the week: 64.14286
```

21. Write program, which finds the sum of numbers formed by consecutive digits. Input : 2415 output : 24+41+15=80.

```
import java.util.Scanner;
public class Q_21 {
public static void main(String[] args) {
Scanner scanner = new Scanner(System.in);
System.out.print("Enter a number: ");
int number = scanner.nextInt();
String numberString = String.valueOf(number);
int sum = 0;
for (int i = 0; i < numberString.length() - 1; i++) {
String consecutiveDigits = numberString.substring(i, i + 2);
System.out.print(consecutiveDigits + " + ");
sum += Integer.parseInt(consecutiveDigits);
}
System.out.println("\b\b= " + sum);
scanner.close();
}}
```



```
Enter a number: 5321
53 + 32 + 21 = 106
```