Hustle

Core Java Basics

Contents

❖ Core Java Basics

Introduction To Java

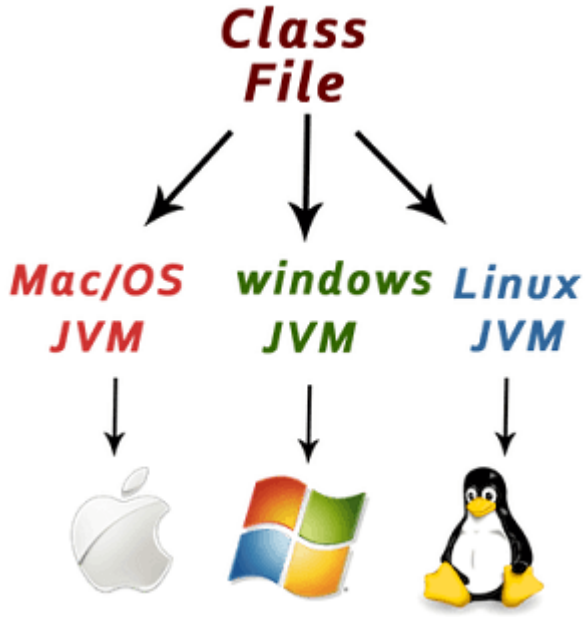
Java is a multipurpose, high-level, object oriented programming language which was developed by Sir James Gosling and team at Sun Microsystems in the year of 1995. Java is a statically typed language and is platform independent. It is used in developing Desktop Applications, Web Applications, Enterprise Softwares, Mobile OS, Embedded Systems, Smart Cards operated Security Systems and in many other spheres of technology.

Java Features

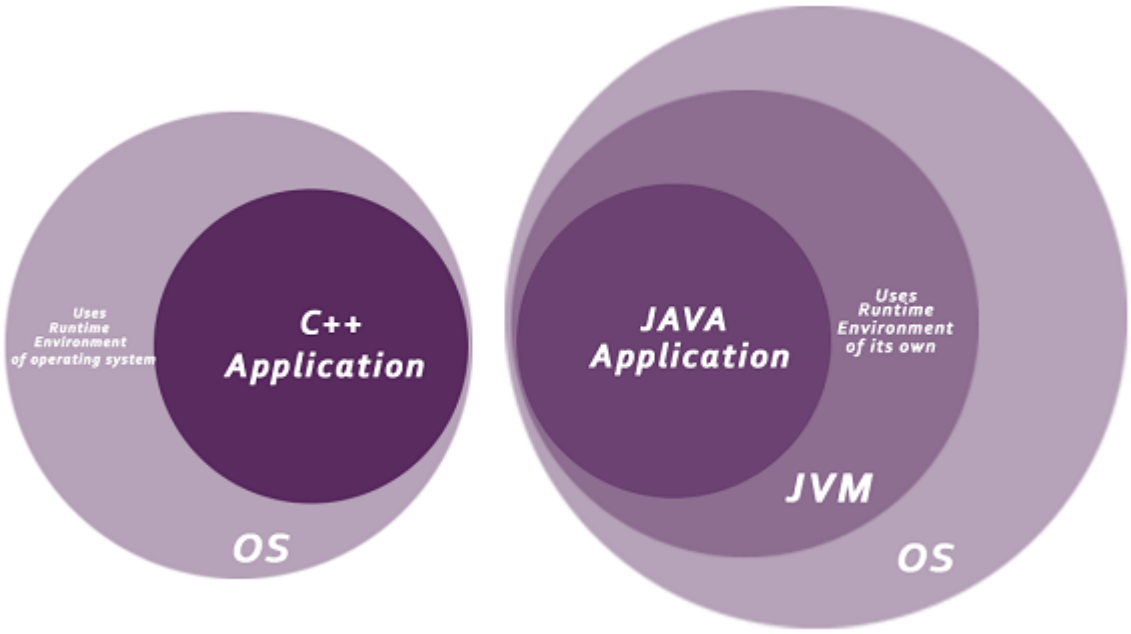
❖ **Simple -** Java is simple as because it has a simplified and clean syntax. Also Java has removed many rarely used functionalities like Operator Overloading and Explicit Pointers. Also due to the presence of an automatic garbage collection mechanism, the programmer is relieved of dealing with unreferenced objects.

❖ **Object Oriented Programming Language -** Java is an object oriented programming language.Object oriented programming language is a new programming paradigm, by which problems are solved referencing to real world objects. Objects are entities which have both data and behaviour

❖ **Platform Independent-** Java is platform independent as it is different from languages like C and C++ which are compiled into platform specific machines. On the other hand Java is a write once and run everywhere language where the source code is first compiled into an all platform bytecode which is common for the JVM of all operating systems like Windows, Mac and Linux. Next the bytecode is interpreted by system specific JVMs.



❖ **Secured-** Java is a very secured language as because there are no explicit pointers and also the program runs inside a virtual machine sandbox. Also beside these



❖ **Robust -** The English meaning of Robust is strong. Java is robust because:

❖ **Multithreaded -** A thread is like a separate program, executing concurrently. We can write Java programs that deal with many tasks at once by defining multiple threads. The main advantage of multi-threading is that it doesn't occupy memory for each thread. It shares a common memory area. Threads are important for multimedia, Web applications, etc.

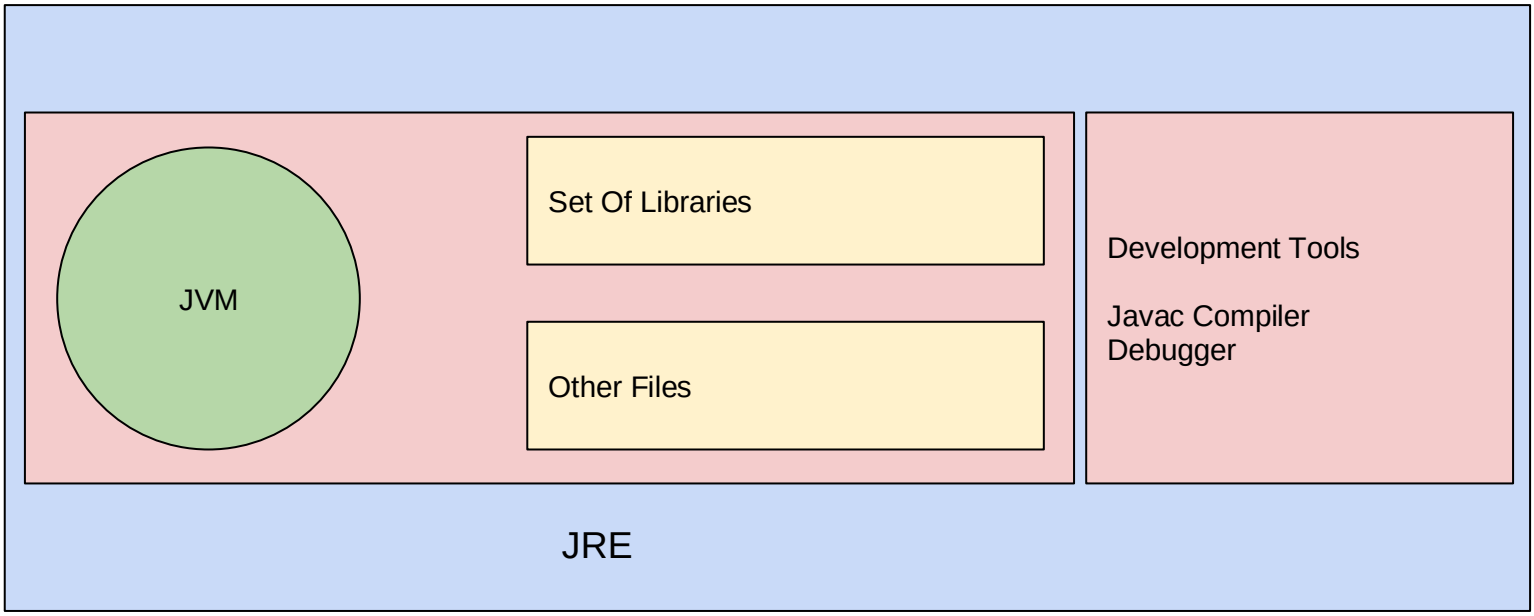
JDK, JRE & JVM

JDK (Java Development Kit)- It is a software development environment which is used to develop Java applications and applets.It contains JRE + development tools

JRE(Java Runtime Environment)- JRE stands for Java Runtime Environment. It is the implementation of JVM. The Java Runtime Environment is a set of software tools which are used for developing Java applications. It is used to provide the runtime environment. It is the implementation of JVM. It physically exists. It contains a set of libraries + other files that JVM uses at runtime.

JVM (Java Virtual Machine)- JVM is an abstract machine which provides the runtime environment in which Java bytecode can be executed. It is present inside the JRE

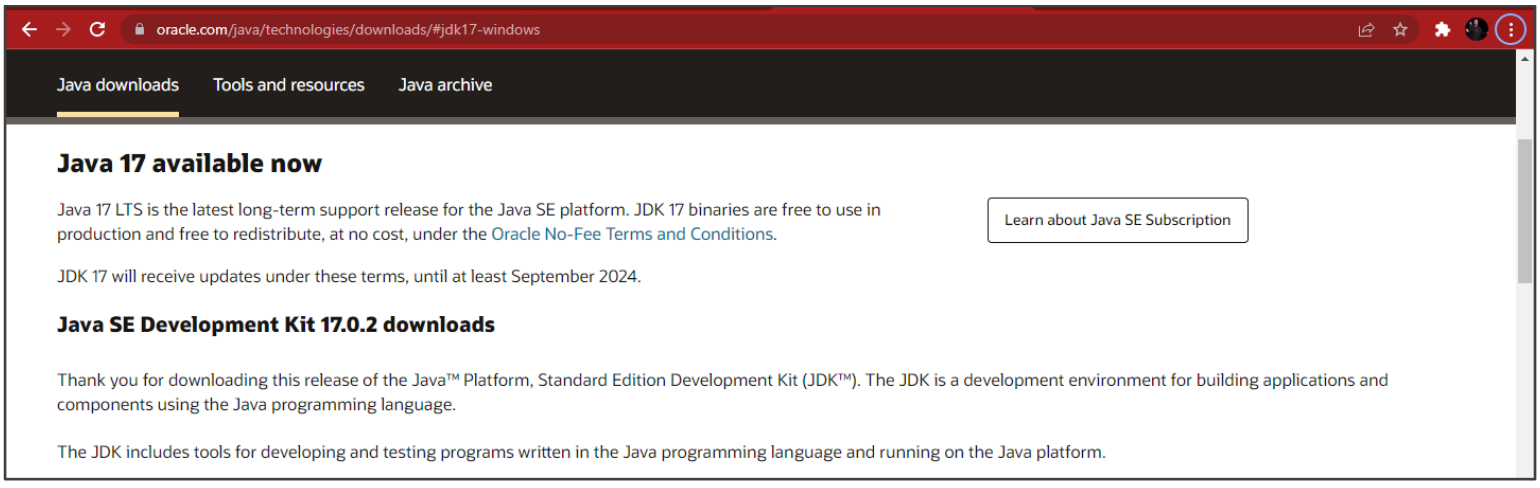
Schematic Diagram



Setting Up The Environment

Step 1- Downloading JDK

We need to visit the official Oracle website, where we can find the JDK files. We need to choose the version of JDK as per our need. During the drafting of this documentation, the latest JDK version which is available is JDK 17. Next we need to download the JDK file in accordance to our Operating System and install it in our system



Step 2 - Installation Of JDK

Find the JDK file in your downloads folder and click on it. Follow all the necessary steps and install the JDK in your preferred directory.

[Click Here To Read Documentation And Understand The Procedure]

Step 3- Verify Installation Of JDK

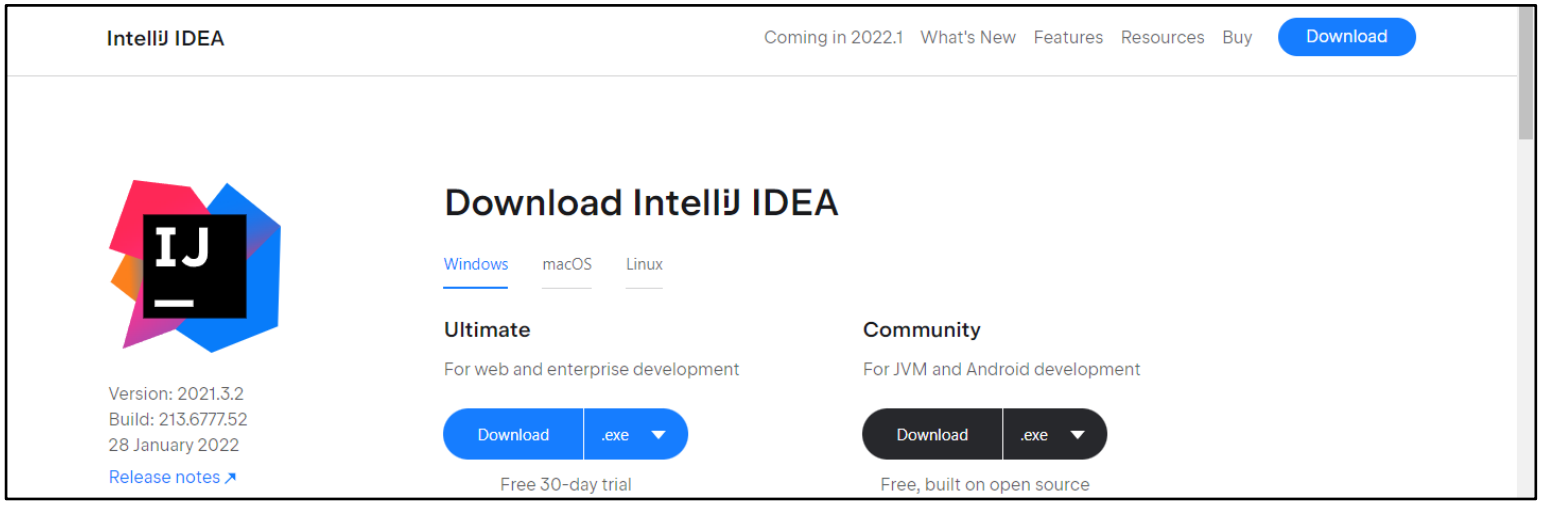
Open Powershell in your Windows Device and write the following command `java --version`. On successful installation, the Powershell will display the JDK version without throwing any warnings or errors

```
Microsoft Windows [Version 10.0.17134.1792]
(c) 2018 Microsoft Corporation. All rights reserved.

C:\Users\Aaron>java --version
java 17.0.1 2021-10-19 LTS
Java(TM) SE Runtime Environment (build 17.0.1+12-LTS-39)
Java HotSpot(TM) 64-Bit Server VM (build 17.0.1+12-LTS-39, mixed mode, sharing)

C:\Users\Aaron>
```

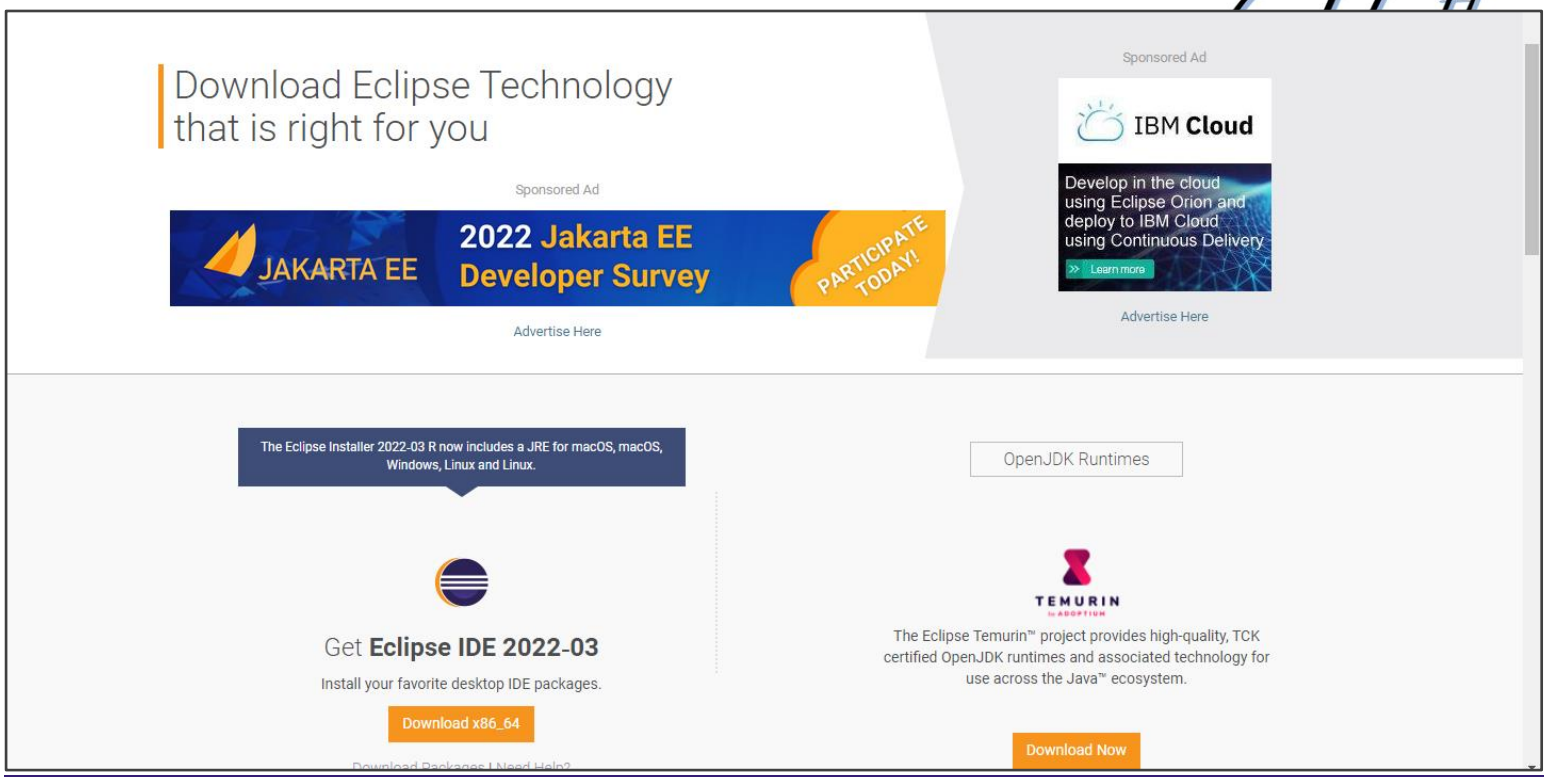
Step 4- Downloading The IDE- IDE stands for **Integrated Development Environment ,** it helps us in writing code by providing features like Syntax Highlighting, Error Pinpointing, Debug Reports and in many more ways. For developing softwares in Java, It is recommended that the user uses any one of the following IDEs. Download links are attached here with

- Jetbrains Intelli J Idea- Download
- Eclipse IDE- Download

Intelli J Idea



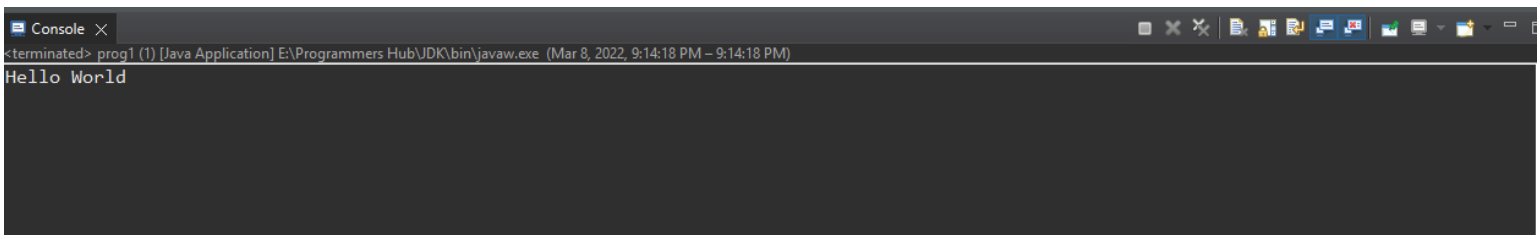
Eclipse IDE



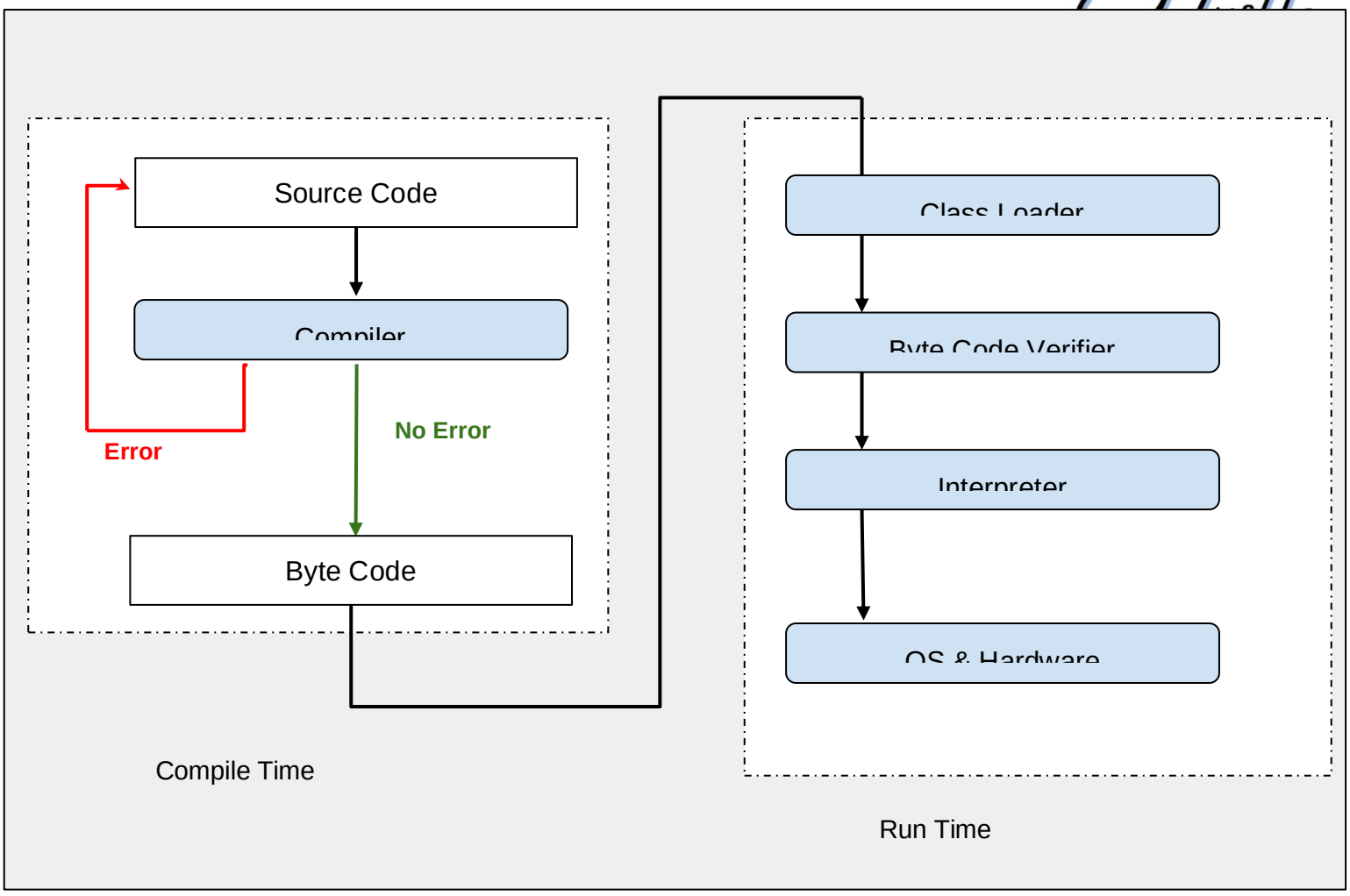
First Java Program

```
package Basics1;
public class prog1
{
    public static void main(String[] args) {

        System.out.println("Hello World");
}
}
```

Output

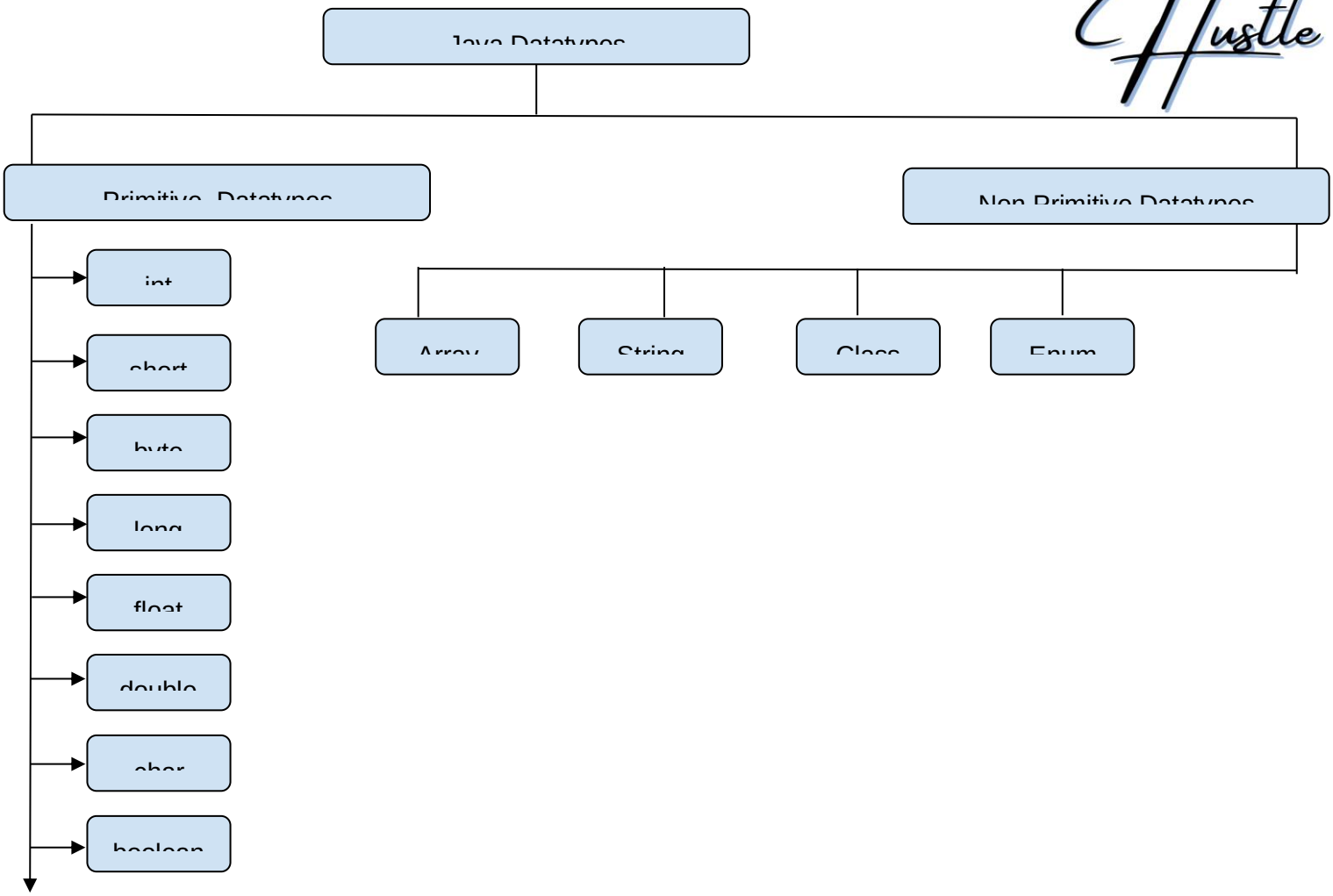


How Code Is Executed In Java



Java Datatypes

Datatypes in Java are primarily of two types



Sl No	Datatype	Size	Lower Range	Higher Range	Default Values
1	int	4 bytes	-2^{31}	$2^{31}-1$	0
2	bytes	1 byte	-2^{7}	$2^{7}-1$	0
3	short	2 bytes	-2^{15}	$2^{15}-1$	0
4	long	8 bytes	-2^{63}	$2^{63}-1$	0
5	float	4 bytes	-3.40282347×10^{38}	3.40282347×10^{38}	0.0f
6	double	8 bytes	$-1.7976931348623157\times10^{308}$	$1.7976931348623157\times10^{308}$	0.0d
7	char	2 bytes	'\u0000'	'\uffff'	'\u0000'
8	boolean	1 bit	false	true	false

We will discuss about the non primitive datatypes in later stages.

Java Variables

A variable is the name of a reserved area allocated in memory.

Rules For Declaring A Variable In Java

Rule	Demonstration	
	Allowed ☑	**Not Allowed ✕**
The variable name must not start with a numeral	abc123	123abc
The variable name must not start with any special characters except Dollar ($) and underscore (_)	$updatedvalue , _updatedvalue	#updatedvalue , @updatedvalue,
We can use hyphens(-), underscores(_), dollars ($) in between the names	updated$value, updated-value, updated_value	updated#value, updated@value, updated^value,
In Java, the naming follows the camel case convention. However capitalized and **snake case** namings are not forbidden.	newVariable new_variable NewVariable	New Variable

```
package Basics1;

public class prog1
{
    public static void main(String[] args) {

        String $name = "Sagnik";
        int $age = 24;
        System.out.println("My Name Is "+$name+" & age is "+$age);

}
}
```

Output-

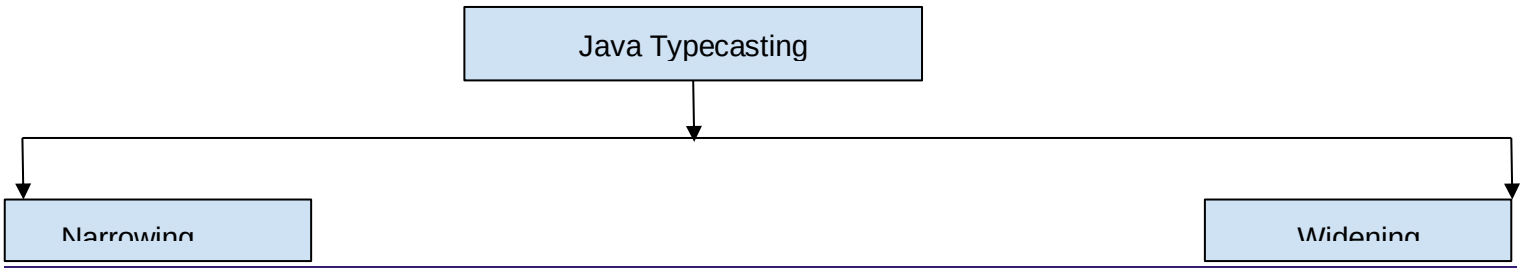
```
Console
<terminated> prog1 (1) [Java Application] E:\Programmers Hub\JDK\bin\javaw.exe (Mar 11, 2022, 12:50:55 PM – 12:50:57 PM)
My Name Is Sagnik & age is 24
```

Taking Input In Java

In Java, input is taken with the help of Scanner Class.

Java Typecasting

Typecasting in Java refers to the process where one datatype is converted to other. Typecasting can be of two types.



Narrowing- Narrowing is a process where a datatype of higher scope is converted to a datatype of a lower scope. e.g- Float to int or char to int

```
package Basics1;

import java.util.Scanner;

public class prog9 {


    public static void main(String[] args)
    {
        float floatValue=8475.3f;
        System.out.println("Example Of Float To Int "+(int)floatValue );
        char charValue='%';
        System.out.println("Example Of Char To Int "+(int)charValue );
    }

}
```

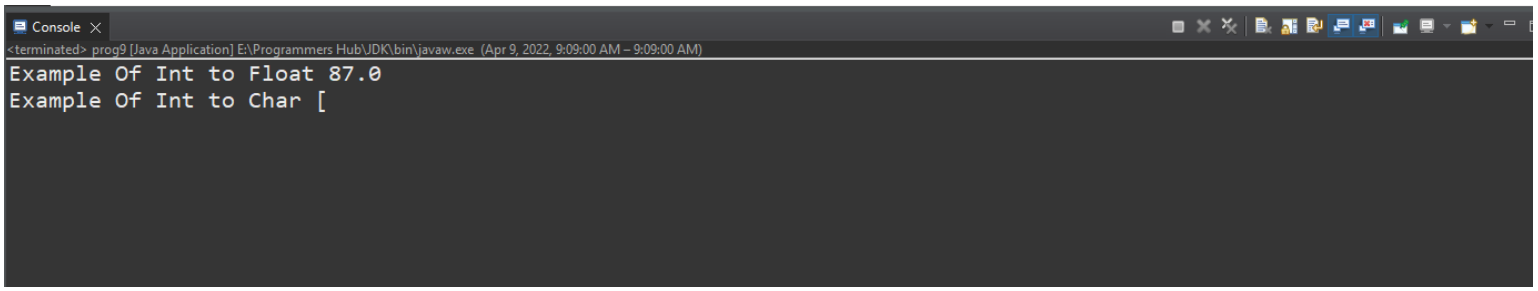
Output

```
Console ×
<terminated> prog9 [Java Application] E:\Programmers Hub\JDK\bin\javaw.exe (Apr 9, 2022, 9:02:35 AM – 9:02:35 AM)
Example Of Float To Int 8475
Example Of Char To Int 37
```

Widening- Widening is a process where a datatype of lower scope is converted to a datatype of a higher scope. e.g- int to float and int to char

```
package Basics1;

import java.util.Scanner;

public class prog9 {


    public static void main(String[] args)
    {
        int intVal=87;
        System.out.println("Example Of Int to Float "+(float)intVal );
        char intVal2=91;
        System.out.println("Example Of Int to Char "+(char)intVal2 );
     }

}
```

Output

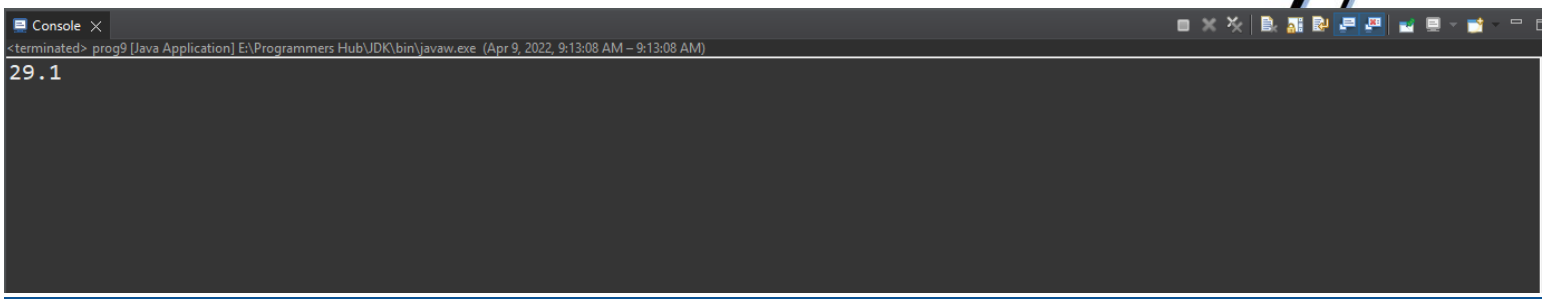


Why Widening Is Called Natural Typecasting.

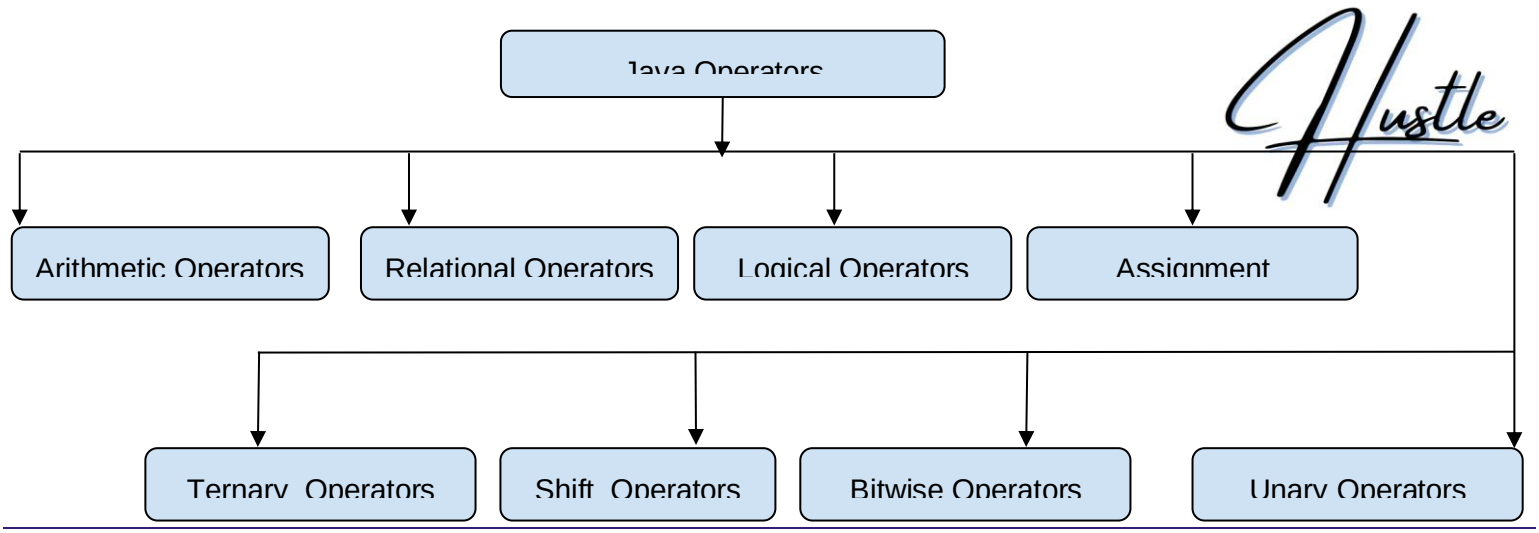
This is because Java can automatically typecast lower scope datatypes into higher scope datatypes.

```
package Basics1;

import java.util.Scanner;

public class prog9 {


    public static void main(String[] args)
    {
        int intVal=3; float floatVal=87.3f;
        System.out.println(floatVal/intVal);
     }

}
```

Output



Java Operators



Sl No	Operator Name	Operation	Symbol	Operation Demo	Explanation
1	**Arithmetic Operator**	Addition	+	2+3=5	Gives the sum of two numbers
		Subtraction	-	7-2 =5	Gives the difference between two numbers
		Multiplication	*	8*2=16	Gives the product of two numbers
		Division	/	9/3 =3	Gives the integral quotient when a number is divided by the other
		Modulation	%	5%2 =1	Gives the remainder when a number is divided by the other
2	**Relational Operator**	Greater Than	>	(9>8) =True (7>10) =False	Checks if a number is greater than the other or not
		Lesser Than	<	(5<6) =True (6<3) = False	Checks if a number is lesser than the other or not
		Greater Than & Equals To	>=	(6>=6) =True (6>=3) =True (6>=12) =False	Checks if a number is greater than or equal to the other number or not
		Lesser Than & Equals To	<=	(6<=6) =True (6<=8) =True (6<=3) =False	Checks if a number is lesser than or equal to the other number or not
		Equals To	==	(5==5) =True	Checks if two number are equal or not
		Not Equals To	!=	(6!=7) =True	Checks if two numbers are unequal or not

3	**Logical Operator**	Logical And	&&		Returns True if all the conditions are satisfied
		Logical Or	||		Returns True if any one of the given conditions are satisfied
		Logical Not	!		Reverts the present output
4	**Assignment Operator**	Equals To	=	a=5	Allocates 5 to variable a
		Plus Equals To	+=	a+=5	Adds 5 to previously stored value of variable a
		Minus Equals To	-=	a-=5	Subtracts 5 from previously stored value of variable a
		Multiply Equals To	*=	a*=5	Multiplies 5 to the previously stored value of variable a
		Divide Equals To	/=	a/=5	Divides the value previously stored in variable a by 5
5	**Unary Operators**	Pre Increment	++a	If a=7; b=++a; Then value of b will be 8 and value of a will be 8	First the value of variable will be incremented and then further operations will take place.
		Post Increment	a++	If a =7; b=a++; Then value if b will be 7 and value of a will be 8.	First the respective operation will take place and then the value of the variable will be incremented.
		Pre Decrement	- -a	If a=7; b=--a; Then value of b will be 6 and value of a will be 6	First the value of variable will be decremented and then further operations will take place.
		Post Decrement	a- -	If a =7; b=a--; Then value if b will be 7 and value of a will be 6.	First the respective operation will take place and then the value of the variable will be decremented. `
6		Bitwise And	&	If a = 7 and b = 6	And operation occurs between bit to bit

				$(7)_{10} = (111)_2$ $(6)_{10} = (110)_2$ a & b = $(111)_2$ & $(110)_2$ = $(110)_2 = (6)_{10}$	
	Bitwise Operators	Bitwise Or	|	If a = 7 and b = 6 $(7)_{10} = (111)_2$ $(6)_{10} = (110)_2$ a | b = $(111)_2$ | $(110)_2$ = $(111)_2 = (7)_{10}$	Or Operation occurs between bit to bit
		Bitwise Not	!	If a = 7 $(7)_{10} = (111)_2$!$(111)_2 = (000)_2$	Not operation occurs between bit to bit
		Bitwise Exclusive Or	^	If a = 7 and b = 6 $(7)_{10} = (111)_2$ $(6)_{10} = (110)_2$ a^ b = $(111)_2$ ^ $(110)_2$ = $(001)_2 = (7)_{10}$	Exclusive OR operation occurs between bit to bit
7	**Shift Operators**	Left Shift	<<	12 << 3 =12 * 2^3 = 96	a<< b = a* 2^b
		Right Shift	>>	12>>3 = 12 / 2^3 = 1	a>> b = a/ 2^b

Precedence And Associativity Of Operators

	Operator	Description	Associativity
P	( )	Parenthesis	Left To Right
	[ ]	Array Subscripts	
	.	Member Selection Via Object Name	
	->	Member Selection Via Pointer	
	a++ / a--	Post Fix Increment /Decrement	Right To Left
	++a / --a	Pre Fix Increment /Decrement	
	+ -	Unary Plus And Minus	
	!~	Logical Negation /Bitwise Compliment	
	(type)	Type Casting	
	*	Dereference	
	&	Address Of Operand	
	sizeof	Determines The Size In Bytes	
	& / *	Modulation,Division,Multiplication	Left To Right
	+ -	Addition ,Substraction	
	<< >>	Bitwise Left Shift , Bitwise Right Shift	
	< <=	Relational Less Than, Relational Less Than Equal To	
	> >=	Relational Greater Than, Relational Greater Than Equal To	
	== !=	Relational Equal To, Or Not Equal To	
	&	Bitwise And	

	^	Bitwise Exclusive Or	
	|	Bitwise Or	
P	&&	Logical And	
	||	Logical Or	
	?:	Ternary Conditional	Right To Left
	=	Assignment	
	+= -=	Addition Assignment, Subtraction Assignment	
	*= /=	Multiplication Assignment, Division Assignment	
	%= &=	Modulation Assignment, Bitwise Assignment	
	^= |=	Bitwise Exclusive Or , Bitwise Inclusive Or	
	<<= >>=	Bitwise Shift Left Operator , Shift Right Operator	

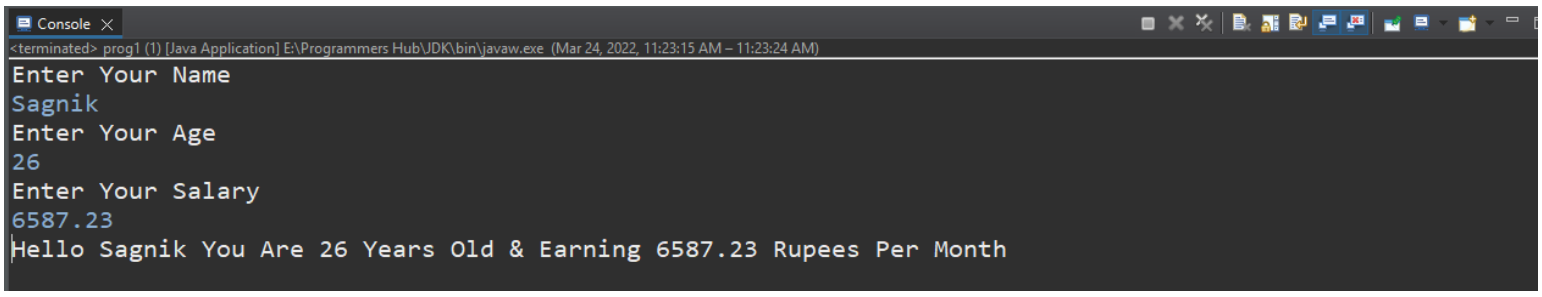
Java Input & Output

Input

Input in Java is taken using **Scanner Class**

```
package Basics1;

import java.util.Scanner;

public class prog1
{
    public static void main(String[] args) {

        Scanner sc = new Scanner (System.in);

        int age; String name; float salary;

        System.out.println("Enter Your Name");
        name= sc.nextLine();
        System.out.println("Enter Your Age");
        age= sc.nextInt();
        System.out.println("Enter Your Salary");
        salary = sc.nextFloat();


        System.out.println("Hello "+name+" You Are "+age+" Years Old & Earning "+salary+"
Rupees Per Month");

}
}
```

Output



Inputting Methods Of Datatypes

Sl No	Datatype	Inputting Code	Function
1	int	sc.nextInt( )	Input an integer
2	float	sc.nextFloat( )	Inputs a float

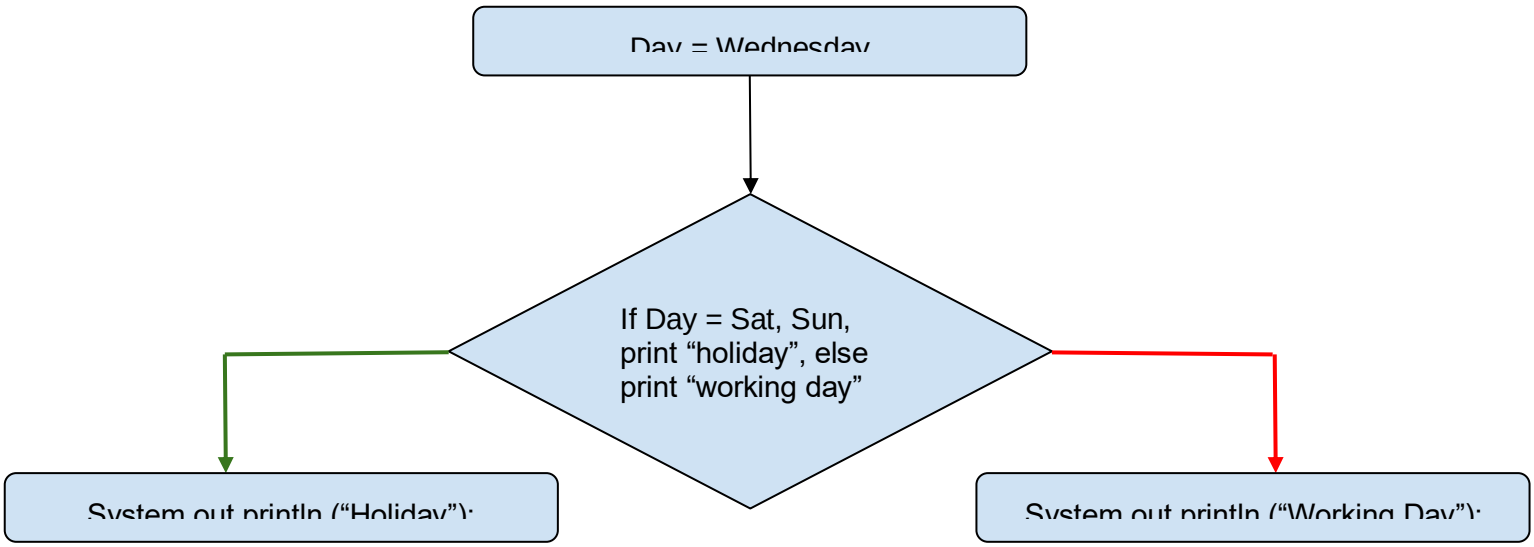
3	short	sc.nextShort( )	Inputs a short
4	byte	sc.nextByte ( )	Inputs a byte
5	boolean	sc.nextBoolean( )	Inputs a boolean
6	char	sc.next.charAt(0)	Inputs a unicode character
7	double	sc.nextDouble( )	Inputs a double
8	long	sc.nextLong( )	Inputs a long
9	string	sc.nextLine( )	Inputs a string

Java Keywords

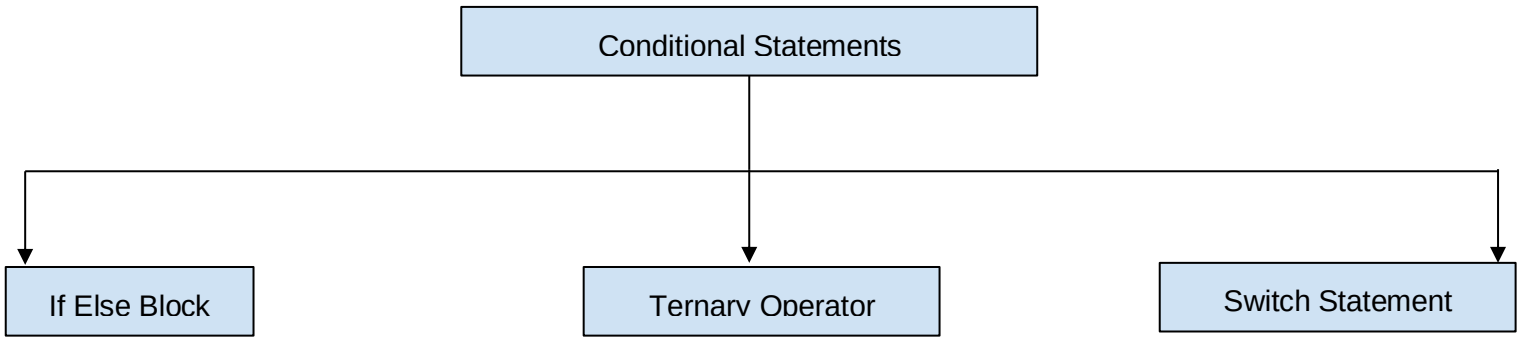
abstract	continue	for	new	switch
assert***	default	while	package	synchronized
boolean	do	if	private	this
break	double	implements	protected	throw
byte	else	import	public	throws
case	enum****	instanceof	return	transient
catch	extends	int	short	try
char	final	interface	static	void
class	finally	long	strictfp**	volatile
const*	float	native	super	

Java Conditional Statements

Conditional Statements are used to control the flow of the program based on satisfaction of certain conditions



In Java, there are three types of conditional statements

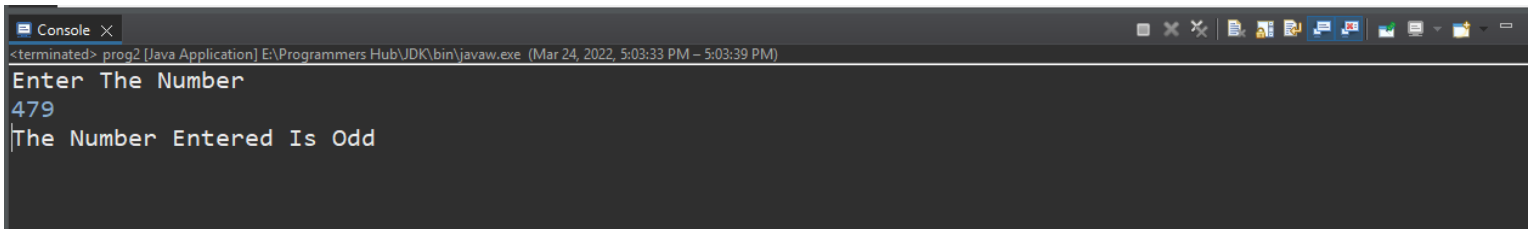


If Else Block

Problem 1:- Write A Program To Check Whether A Given Number Is Even Or Odd

```java
package Basics1;

import java.util.Scanner;

public class prog2 {

    public static void main(String[] args) {

        Scanner sc = new Scanner(System.in);
        System.out.println("Enter The Number ");
        int num = sc.nextInt();

        if(num%2==0)
        {
            System.out.println("The Number Entered Is Even");
        }
        else
        {
            System.out.println("The Number Entered Is Odd");
        }
    }

}
```

Output



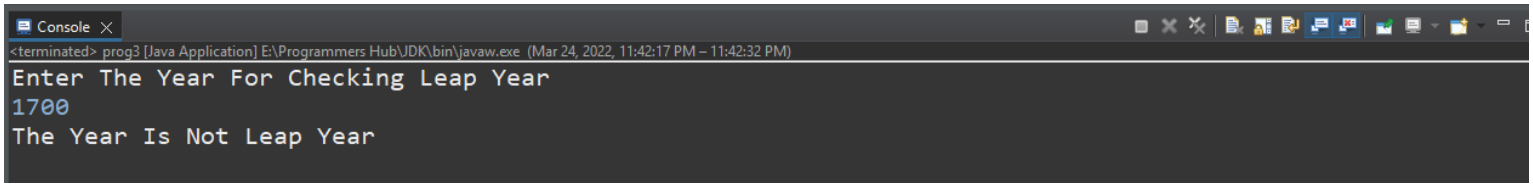
Problem 2:- Write A Program To Check Whether A Given Year Is Leap Year Or Not

```java
package Basics1;

import java.util.Scanner;

public class prog3 {

    public static void main(String[] args) {

        Scanner sc =  new Scanner(System.in);
        System.out.println("Enter The Year For Checking Leap Year");
        int year = sc.nextInt();

        if(year%100==0)
        {
            if(year%400==0)
            System.out.println("The Year Is Leap Year");
            else
            {
                System.out.println("The Year Is Not Leap Year");
            }
        }
        else
        {
            if(year%4==0)
```

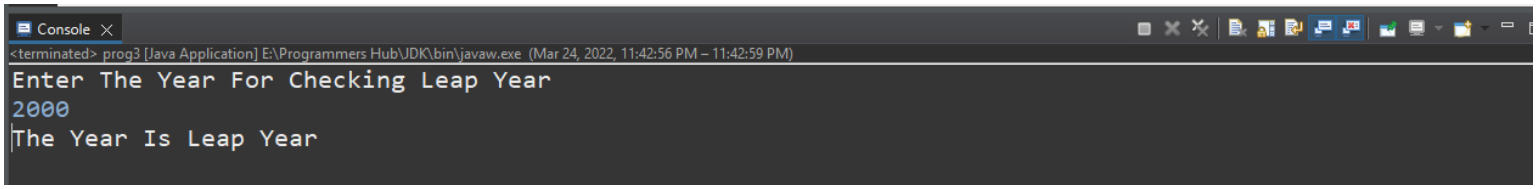
```
            {
                System.out.println("The Year Is Leap Year");
            }
            else
            {
                System.out.println("The Year Is Not Leap Year");
            }
        }
    }


}
```

Output

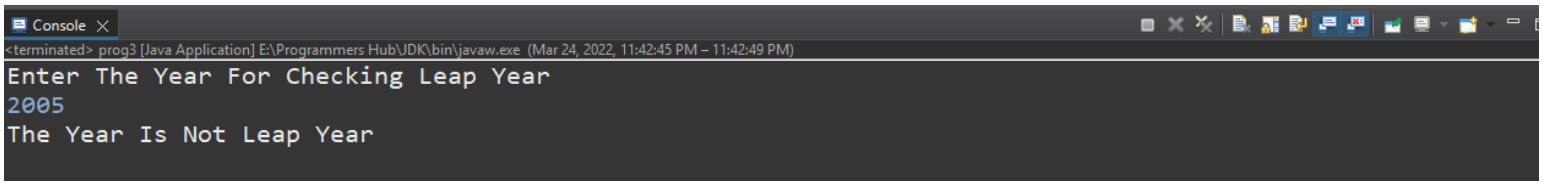
Case-1:- Centurial Non Leap Year



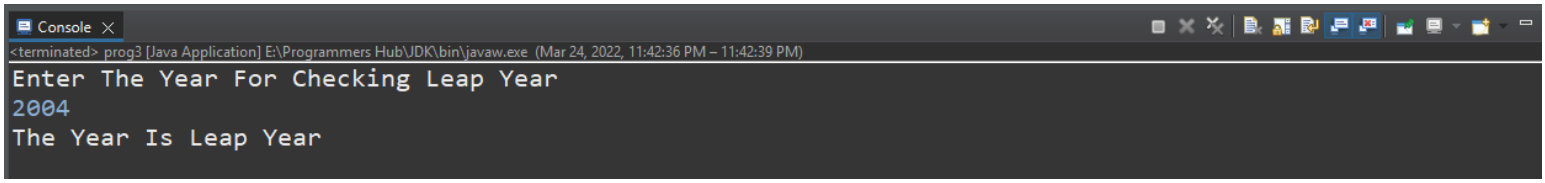
Case-2:-Centurial Leap Year



Case 3:- Non Centurial Non Leap Year



Case 4:- Non Centurial Leap Year



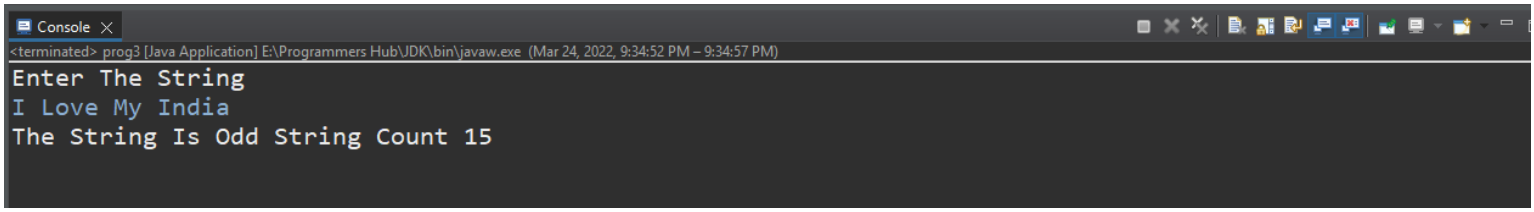
Ternary Operator

Ternary operator is can be used where there are no complex cases of decision making which can involve nested if-else blocks

Write a program to check if a given string is odd or even

```
package Basics1;

import java.util.Scanner;

public class prog3 {

    public static void main(String[] args) {

        Scanner sc =  new Scanner(System.in);
        System.out.println("Enter The String");
        String sample = sc.nextLine();
        int count = sample.length();

        String res=(count%2==0)?"The String Is Even":"The String Is Odd";
        System.out.println(res+" String Count "+count);
    }


}
```

Output



Switch Statement

Switch Statement is used where there are multiple options that can be chosen by the user

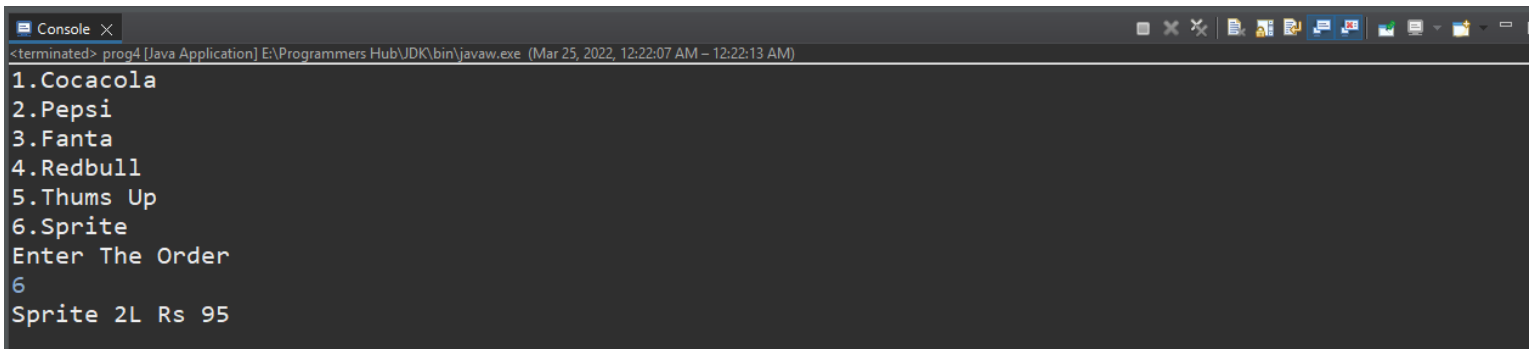
Write A Program For A Simple Vending Machine

```
package Basics1;
import java.util.Scanner;
public class prog4 {

    public static void main(String[] args) {
        Scanner sc = new Scanner(System.in);
        System.out.println("1.Cocacola");
        System.out.println("2.Pepsi");
        System.out.println("3.Fanta");
        System.out.println("4.Redbull");
        System.out.println("5.Thums Up");
        System.out.println("6.Sprite");

        System.out.println("Enter The Order");
        int order=sc.nextInt();
        switch(order)
        {
```

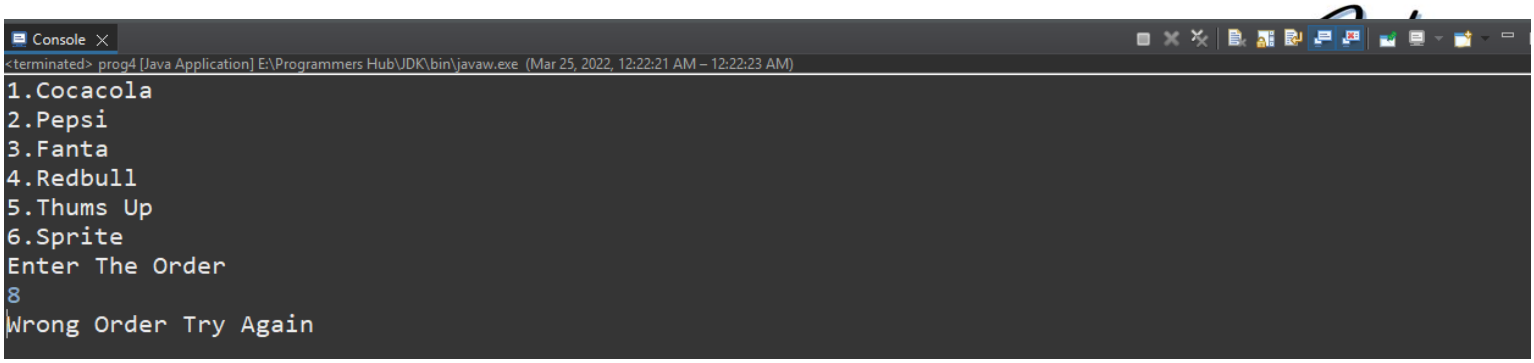
```
        case 1:
        {
            System.out.println("Cocacola 2L Rs 85");
            break;
        }
        case 2:
        {
            System.out.println("Pepsi 2L Rs 80");
            break;
        }
        case 3:
        {
            System.out.println("Fanta 2L Rs 95");
            break;
        }
        case 4:
        {
            System.out.println("Redbull 2L Rs 120");
            break;
        }
        case 5:
        {
            System.out.println("Thums Up 2L Rs 90");
            break;
        }
        case 6:
        {
            System.out.println("Sprite 2L Rs 95");
            break;
        }
         default:
        {
            System.out.println("Wrong Order Try Again");
        }
    }
}
}
```

Output

Case 1:- Putting A Legit Option



Case 2:- Putting A Wrong Option



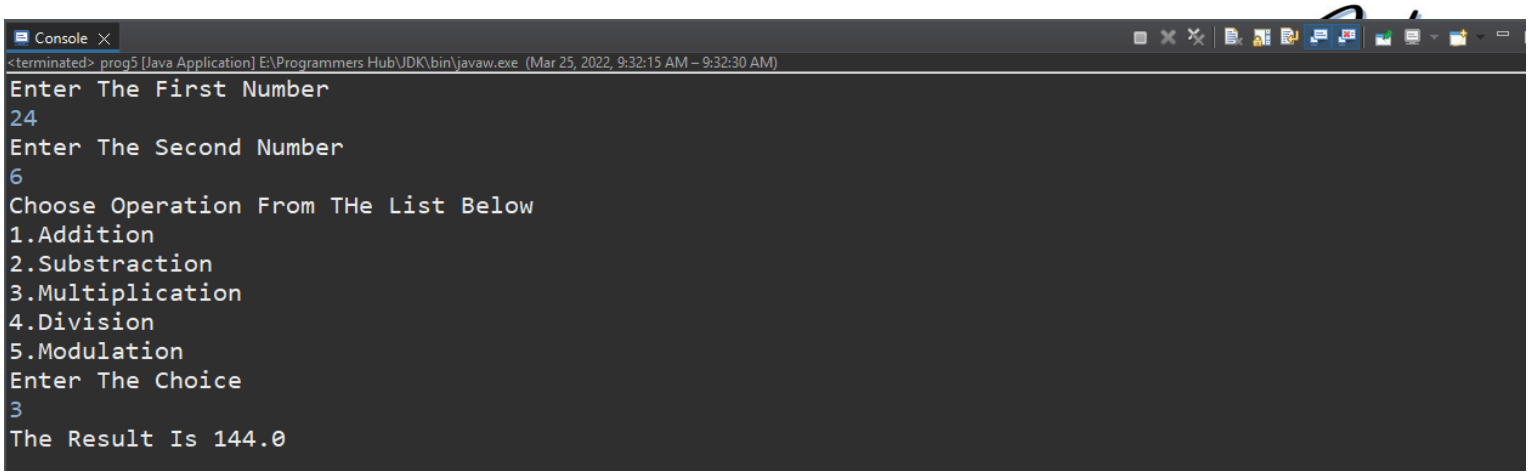
We have to use break statement after each case as it prevents the next cases from getting auto triggered.

Enhanced Switch Structure

Write A Program To Build A Simple Calculator With Enhanced Switch Syntax

```
package Basics1;
import java.util.Scanner;
public class prog5 {

    public static void main(String[] args)
    {
        Scanner sc = new  Scanner(System.in);
        System.out.println("Enter The First Number");
        float num1= sc.nextFloat();
        System.out.println("Enter The Second Number");
        float num2=sc.nextFloat();

        System.out.println("Choose Operation From THe List Below");

        System.out.println("1.Addition");
        System.out.println("2.Substraction");
        System.out.println("3.Multiplication");
        System.out.println("4.Division");
        System.out.println("5.Modulation");


        System.out.println("Enter The Choice");
        int choice = sc.nextInt();

        switch(choice)
        {
        case 1->System.out.println("The Result Is "+(num1+num2));
        case 2->System.out.println("The Result Is "+(num1-num2));
        case 3->System.out.println("The Result Is "+(num1*num2));
        case 4->System.out.println("The Result Is "+(num1/num2));
        case 5->System.out.println("The Result Is "+(num1%num2));
        default->System.out.println("Wrong Choice Try Again");
        }
    }
}
```

Output:-



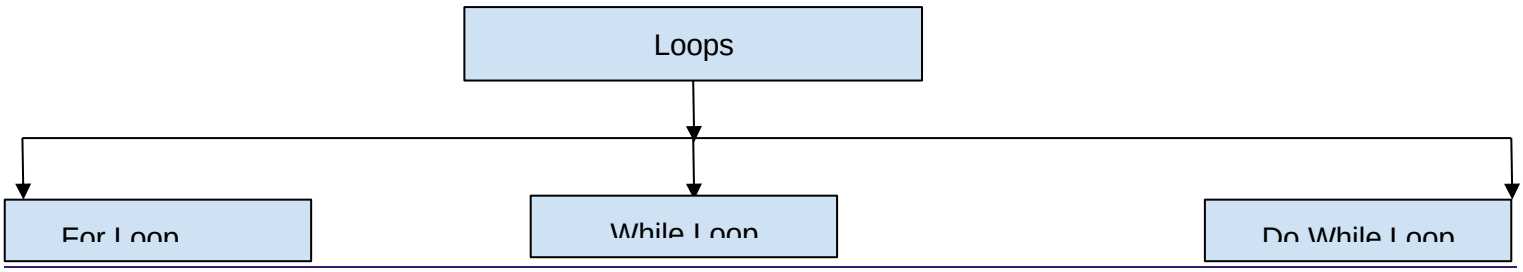
Switch Cases With Range In Java

```
package Basics1;
import java.util.Scanner;
public class prog6
{
    public static char SeatNumber(int tokenNo)
    {
        if(tokenNo>=100 && tokenNo<200)
        {
            return 'a';
        }
        if(tokenNo>=200 && tokenNo<300)
        {
            return 'b';
        }
        if(tokenNo>=300 && tokenNo<400)
        {
            return 'c';
        }
        return '\0';


    }
    public static void main(String[] args)
    {
        Scanner sc = new Scanner(System.in);
        System.out.println("Enter Your Token Number");
        int  tokenNo=sc.nextInt();
        char switchVar=SeatNumber(tokenNo);

        switch(switchVar)
        {
        case 'a'->System.out.println("You Are Allocated In Hall Number A1");
        case 'b'->System.out.println("You Are Allocated In Hall Number B1");
        case 'c'->System.out.println("You Are Allocated In Hall Number C1");
        default->System.out.println("Invalid Token Number ");

        }
    }

}
```

Output

```
Console
<terminated> prog6 [Java Application] E:\Programmers Hub\JDK\bin\javaw.exe (Mar 25, 2022, 10:20:15 AM – 10:20:27 AM)
Enter Your Token Number
324
You Are Allocated In Hall Number C1
```

Java Loops



For Loops

The Syntax For For Loops Is As Follows

```
for( initialization condition ; termination condition ; increment/decrement)
{
Loop Body
}
```

Write A Program To Print All Even Numbers Between 1 to 40

```
package Basics1;
public class prog7 {

    public static void main(String[] args) {

        for(int i=0;i<=40;i++)
        {
            if(i%2==0)
            System.out.print(i+" ");
        }
    }

}
```

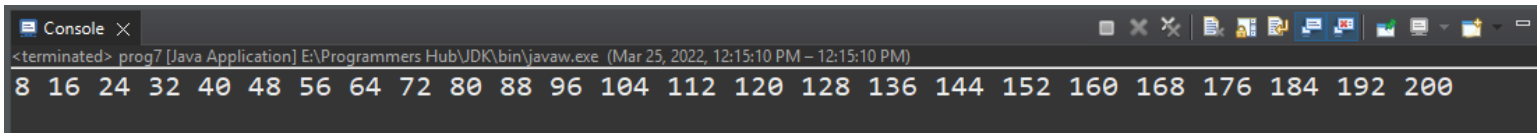
Output

```
Console
<terminated> prog7 [Java Application] E:\Programmers Hub\JDK\bin\javaw.exe (Mar 25, 2022, 12:17:39 PM – 12:17:39 PM)
0 2 4 6 8 10 12 14 16 18 20 22 24 26 28 30 32 34 36 38 40
```

Write A Program To Print All Numbers Between 1 to 200 which are multiples of 8

```java
package Basics1;

public class prog7 {

    public static void main(String[] args) {

        for(int i=8;i<=200;i+=8)
        {
            System.out.print(i+" ");
        }
    }

}
```

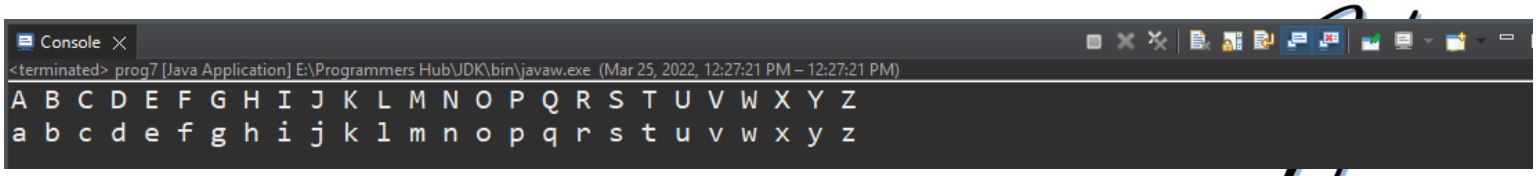
Output



Write A Program To Print Alphabets From A to Z in both Uppercase And Lowercase

```java
package Basics1;
public class prog7 {

    public static void main(String[] args) {

        for(int i=65;i<91;i++)
        {

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

        System.out.println();

        for(int i=97;i<123;i++)
        {
            System.out.print((char)i+" ");
        }
    }

}
```

Output



While Loop

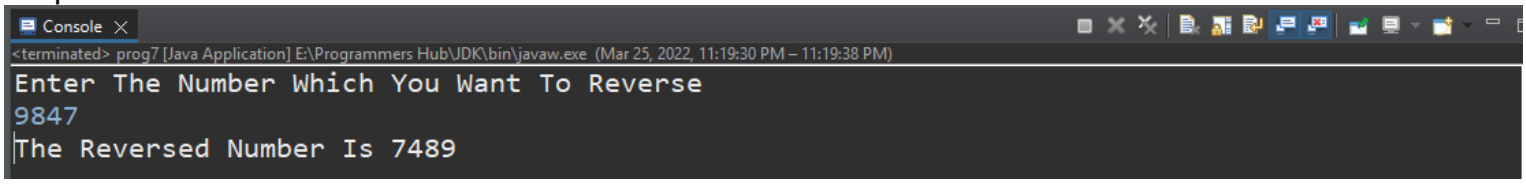
The syntax for while loop is

```
while(condition )
{
Loop Body
Increment/Decrement
}
```

Write A Program To Reverse A Given Number

```
package Basics1;
import java.util.Scanner;

public class prog7 {

    public static void main(String[] args)
    {
        Scanner sc = new Scanner(System.in);

        System.out.println("Enter The Number Which You Want To Reverse");
        int num= sc.nextInt();

        int rev = 0;
        while(num!=0)
        {
            int dig=num%10;
            rev=(rev*10)+dig;
            num/=10;
        }

        System.out.println("The Reversed Number Is "+rev);
```

Output



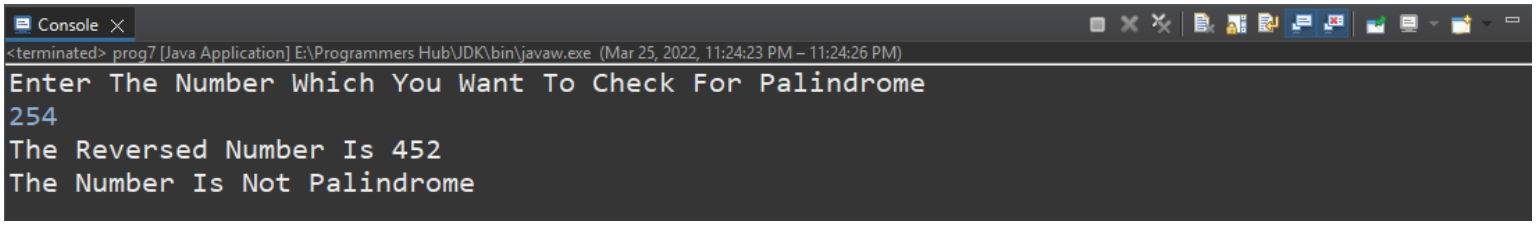
Write A Program To Check If A Given Number Is Palindrome Or Not

```
package Basics1;
```

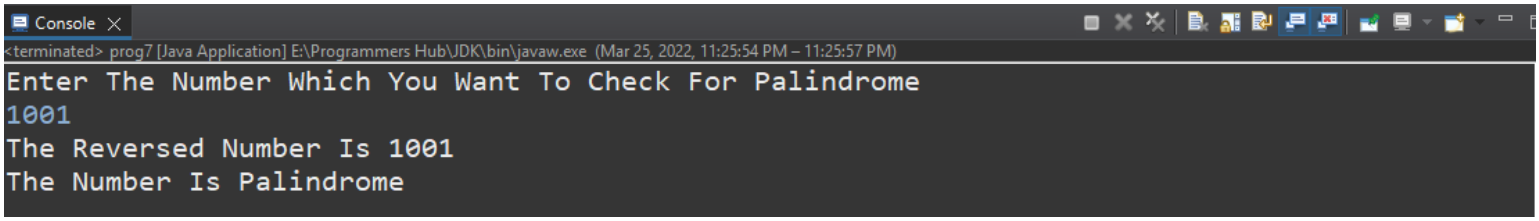
```java
import java.util.Scanner;

public class prog7 {

    public static void main(String[] args)
    {
        Scanner sc = new Scanner(System.in);

        System.out.println("Enter The Number Which You Want To Check For Palindrome");
        int num= sc.nextInt();

        int rev = 0;
        int copy=num;
        while(num!=0)
        {
            int dig=num%10;
            rev=(rev*10)+dig;
            num/=10;
        }

        System.out.println("The Reversed Number Is "+rev);

        if(copy==rev)
        System.out.println("The Number Is Palindrome");
        else
        System.out.println("The Number Is Not Palindrome");

}
}
```

Output

Case 1- Non Palindrome



Case 2- Palindrome



Do While Loop

```
do
{
Loop Body
Increment/Decrement
}
while(condition )
```

Write A Program For A Basic User Authentication System

```
package Basics1;
 import java.util.Scanner;
public class prog7 {

    public static void main(String[] args)
    {
        Scanner sc = new Scanner(System.in);
        String password ="sagnik@123#";

        int attempts=3;



        do
        {
                System.out.println("Enter Your Password ");
                String passwordInput=sc.nextLine();

                if(passwordInput.equals(password))
                {
                    System.out.println("Access Granted !!");
                    break;
                }
                else
                {
                    attempts--;
                    if(attempts==0)
                    {
           System.out.println("You Made Maximum Invalid Attempts! Contact System Admin");
                        break;
                    }
                    else
                    {
           System.out.println("Access Denied !! You Have "+attempts+ " more attempts
left");

                    }
                }

        }
        while(true);

}
}
```

Output

```
Console
<terminated> prog7 [Java Application] E:\Programmers Hub\JDK\bin\javaw.exe (Apr 23, 2022, 12:26:38 AM – 12:27:03 AM)
Enter Your Password
sagnik@123
Access Denied !! You Have 2 more attempts left
Enter Your Password
sagnik@1234
Access Denied !! You Have 1 more attempts left
Enter Your Password
sagnik@123#
Access Granted !!
```

Nested Loops

When there are one or more inner loops present inside an outer loop, then it is called a Nested Loop

Write A Program To Print The Following Pattern

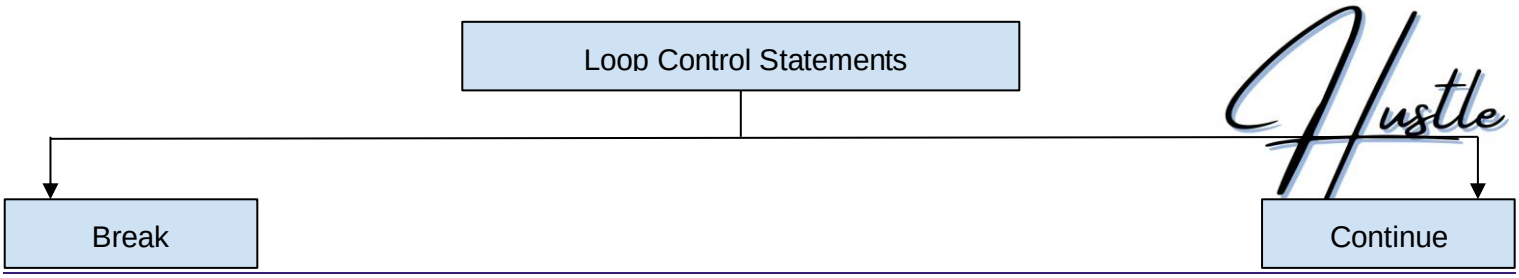
```
A
A B
A B C
A B C D
A B C D E
A B C D E F
```

```java
package Basics1;

public class prog8 {

    public static void main(String[] args)
    {
        int i,j;

        for(i=0;i<6;i++)
        {
            for(j=0;j<=i;j++)
            {

                System.out.print((char)(65+j)+" ");
            }
            System.out.println();
        }

    }

}
```

Output

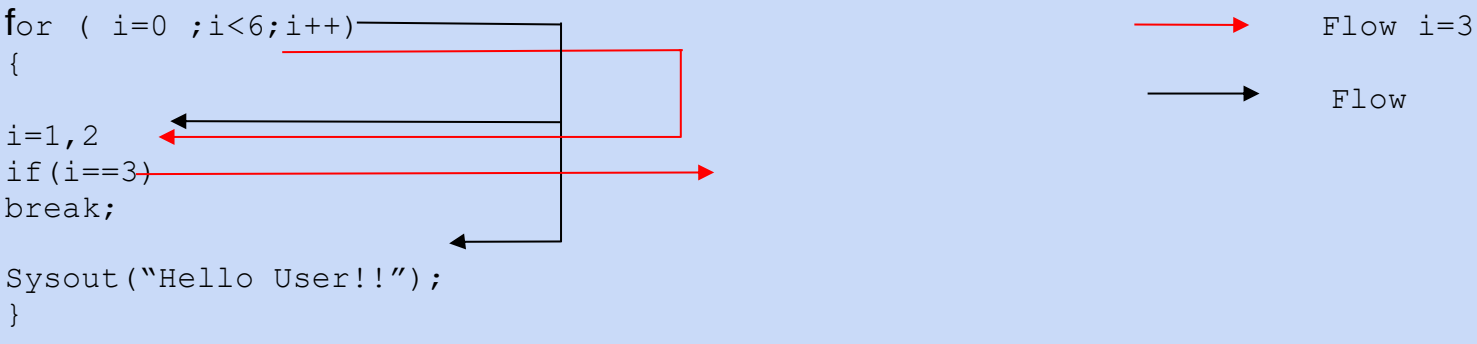
```
Console
<terminated> prog8 [Java Application] E:\Programmers Hub\JDK\bin\javaw.exe (Mar 27, 2022, 9:24:41 AM – 9:24:41 AM)
A
A B
A B C
A B C D
A B C D E
A B C D E F
```

[Java Control Statements]



Break Statement

Break Statement is used to terminate the loop when certain conditions are satisfied



Write A Program To Print All Prime Numbers Within A Given Range

```
package Basics1;

import java.util.Scanner;

public class prog8 {

    public static void main(String[] args)
    {
        int startingNo ,endingNo;

        Scanner sc = new Scanner(System.in);
        System.out.println("Enter The Starting Number ");
        startingNo = sc.nextInt();
        System.out.println("Enter The Ending Number ");
        endingNo = sc.nextInt();


        int i,j;

        for(i=startingNo;i<=endingNo;i++)
        {
            boolean isPrime =true;

            for(j=2;j<i;j++)
            {
                if(i%j==0)
                {
                    isPrime=false;
                    break;
                }


            }
```

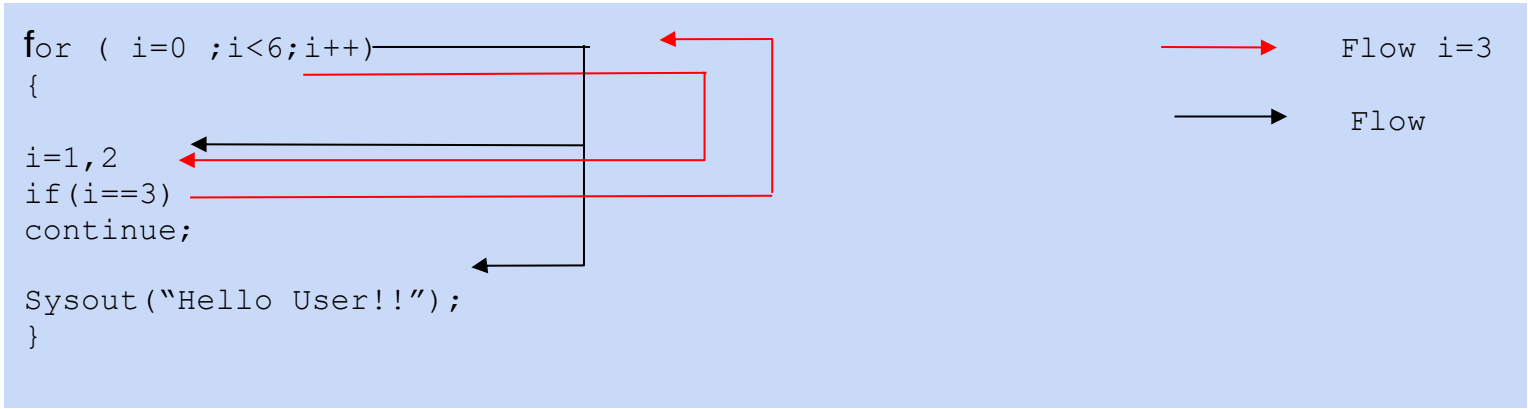
```
            if(isPrime)
                System.out.print(i+" ");
        }


    }

}
```

Output

```
Console ×
<terminated> prog8 [Java Application] E:\Programmers Hub\JDK\bin\javaw.exe (Mar 27, 2022, 10:27:46 AM – 10:27:51 AM)
Enter The Starting Number
1
Enter The Ending Number
24
1 2 3 5 7 11 13 17 19 23
```

Continue Statement

Continue Statement is used to ignore all following statements written inside the loop body for that particular iteration and is triggered when certain conditions are satisfied



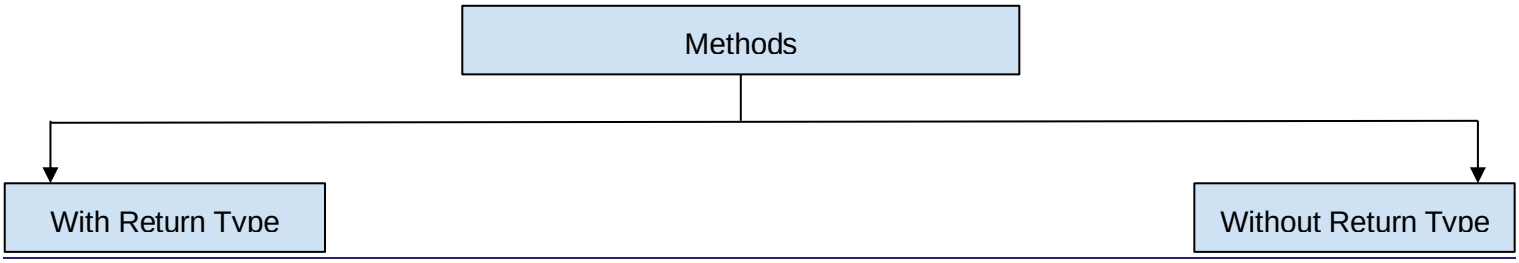

```
package Basics1;
import java.util.Scanner;
public class prog8 {

    public static void main(String[] args)
    {
        int i;
        for(i=1;i<=10;i++)
        {
            if(i>4 && i<8)
            {
                System.out.print(i);
                System.out.println();
                continue;
            }
            System.out.println(i+" Hello User!!");
        }

    }

}
```

Output

```
Console
<terminated> prog8 [Java Application] E:\Programmers Hub\JDK\bin\javaw.exe  (Mar 27, 2022, 10:53:26 AM – 10:53:27 AM)
1 Hello User!!
2 Hello User!!
3 Hello User!!
4 Hello User!!
5
6
7
8 Hello User!!
9 Hello User!!
10 Hello User!!
```

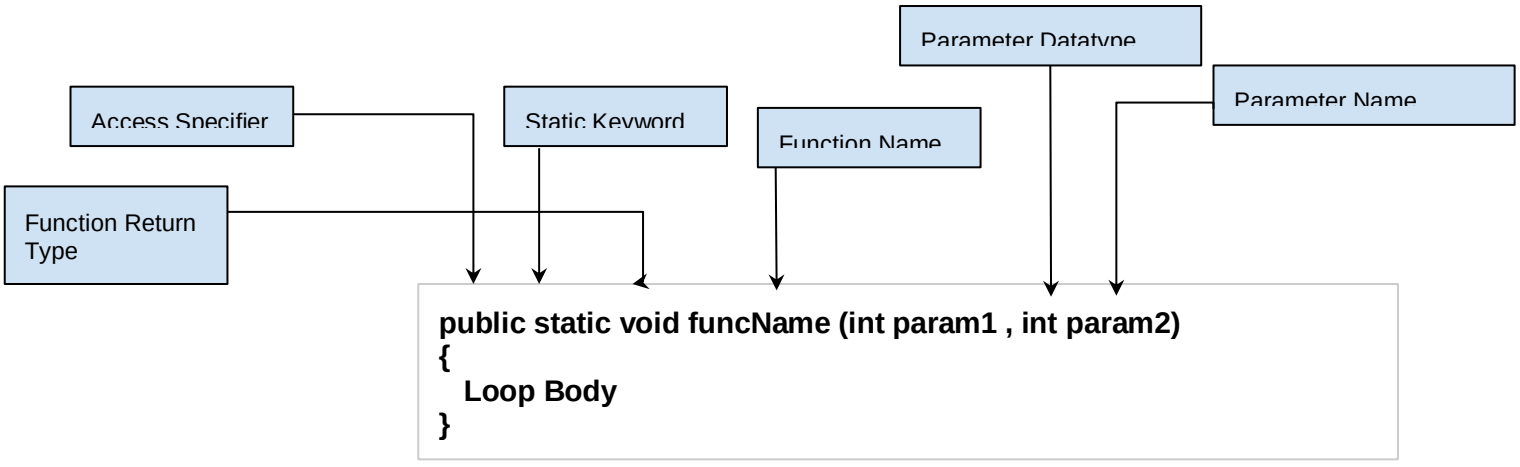
Java Methods

Methods are blocks of code which are designated to perform certain tasks by receiving some parameters as its arguments



Methods Without Return Type

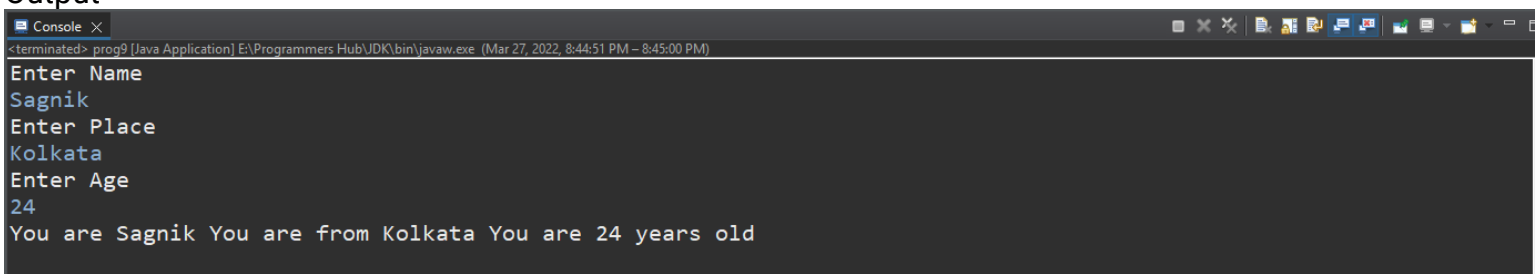
Methods which are not supposed to return anything to its parent method, from which it is called are called Methods without return type. They are denoted by using void keyword



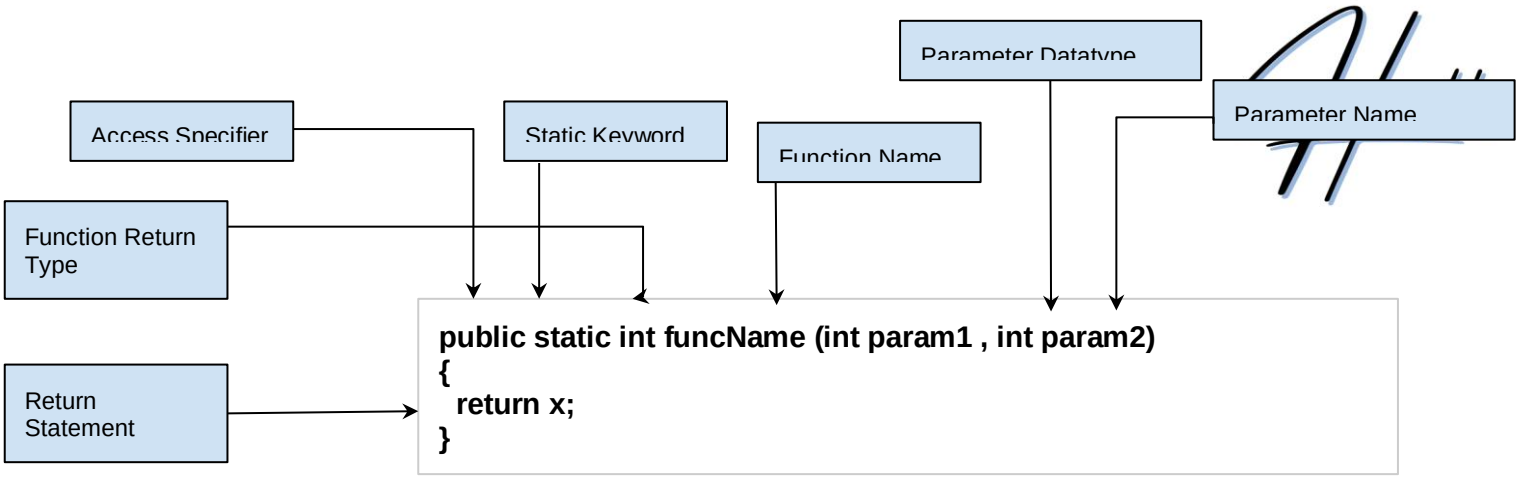

```
package Basics1;

import java.util.Scanner;
```

```java
public class prog9 {

    public static void displayData(String name,String place,int age)
    {

    System.out.println("You are "+name+" You are from "+place+" You are "+age+" yearsold");

    }

    public static void main(String[] args)
    {
        Scanner sc = new Scanner(System.in);
        System.out.println("Enter Name");
        String name= sc.nextLine();

        System.out.println("Enter Place");
        String place= sc.nextLine();

        System.out.println("Enter Age");
        int age= sc.nextInt();

        displayData(name,place,age);
    }

}
```

Output



Methods With Return Type



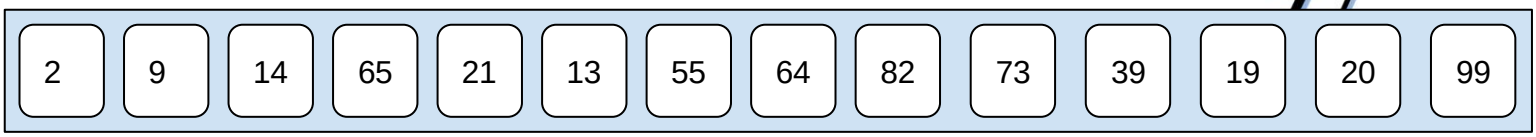

```java
package Basics1;

import java.util.Scanner;

public class prog9 {

    public static String sentence(String name,String Place, int age)
    {
        return name+" "+"You Are From "+Place+" And Is "+age+" Years Old";
    }

    public static void main(String[] args)
    {
        Scanner sc = new Scanner(System.in);
        System.out.println("Enter Name");
        String name= sc.nextLine();
        System.out.println("Enter Place");
        String place= sc.nextLine();
        System.out.println("Enter Age");
        int age= sc.nextInt();
        System.out.println(sentence(name,place,age));
    }

}
```

Output

```
Console
<terminated> prog9 [Java Application] E:\Programmers Hub\JDK\bin\javaw.exe (Mar 31, 2022, 8:46:08 PM – 8:46:21 PM)
Enter Name
Sagnik
Enter Place
Kolkata
Enter Age
25
Sagnik You Are From Kolkata And Is 25 Years Old
```

Introduction To Arrays

Arrays are data structures which store elements of same datatype in a contiguous memory allocation. Array in Java is index based, which means the first element is stored in index '0', second element at '1' and so on. The final element of array is stored at index 'size-of-array - 1'.



Array

Creation And Initialisation Of Arrays

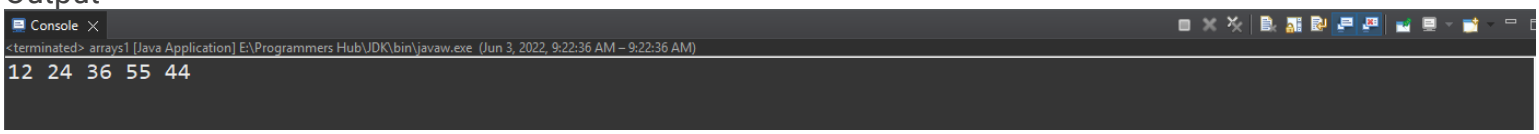
The syntax for creation of arrays is as follows.

```
package Arrays1;

public class arrays1 {

    public static void main(String[] args) {

        int[] arrayName = {12,24,36,55,44};

    }

}
```

The syntax for displaying the array is as follows

```
package Arrays1;

public class arrays1 {

    public static void main(String[] args) {

        int[] arrayName = {12,24,36,55,44};
        for(int i=0;i<arrayName.length;i++)
        {
            System.out.print(arrayName[i]+" ");
        }

    }

}
```

Output



Size Of An Array

Whenever we create an Array, we need to declare beforehand the size which it will occupy. This size cannot be changed.

```java
class arrays
{
    public static void main(String[] args)
    {
        int[] array1 = new int[10];
    }
}
```

What Happens When We Insert A Negative Value As Array Size?

```java
class arrays
{
    public static void main(String[] args)
    {
        int[] array1 = new int[-1];
    }
}
```

Output

```
PROBLEMS 2  OUTPUT  DEBUG CONSOLE  TERMINAL                                  bash

user@r1pc0rd-1:/media/user/Volume 1/Programmer's Hub/Source Codes/Java/Basics1$ javac arrays.java
user@r1pc0rd-1:/media/user/Volume 1/Programmer's Hub/Source Codes/Java/Basics1$ java arrays
Exception in thread "main" java.lang.NegativeArraySizeException: -1
        at arrays.main(arrays.java:5)
user@r1pc0rd-1:/media/user/Volume 1/Programmer's Hub/Source Codes/Java/Basics1$
```

The program compiles successfully, however during runtime, it throws a Negative Array Size Exception

What Happens If We Put A Character As Array Size?

```java
class arrays
{
    public static void main(String[] args)
    {
        int[] array1 = new int['a'];
        System.out.println(array1.length);
    }
}
```

Output

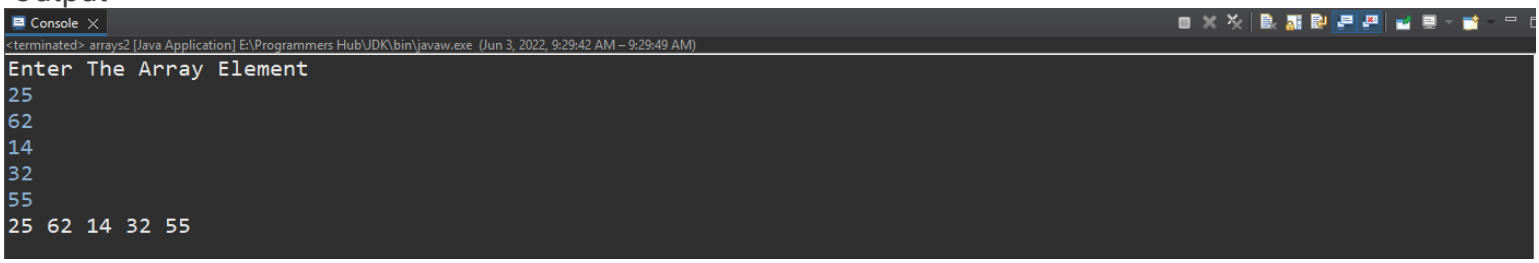
```
PROBLEMS 1  OUTPUT  DEBUG CONSOLE  TERMINAL                                  bash

user@r1pc0rd-1:/media/user/Volume 1/Programmer's Hub/Source Codes/Java/Basics1$ javac arrays.java
user@r1pc0rd-1:/media/user/Volume 1/Programmer's Hub/Source Codes/Java/Basics1$ java arrays
97
user@r1pc0rd-1:/media/user/Volume 1/Programmer's Hub/Source Codes/Java/Basics1$
```

Traversing Across An Array

Write a program to declare an array of size 5, and traverse across it and insert elements of your choice. Finally, display the array.

```
package Arrays1;

import java.util.Scanner;

public class arrays2 {

    public static void main(String[] args) {
        int[] array = new int[5];
        int i;

        Scanner sc = new Scanner(System.in);
        System.out.println("Enter The Array Element");
        for(i=0;i<array.length;i++)
        {
            array[i]=sc.nextInt();
        }

        for(i=0;i<array.length;i++)
        {
            System.out.print(array[i]+" ");
        }


    }

}
```

Output



Accessing Elements In The Array

Initialize an array with random values and then modify the values present at even indexes by increasing it by 1.

```
package Arrays1;

import java.util.Scanner;

public class arrays2 {

    public static void main(String[] args) {
        int[] array = new int[5];
        int i;

        Scanner sc = new Scanner(System.in);
        System.out.println("Enter The Array Element");
        for(i=0;i<array.length;i++)
        {
            array[i]=sc.nextInt();
        }
```

```
        for(i=0;i<array.length;i++)
        {
            if(i%2!=0)
            {
                array[i]+=1;
                System.out.print(array[i]+" ");
            }
            else
            System.out.print(array[i]+" ");
        }


    }

}
```

Output

```
Console
<terminated> arrays2 [Java Application] E:\Programmers Hub\JDK\bin\javaw.exe (Jun 3, 2022, 10:08:26 AM – 10:08:33 AM)
Enter The Array Element
14
25
10
87
95
14 26 10 88 95
```

Problem- **Geeks For Geeks**

[Find Out The Peak Elements]

Given an array of size n. You have to return an array consisting of all the peak elements present in the array. A peak element is an element present in the array which is greater than its adjacent elements. Also, if the first element is bigger than the immediate next element or the final element is bigger the preceding element, then those elements are considered as peak elements too.

Solution-

```
package Arrays1;

import java.util.Scanner;

public class arrays4 {

    public static int[] returnPeakElementArray(int[] arr)
    {
        int n = arr.length;
        int peakArray[] = new int[n/2];
        int k=0;



           if(arr[0]>arr[1])
           {
               peakArray[k]=arr[0];
                k++;
           }
           if(arr[n-1]>arr[n-2])
            {
```

```
                peakArray[k]=arr[n-1];
                k++;
            }
          int counter =1;
          while(counter!=n-1)
          {
            if((arr[counter]>arr[counter-1] )&& (arr[counter]>arr[counter+1]) )
            {
                peakArray[k]=arr[counter];
                   k++;
            }

            counter++;
           }

          return peakArray;
    }

    public static void main(String[] args) {

        Scanner sc = new Scanner(System.in);
        System.out.println("Enter The Input Size");
        int inputSize= sc.nextInt();
        System.out.println("Enter The Array Elements");
        int i;int arr[] = new int[inputSize];

        for(i=0;i<inputSize;i++)
            arr[i]=sc.nextInt();
        System.out.println("The Array Looks Like");
        for(i=0;i<inputSize;i++)
            System.out.print(arr[i]+" ");

        int[] peakArray=returnPeakElementArray(arr);

        System.out.println("The Peak Element Array Is ");
        for(i=0;i<peakArray.length;i++)
            System.out.print(peakArray[i]+" ");


    }
}
```

Output

```
Console
<terminated> arrays4 [Java Application] E:\Programmers Hub\JDK\bin\javaw.exe (Jun 3, 2022, 8:16:46 PM – 8:16:55 PM)
Enter The Input Size
5
Enter The Array Elements
16
7
54
32
15
The Array Looks Like
16 7 54 32 15 The Peak Element Array Is
16 54
```

Problem- **Geeks For Geeks**

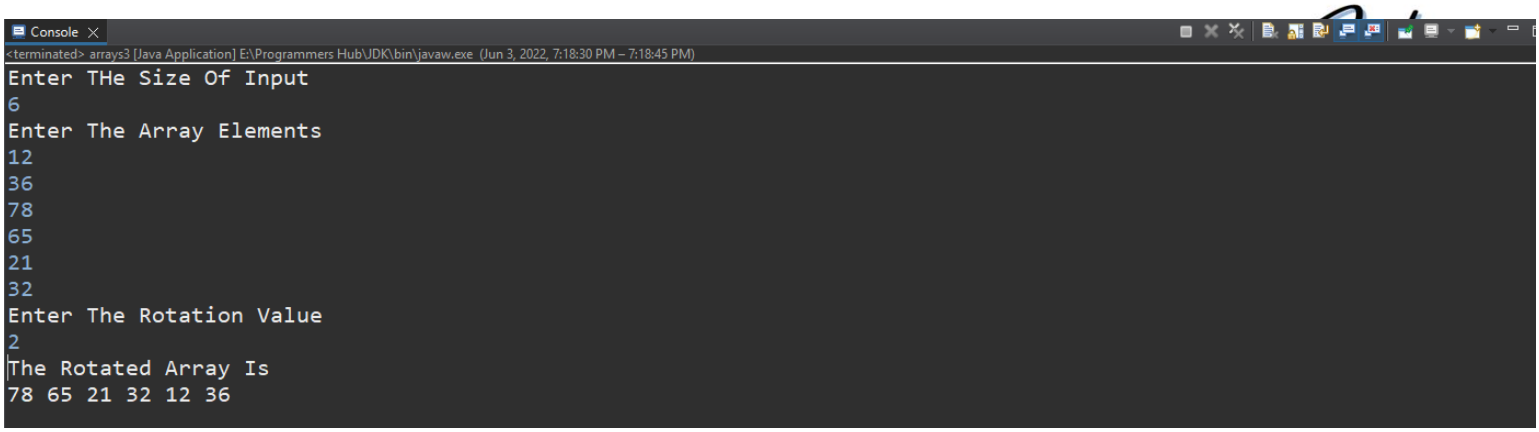
[Cyclically Rotate Array By n Elements]

Given an array of size n. You have to cyclically rotate in a counter clockwise direction by a value of d, which is inputted by user.

Solution-

```
package Arrays1;

import java.util.Scanner;

public class arrays3 {

    public static void rotateArray(int[] array,int rotationVal)
    {
         int i=0,counter=0;

           for(counter=0;counter<rotationVal;counter++){
               int temp=array[0];
               for(i=0;i<array.length-1;i++)
               {
                   array[i]=array[i+1];
               }
               array[array.length-1]=temp;

        }

           System.out.println("The Rotated Array Is ");

           for(i=0;i<array.length;i++)
               System.out.print(array[i]+" ");
    }

    public static void main(String[] args) {

        Scanner sc = new Scanner(System.in);
        System.out.println("Enter THe Size Of Input");
        int inputSize = sc .nextInt();
        int [] array = new int[inputSize];
        System.out.println("Enter The Array Elements");
        for(int i=0;i<inputSize;i++)
            array[i]=sc.nextInt();
        System.out.println("Enter The Rotation Value");
        int rotationVal = sc.nextInt();

        rotateArray(array, rotationVal);

    }

}
```

Output



Searching In Arrays

There are two types of searching methods in arrays.

1. Linear Search
2. Binary Search

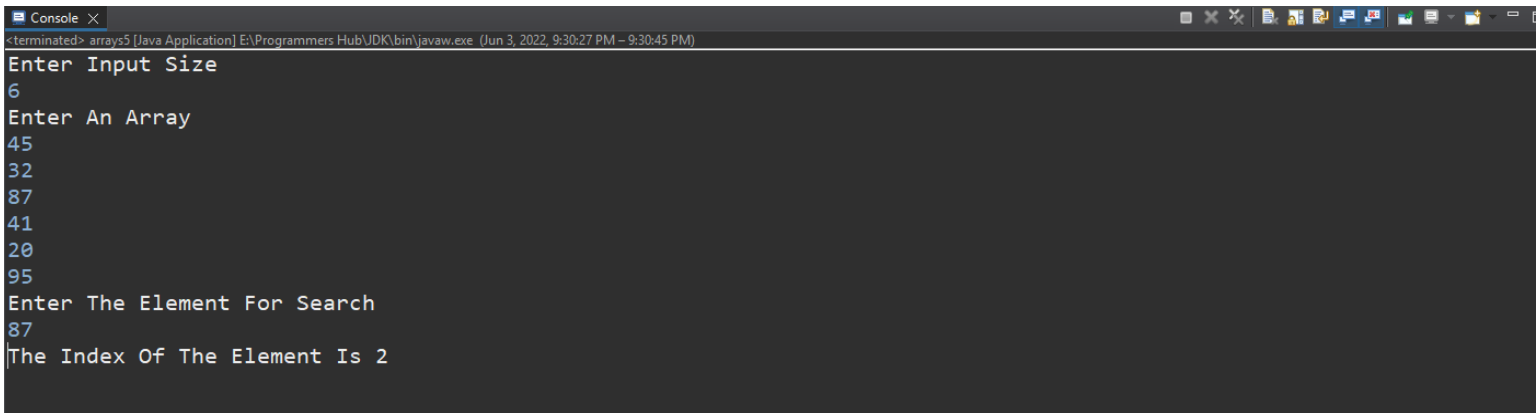
Linear Search

Linear Search is the process when we traverse the array by looping across every element present in it and then return the index of the desired element we were looking for. If we donot find it, we return a negative -1 value

```
package Arrays1;
import java.util.Scanner;

public class arrays5 {

    public static int linearSearch(int[] array,int searchElement)
    {
        int i;
        for(i=0;i<array.length;i++)
        {
            if(searchElement==array[i])
                return i;

        }
        return -1;
    }

    public static void main(String[] args) {
        Scanner sc = new Scanner(System.in);
        System.out.println("Enter Input Size");
        int inputSize=sc.nextInt();
        System.out.println("Enter An Array");
        int[] array = new int[inputSize];
        for(int i=0;i<array.length;i++)
        {
            array[i]= sc.nextInt();

        }
        System.out.println("Enter The Element For Search");
        int searchElement = sc.nextInt();
        System.out.println("The Index Of The Element Is "+linearSearch(array,
searchElement));
```

```
    }

}
```

Output



Binary Search

Binary Search is a searching algorithm, used in a sorted array by repeatedly dividing the search interval in half. The idea of binary search is to use the information that the array is sorted and reduce the time complexity to O(Log n).

Binary Search Algorithm:

The basic steps to perform Binary Search are:

- Begin with the mid element of the whole array as search key.
- If the value of the search key is equal to the item then return index of the search key.
- Or if the value of the search key is less than the item in the middle of the interval, narrow the interval to the lower half.
- Otherwise, narrow it to the upper half.
- Repeatedly check from the second point until the value is found or the interval is empty.

```
package Arrays1;

import java.util.Scanner;

public class arrays6 {

    public static int binarySearch(int[] array,int searchElement)
    {
        int start=0, end=array.length-1;
```

```
        while(start<=end)
        {
            int mid=(start+end)/2;
            if(searchElement==array[mid])
                return mid;
            else
            {
                if(searchElement<array[mid])
                {
                    end=mid-1;

                }
                if(searchElement>array[mid])
                {
                    start=mid+1;
                }
            }

        }

        return -1;
    }

    public static void main(String[] args) {
        Scanner sc = new Scanner(System.in);
        System.out.println("Enter Input Size");
        int inputSize=sc.nextInt();
        System.out.println("Enter An Array In Sorted Order");
        int[] array = new int[inputSize];
        for(int i=0;i<array.length;i++)
        {
            array[i]= sc.nextInt();

        }
        System.out.println("Enter The Element For Search");
        int searchElement = sc.nextInt();
        System.out.println("The Index Of The Element Is "+binarySearch(array,
searchElement));



    }

}
```

Output

```
Console
<terminated> arrays6 [Java Application] E:\Programmers Hub\JDK\bin\javaw.exe (Jun 3, 2022, 11:29:45 PM – 11:30:01 PM)
6
Enter An Array In Sorted Order
14
16
18
20
22
24
Enter The Element For Search
14
The Index Of The Element Is 0
```

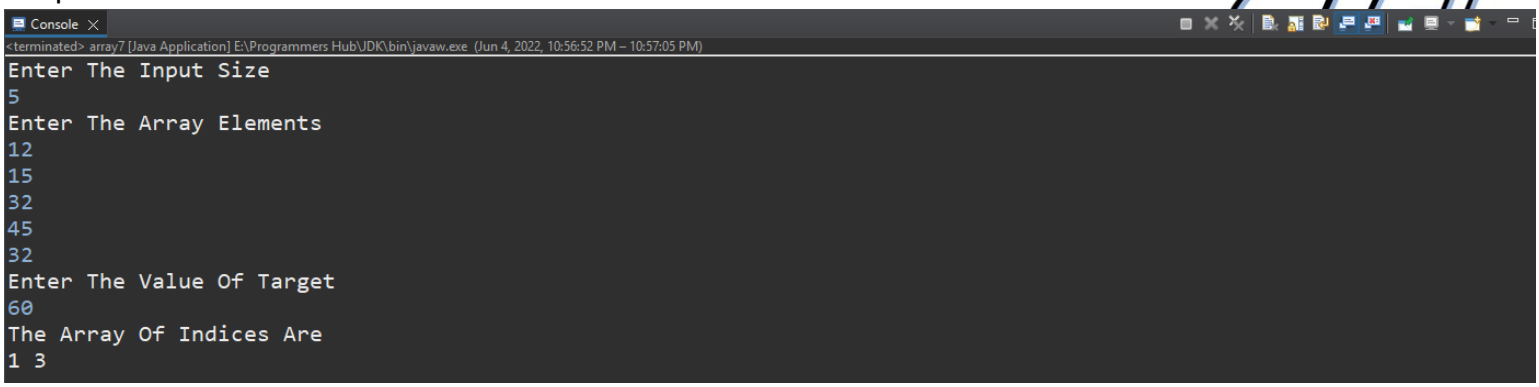
Problem- **Leetcode**

[Two Sums Problem]

Given an array of numbers nums, and a value target, find the elements present in the arrays which adds up to the value of target and return an array consisting of indices. If no such elements exist, return -1 as indices.

```
package Arrays1;
import java.util.Scanner;
public class array7
{
public static int[] twoSum(int[] nums, int target) {

        int i,j;
        int indices[] = new int[2];
        boolean isFound=false;
        for(i=0;i<nums.length-1;i++)
        {

           for(j=i+1;j<nums.length;j++)
           {
            if(nums[i]+nums[j]==target)
            {
                indices[0]=i;
                indices[1]=j;
               isFound=true;
                break;
            }

           }

            if(isFound==true)
                break;

        }

        if(isFound==false)
        {
            indices[0]=-1;
            indices[1]=-1;
        }

        return indices;
        }

public static void main(String[] args) {

    Scanner sc= new Scanner(System.in);

    System.out.println("Enter The Input Size");
    int inputSize=sc.nextInt();

    int[] nums = new int[inputSize];
    System.out.println("Enter The Array Elements");

    for(int i=0;i<nums.length;i++)
        nums[i]=sc.nextInt();
    System.out.println("Enter The Value Of Target");
    int target=sc.nextInt();

    int[] arr=twoSum(nums, target);

    System.out.println("The Array Of Indices Are ");
    for(int i=0;i<arr.length;i++)
        System.out.print(arr[i]+" ");
   }
}
```

Output



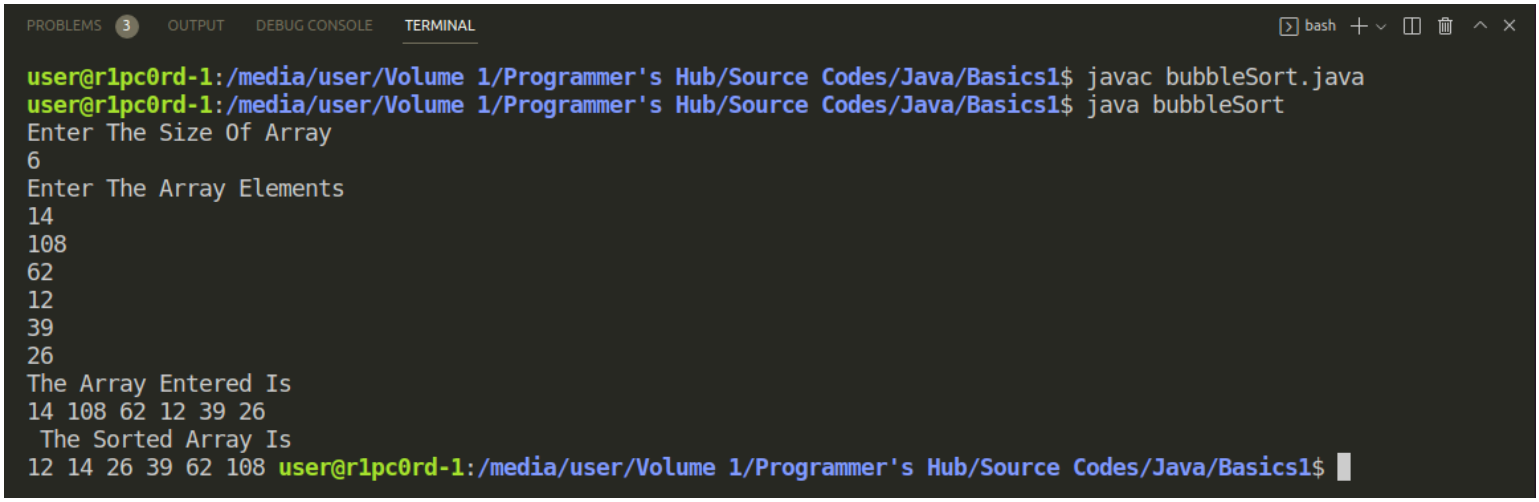
Sorting In Arrays

Bubble Sort

```
import java.util.Scanner;
public class bubbleSort
{
   public static int[] bubbleSortArray(int[] array)
   {
     int counter=1,i;
     while(counter!=array.length-1)
     {
       for(i=0;i<array.length-counter;i++)
       {
           int temp=0;
           if(array[i]>array[i+1])
           {
               temp=array[i];
               array[i]=array[i+1];
               array[i+1]=temp;
           }

       }
       counter++;
     }
     return array;
   }

   public static void main(String[] args)
   {
       Scanner sc = new Scanner(System.in);
       System.out.println("Enter The Size Of Array");
       int arraySize = sc.nextInt();

       int[] array = new int[arraySize];
       int i;
       System.out.println("Enter The Array Elements");

       for(i=0;i<array.length;i++)
       {
           array[i]=sc.nextInt();
       }
       System.out.println("The Array Entered Is ");

       for(i=0;i<array.length;i++)
       {
           System.out.print(array[i]+" ");
```

```
        }
        array = bubbleSortArray(array);
        System.out.println("\n The Sorted Array Is");
        for(i=0;i<array.length;i++)
        {
            System.out.print(array[i]+" ");
        }

    }
}
```

Output



Insertion Sort

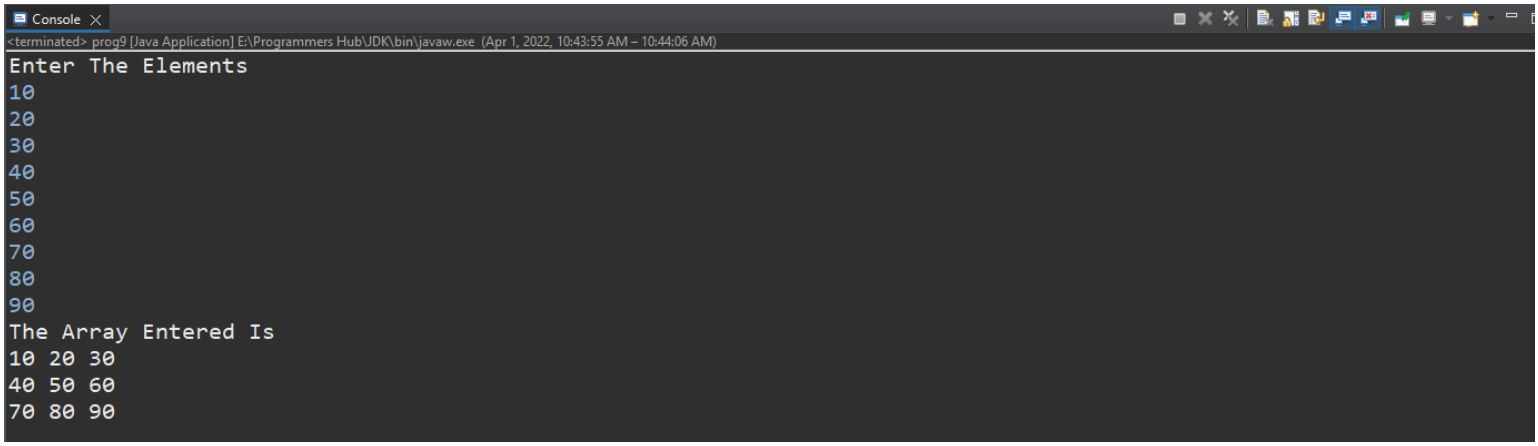
Java 2D Arrays

Creating A 2D Array

```
package Basics1;
import java.util.Scanner;
public class prog9 {


    public static void main(String[] args)
    {
        Scanner sc = new Scanner(System.in);

        int new2DArray [][] = new int [3][3];
        int i,j;
        System.out.println("Enter The Elements");

        for(i=0;i<3;i++)
        {
            for(j=0;j<3;j++)
            {
                new2DArray[i][j]=sc.nextInt();
            }
        }
        System.out.println("The Array Entered Is ");
        for(i=0;i<3;i++)
        {
            for(j=0;j<3;j++)
            {
                System.out.print(new2DArray[i][j]+" ");
            }
            System.out.println();
```

```
        }
    }

}
```

Output



Accessing Elements In 2D Arrays

```
package Basics1;
import java.util.Scanner;
public class prog9 {


    public static void main(String[] args)
    {
        Scanner sc = new Scanner(System.in);

        int new2DArray [][] = new int [3][3];
        int i,j;
        System.out.println("Enter The Elements");

        for(i=0;i<3;i++)
        {
            for(j=0;j<3;j++)
            {
                new2DArray[i][j]=sc.nextInt();
            }
        }

        System.out.println("The Array Entered Is ");

        for(i=0;i<3;i++)
        {
            for(j=0;j<3;j++)
            {
                System.out.print(new2DArray[i][j]+" ");
            }
            System.out.println();
        }

        System.out.println("Enter Row Of Element To Be Replaced");
        int rows = sc.nextInt();

        System.out.println("Enter Col  Of Element To Be Replaced");
        int cols = sc.nextInt();
```

```
        System.out.println("Enter The Replacement");
        new2DArray[rows-1][cols-1] = sc.nextInt();

        System.out.println("The Modified Array  Is ");

        for(i=0;i<3;i++)
        {
            for(j=0;j<3;j++)
            {
                System.out.print(new2DArray[i][j]+" ");
            }
            System.out.println();
        }

    }
}
```

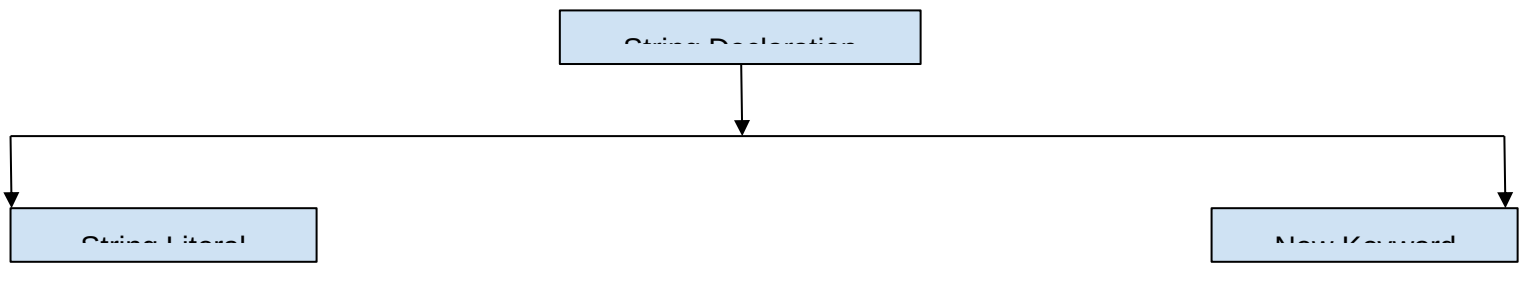
Output

```
Console
<terminated> prog9 [Java Application] E:\Programmers Hub\JDK\bin\javaw.exe  (Apr 1, 2022, 12:06:10 PM – 12:06:31 PM)
Enter The Elements
24
26
51
58
95
45
65
32
10
The Array Entered Is
24 26 51
58 95 45
65 32 10
Enter Row Of Element To Be Replaced
2
Enter Col  Of Element To Be Replaced
1
Enter The Replacement
89
The Modified Array  Is
24 26 51
89 95 45
65 32 10
```

Detailed Discussion On Arrays And Strings Will Be Done In The Upcoming Notes

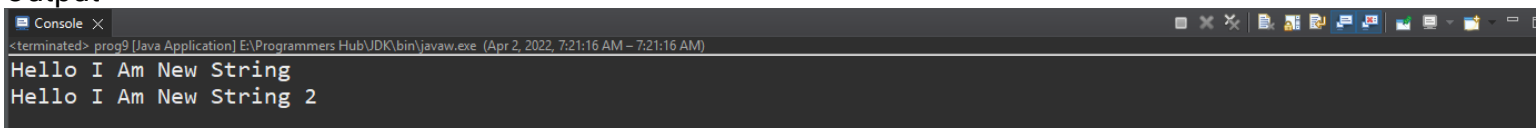
Java Strings

Strings in java are a sequence of characters . Strings in Java can be declared in two ways



```java
package Basics1;
import java.util.Scanner;
public class prog9 {


    public static void main(String[] args)
    {
        String newString = "Hello I Am New String";
        String newString2 = new String("Hello I Am New String 2 ");

        System.out.println(newString);
        System.out.println(newString2);
    }

}
```

Output



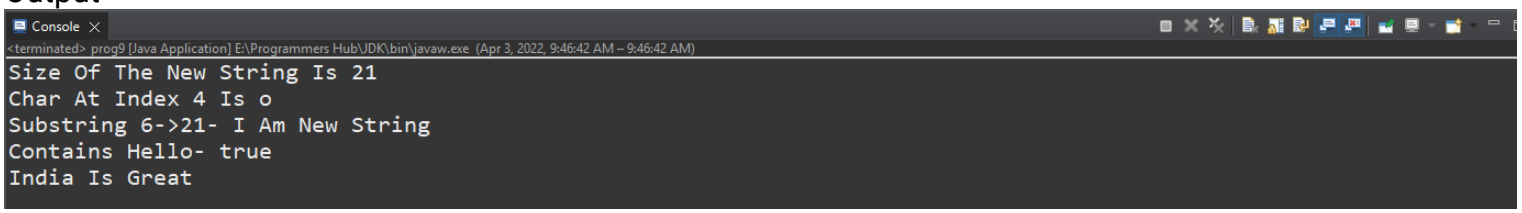
String Functions

Sl No	Function Name	Returns	Description
1	length( )	int	Finds out the length of a string
2	charAt(int index)	char	Returns the character present at the given index
3	subString(int startIndex, int endIndex)	String	Returns the substring starting from the given index and ending at the given index
4	contains(String Substring)	boolean	Returns true or false depending upon the presence of the substring inside the given string
5	join(String str1,String str2)	String	Returns a String by joining all the given parameters

```java
package Basics1;

import java.util.Scanner;

public class prog9 {


    public static void main(String[] args)
    {
        String newString = "Hello I Am New String";

        System.out.println("Size Of The New String Is "+newString.length());
        System.out.println("Char At Index 4 Is "+newString.charAt(4));
        System.out.println("Substring 6->21- " +newString.substring(6, 21));

    }
```

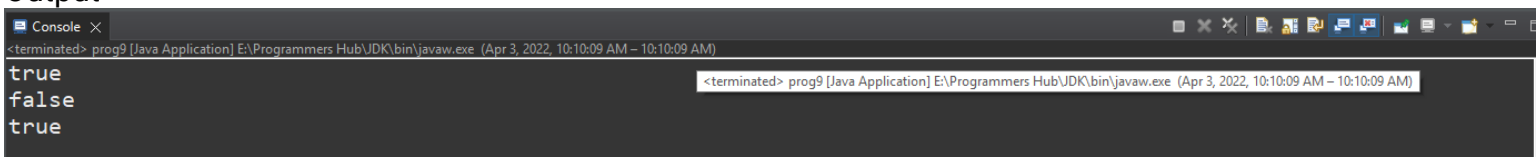
```
}
```

Output




```
package Basics1;
import java.util.Scanner;
public class prog9 {


    public static void main(String[] args)
    {
        String newString = "Hello I Am New String";

        System.out.println("Size Of The New String Is "+newString.length());
        System.out.println("Char At Index 4 Is "+newString.charAt(4));
        System.out.println("Substring 6->21- " +newString.substring(6, 21));
     }

}
```

Output



Sl No	Function Name	Returns	Description
6	equals(String str)	boolean	Checks if the two Strings are equal or not
7	isEmpty(String str)	boolean	Checks if the given string is empty
8	replace(char new, char old)	void	Replaces all old char sequences with the new one
9	concat(String str1,String str2)	String	Returns a concatenated string by concatenating all the Strings entered as parameters
10	indexOf(String subString)	int	Returns the index of the first occurrence of the substring
11	indexOf(String subString, int index)	int	Returns the index of the occurrence of the substring from the given index

12	toLowercase( )	void	Converts the given string to lowercase
13	toUppercase( )	void	Converts the given string to uppercase
14	String.valueOf(int num)	void	Converts the input into a String
15	trim( )	void	It removes whitespace from the String

isEmpty( )

```
package Basics1;
import java.util.Scanner;
public class prog9 {


    public static void main(String[] args)
    {
        String s1="";
        System.out.println(s1.isEmpty());

        String s2="Welcome To Coding";
        System.out.println(s2.isEmpty());
     }

}
```

```
Console
<terminated> prog9 [Java Application] E:\Programmers Hub\JDK\bin\javaw.exe (Apr 5, 2022, 9:16:07 PM – 9:16:07 PM)
true
false
```

replace(char new , char old)

```
package Basics1;
import java.util.Scanner;
public class prog9 {


    public static void main(String[] args)
    {
        String s1="kantry";
        System.out.println(s1.replace("ka","cou"));
    }

}
```

```
Console
<terminated> prog9 [Java Application] E:\Programmers Hub\JDK\bin\javaw.exe (Apr 5, 2022, 9:24:22 PM – 9:24:22 PM)
country
```

concat(String str1,String str2)

```
package Basics1;
import java.util.Scanner;
public class prog9 {
```

```
    public static void main(String[] args)
    {
        String s1="India";
        String s2="Is In South Asia";
        System.out.println(s1+" ".concat(s2));
    }

}
```

```
Console
<terminated> prog9 [Java Application] E:\Programmers Hub\JDK\bin\javaw.exe (Apr 5, 2022, 9:30:37 PM – 9:30:38 PM)
India Is In South Asia
```

indexOf(String str, int index)

```
package Basics1;
import java.util.Scanner;
public class prog9 {


    public static void main(String[] args)
    {
        String s1="Malyalam";
        System.out.println(s1.indexOf("al"));
        System.out.println(s1.indexOf("al",2));
    }

}
```

Output

```
Console
<terminated> prog9 [Java Application] E:\Programmers Hub\JDK\bin\javaw.exe (Apr 5, 2022, 9:47:07 PM – 9:47:07 PM)
1
4
```

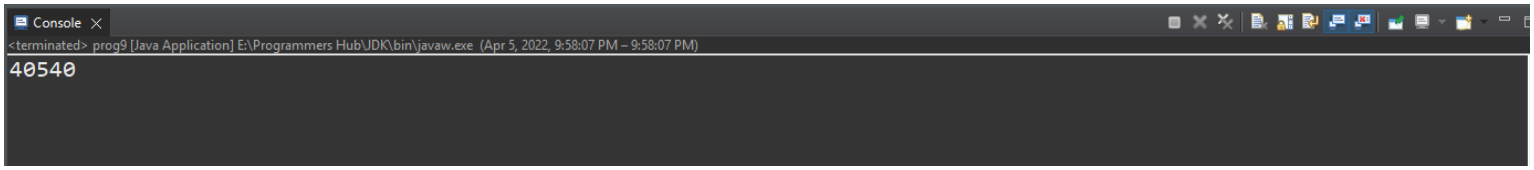
toLowerCase( ), toUpperCase( )

```
package Basics1;
import java.util.Scanner;
public class prog9 {


    public static void main(String[] args)
    {
        String s1="Java is an Object Oriented Programming Language";
        System.out.println(s1.toLowerCase());
        System.out.println(s1.toUpperCase());
    }
}
```

```
Console
<terminated> prog9 [Java Application] E:\Programmers Hub\JDK\bin\javaw.exe (Apr 5, 2022, 9:50:46 PM – 9:50:47 PM)
java is an object oriented programming language
JAVA IS AN OBJECT ORIENTED PROGRAMMING LANGUAGE
```

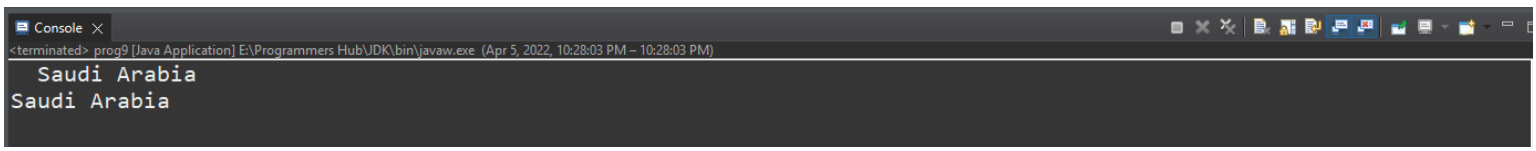
String.valueOf( )

```
package Basics1;

import java.util.Scanner;

public class prog9 {


    public static void main(String[] args)
    {
        int val1=405;
        int val2=40;

        System.out.println(val1+String.valueOf(val2));
    }

}
```



trim ( )

```
package Basics1;
import java.util.Scanner;
public class prog9 {


    public static void main(String[] args)
    {
        String name="  Saudi Arabia  ";
        System.out.println(name);
        System.out.println(name.trim());
    }

}
```



Detailed Discussion On Arrays And Strings Will Be Done In The Upcoming Notes

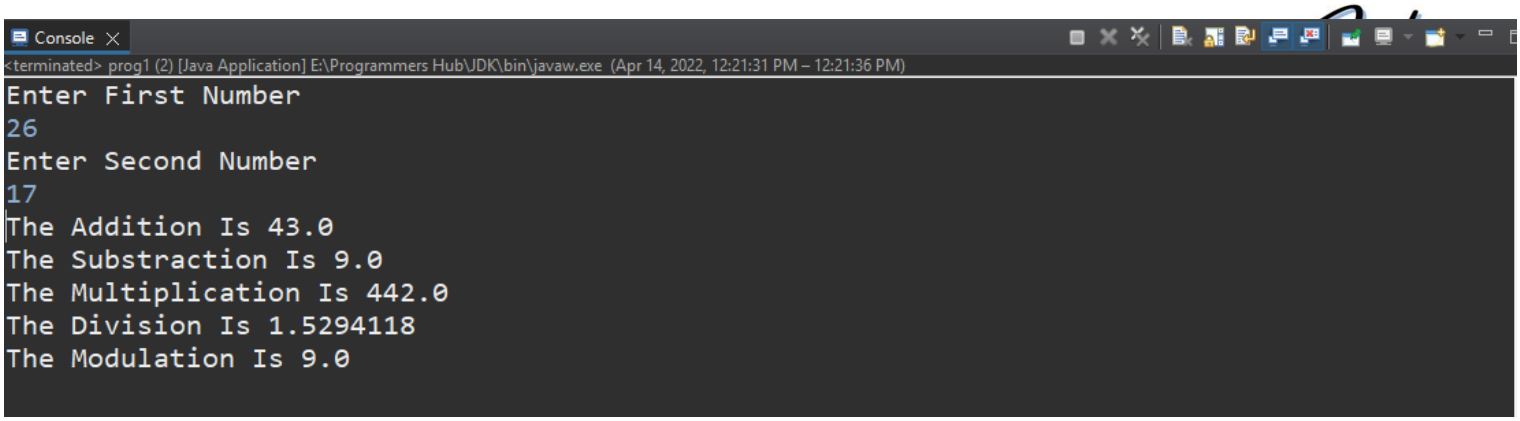
Java Programs

In this section, we are going to solve an entire question paper comprising of 30+ Core Java problems of various difficulties .

Problem 1- WAP to demonstrate Addition, Subtraction, Multiplication, Division And Modulation

```
package Questions;

import java.util.Scanner;

public class prog1 {

    public static void main(String[] args) {

        Scanner sc = new Scanner(System.in);

        System.out.println("Enter First Number");
        float num1=sc.nextFloat();
        System.out.println("Enter Second Number");
        float num2 = sc.nextFloat();

        System.out.println("The Addition Is "+(num1+num2));
        System.out.println("The Substraction Is "+(num1-num2));
        System.out.println("The Multiplication Is "+(num1*num2));
        System.out.println("The Division Is "+(num1/num2));
        System.out.println("The Modulation Is "+(num1%num2));

    }

}
```

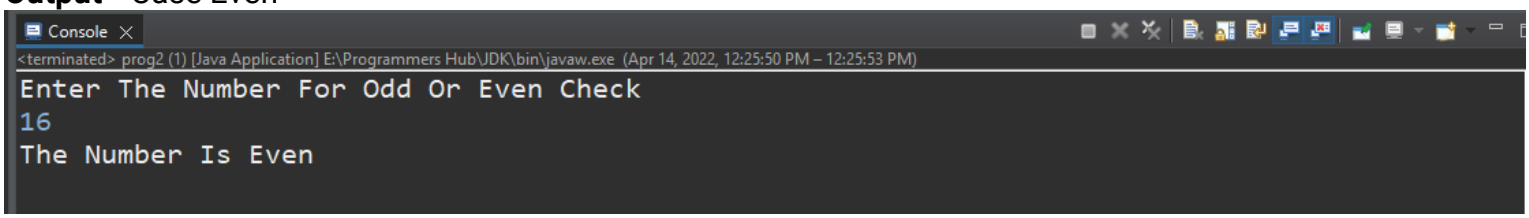
Output



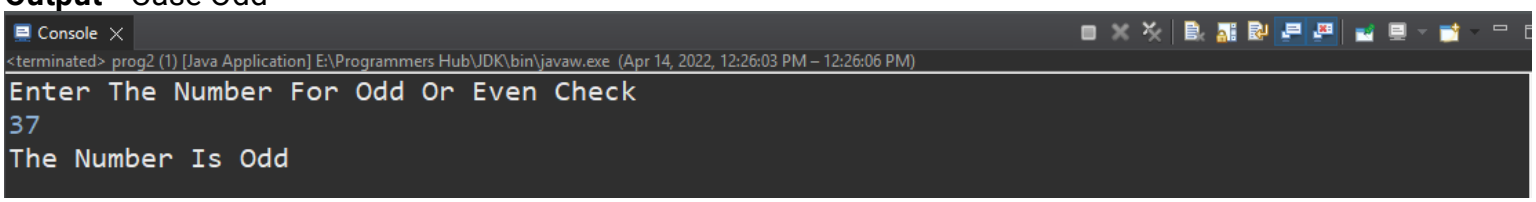
Problem 2- WAP to check if a given number is odd or even

```
package Questions;

import java.util.Scanner;

public class prog2 {

    public static void main(String[] args) {

        Scanner sc = new Scanner(System.in);
        System.out.println("Enter The Number For Odd Or Even Check");
        int num= sc.nextInt();

        if(num%2==0)
            System.out.println("The Number Is Even ");
        else
            System.out.println("The Number Is Odd");
    }

}
```

Output - Case Even



Output - Case Odd



Problem 3- WAP to display the size of all primitive datatypes in Java

```
package Questions;
import java.util.Scanner;
public class prog3 {
```

```
    public static void main(String[] args)
    {
        System.out.println("The Size Of Integer Is "+Integer.SIZE/8+" bytes");
        System.out.println("The Size Of Float Is "+Float.SIZE/8+" bytes");
        System.out.println("The Size Of Char Is "+Character.SIZE/8+" bytes");
        System.out.println("The Size Of Long Is "+Long.SIZE/8);
        System.out.println("The Size Of Double Is "+Double.SIZE/8);
        System.out.println("The Size Of Short Is "+Short.SIZE/8);
        System.out.println("THe Size Of Byte Is "+Byte.SIZE/8);
        System.out.println("The Size Of Boolean Is 1 Bit");

    }
}
```

Output

```
Console
<terminated> prog3 (1) [Java Application] E:\Programmers Hub\JDK\bin\javaw.exe  (Apr 14, 2022, 6:20:02 PM – 6:20:03 PM)
The Size Of Integer Is 4 bytes
The Size Of Float Is 4 bytes
The Size Of Char Is 2 bytes
The Size Of Long Is 8
The Size Of Double Is 8
The Size Of Short Is 2
THe Size Of Byte Is 1
The Size Of Boolean Is 1 Bit
```

Problem 4 - WAP to convert the following algebraic expressions into Java Expressions

a. $b^2 - 4ac$

```
package Questions;
import java.util.Scanner;
public class prog4a {

    public static void main(String[] args)
    {
        Scanner sc = new Scanner(System.in);
        System.out.println("Enter the values of a");
        int a= sc.nextInt();
        System.out.println("Enter the value of b");
        int b = sc.nextInt();
        System.out.println("Enter the value of c");
        int c = sc.nextInt();
        float res = (b*b)-(4*a*c);
        System.out.println("The Result Is "+res);

     }

}
```

Output

```
Console
<terminated> prog4a [Java Application] E:\Programmers Hub\JDK\bin\javaw.exe  (Apr 14, 2022, 7:04:56 PM – 7:05:05 PM)
Enter the values of a
2
Enter the value of b
4
Enter the value of c
3
The Result Is -8.0
```

b. $a^2 + 2ab + b^2$

```java
package Questions;
import java.util.Scanner;
public class prog4a {


     public static void main(String[] args)
    {
        Scanner sc = new Scanner(System.in);
        System.out.println("Enter the values of a");
        int a= sc.nextInt();
        System.out.println("Enter the value of b");
        int b = sc.nextInt();
        float res = (a*a)+2*a*b+(b*b);
        System.out.println("The Result Is "+res);
     }
}
```

c. (a+b) / (c+d)

```java
package Questions;
import java.util.Scanner;
public class prog4a {

    public static void main(String[] args)
    {
        Scanner sc = new Scanner(System.in);
        System.out.println("Enter the values of a");
        float a= sc.nextFloat();
        System.out.println("Enter the value of b");
        float b = sc.nextFloat();
        System.out.println("Enter the value of c");
        float c = sc.nextFloat();
        System.out.println("Enter the value of d");
        float d = sc.nextFloat();
        float res = (a+b)/(c+d);
        System.out.println("The Result Is "+res);
     }

}
```

d. [(ab - c) / (a^2 + b)]*e

```java
package Questions;
import java.util.Scanner;
public class prog4a {

    public static void main(String[] args)
    {
        Scanner sc = new Scanner(System.in);
        System.out.println("Enter the values of a");
```

```
        float a= sc.nextFloat();
        System.out.println("Enter the value of b");
        float b = sc.nextFloat();
        System.out.println("Enter the value of c");
        float c= sc.nextFloat();
        System.out.println("Enter the value of d");
        float d = sc.nextFloat();
        System.out.println("Enter the value of e ");
        float e=sc.nextFloat();
        float res = (((a*b)-c) /((a*a)+b))*e;
        System.out.println("The Result Is "+res);
    }

}
```

Output

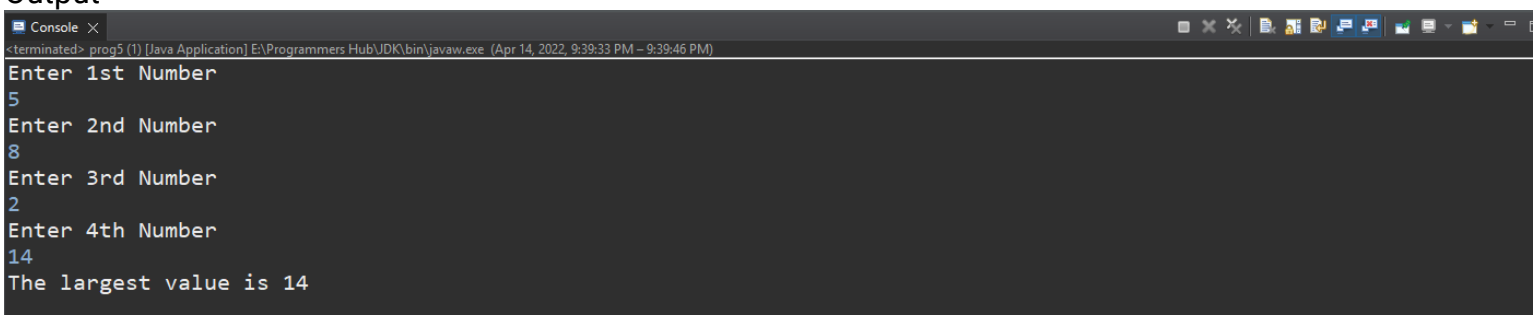
```
Console
<terminated> prog4a [Java Application] E:\Programmers Hub\JDK\bin\javaw.exe (Apr 15, 2022, 10:36:39 AM – 10:36:50 AM)
Enter the values of a
2
Enter the value of b
4
Enter the value of c
6
Enter the value of d
7
Enter the value of e
3
The Result Is 0.75
```

Problem 5 - WAP to input 4 numbers and find largest number among them (without using array)

```
package Questions;
import java.util.Scanner;
public class prog5 {

    public static void main(String[] args) {

        Scanner sc= new Scanner(System.in);
        System.out.println("Enter 1st Number");
        int a=sc.nextInt();
        System.out.println("Enter 2nd Number");
        int b=sc.nextInt();
        System.out.println("Enter 3rd Number");
        int c= sc.nextInt();
        System.out.println("Enter 4th Number");
        int d=sc.nextInt();

        int comparator = Math.max(a, b);
        comparator=Math.max(comparator, c);
```

```
        comparator=Math.max(comparator,d);

        System.out.println("The largest value is "+comparator);
    }

}
```

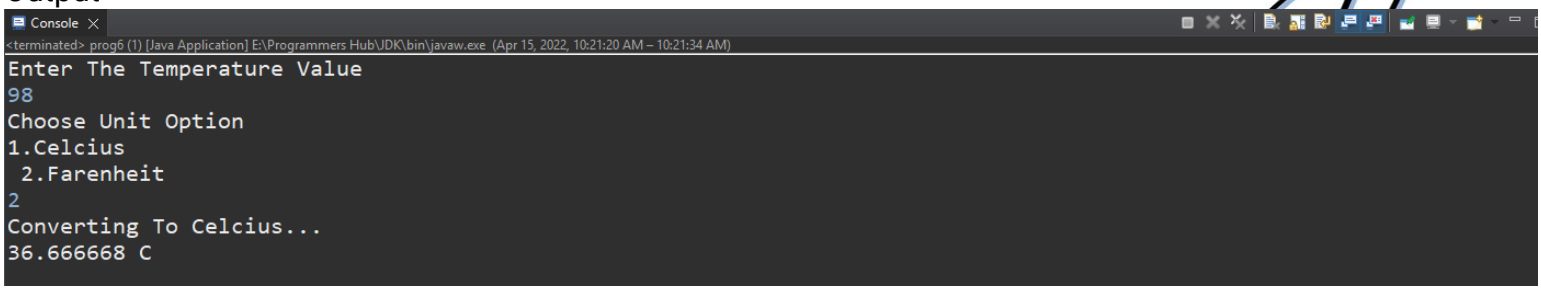
Output



Problem 6. WAP to input temperature in Celcius or Farenheit and convert it into vice versa

```
package Questions;
import java.util.Scanner;
public class prog6 {

    public static String convertTemp(float tempVal,int option)
    {
        if(option==1)
        {
            System.out.println("Converting To Farenhite...");

            return (9*tempVal+160)/5+" F";
        }
        else
        {
          System.out.println("Converting To Celcius...");

            return (5*tempVal-160)/9+" C";
        }
    }
    public static void main(String[] args) {

        Scanner sc = new Scanner(System.in);
        System.out.println("Enter The Temperature Value");
        float tempVal=sc.nextFloat();
        System.out.println("Choose Unit Option");
        System.out.println("1.Celcius\n 2.Farenheit");
        int opt=sc.nextInt();
        System.out.println(convertTemp(tempVal,opt));
    }

}
```

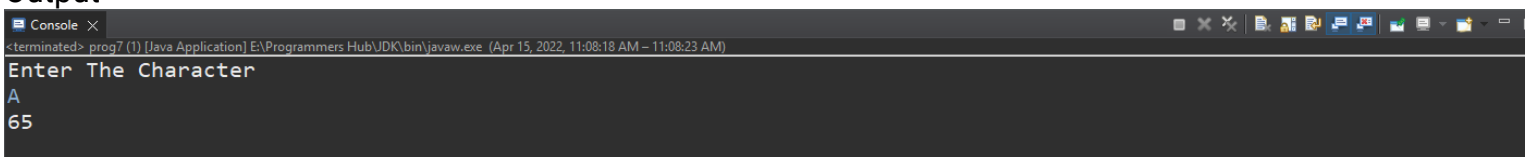
Output



Problem 7 - WAP to input a character and print its ASCII value

```
package Questions;
import java.util.Scanner;
public class prog7 {
   public static void main(String[] args) {
    Scanner sc = new Scanner(System.in);
    System.out.println("Enter The Character ");
    System.out.println((int)sc.next().charAt(0));
   }
}
```

Output

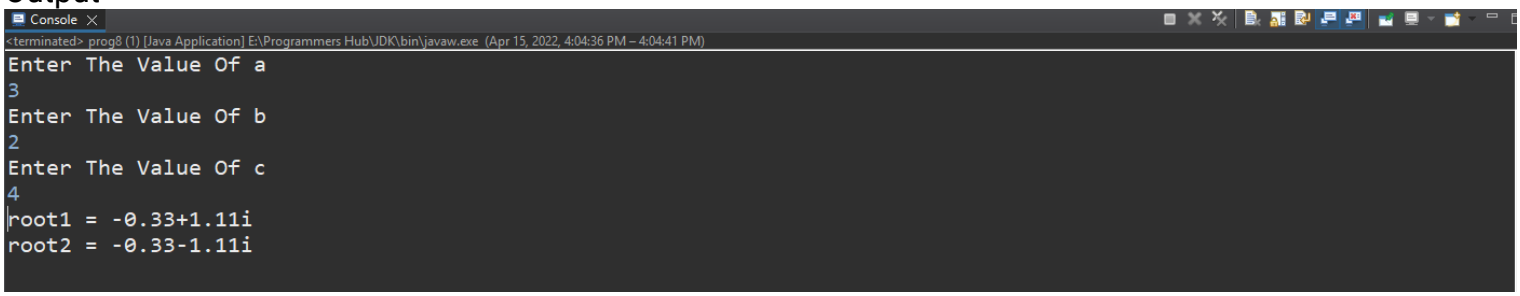


Problem 8 - WAP to input the coefficients of quadratic equation and find out its roots

```
package Questions;
import java.util.Scanner;
public class prog8 {

    public static void main(String[] args) {

        Scanner sc = new Scanner(System.in);
        System.out.println("Enter The Value Of a");
        float a = sc.nextFloat();
        System.out.println("Enter The Value Of b");
        float b = sc.nextFloat();
        System.out.println("Enter The Value Of c");
        float c = sc.nextFloat();

        //Calculation Of Determinant

        float det = (b*b)-4*a*c;
        if(det>0)
        {
            float root1,root2;
            root1= (float)((- b+ Math.sqrt(b*b-4*a*c))/2*a);
            root2=(float)((- b+ Math.sqrt(b*b-4*a*c))/2*a);
            System.out.format("The Roots Are %0.3f and 0.3%f",root1,root2);
        }
        else if(det==0)
        {
            float root =(float)-b/(2*a);
```

```
            System.out.format("The Roots Are %0.3f and %0.3f",root,root);
        }
        else
        {
              float real =(float) -b / (2 * a);
              float imaginary =(float) Math.sqrt(-det) / (2 * a);
              System.out.format("root1 = %.2f+%.2fi", real, imaginary);
              System.out.format("\nroot2 = %.2f-%.2fi", real, imaginary);
        }
    }

}
```

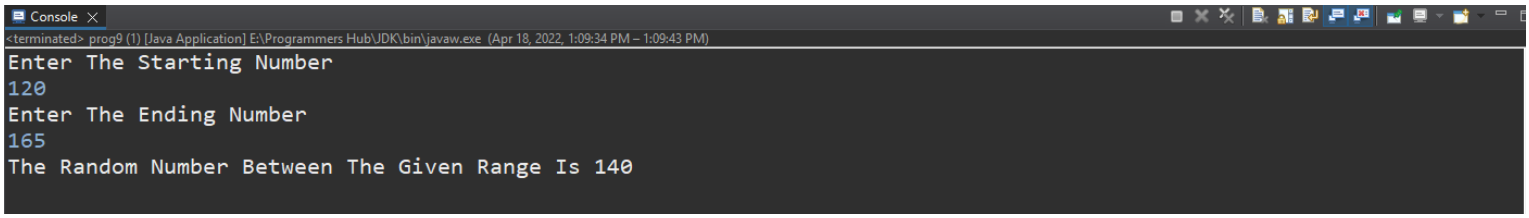
Output



Problem 9 - WAP to input a range and print a random number between the given range

```
package Questions;

import java.util.Random;
import java.util.Scanner;

public class prog9 {

    public static void main(String[] args) {

        Scanner sc = new Scanner(System.in);
        Random rand = new Random();
        System.out.println("Enter The Starting Number ");
        int numStart = sc.nextInt();
        System.out.println("Enter The Ending Number ");
        int numEnd = sc.nextInt();
        System.out.println("The Random Number Between The Given Range Is
"+rand.nextInt(numStart,numEnd));

      }

}
```

Output

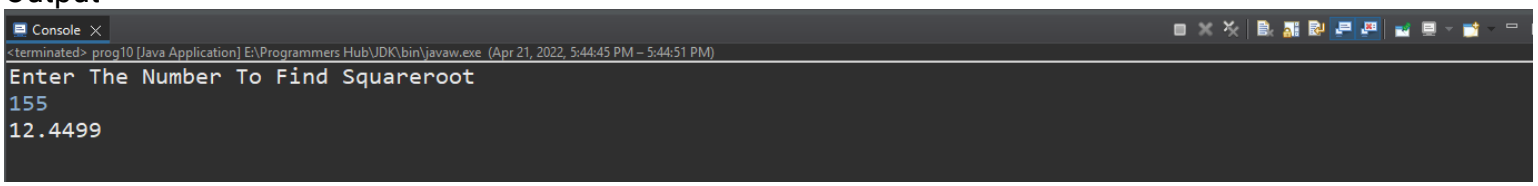


Problem 10- WAP to input a number and find its square root

```
package Questions;
```

```java
import java.util.Scanner;

public class prog10 {

    public static void main(String[] args) {

        Scanner sc = new Scanner(System.in);
        System.out.println("Enter The Number To Find Squareroot");
        int num = sc.nextInt();
        System.out.format("%.4f",Math.sqrt(num));

    }
}
```

Output

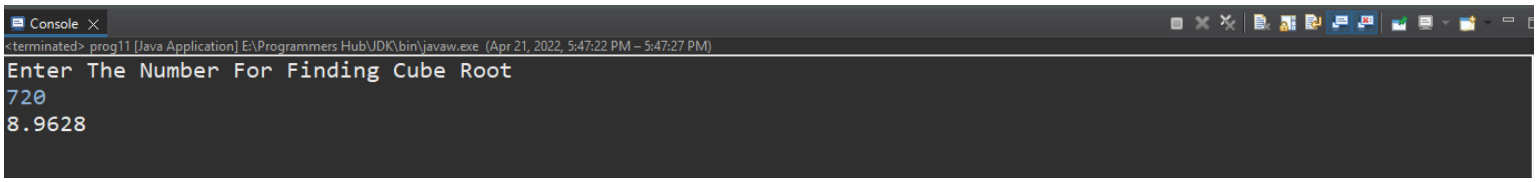


Problem 11- WAP to input a number and find its cube root

```java
package Questions;
import java.util.Scanner;
public class prog10 {

    public static void main(String[] args) {

        Scanner sc = new Scanner(System.in);
        System.out.println("Enter The Number To Find Squareroot");
        int num = sc.nextInt();
        System.out.format("%.4f",Math.sqrt(num));

    }
}
```

Output

Output

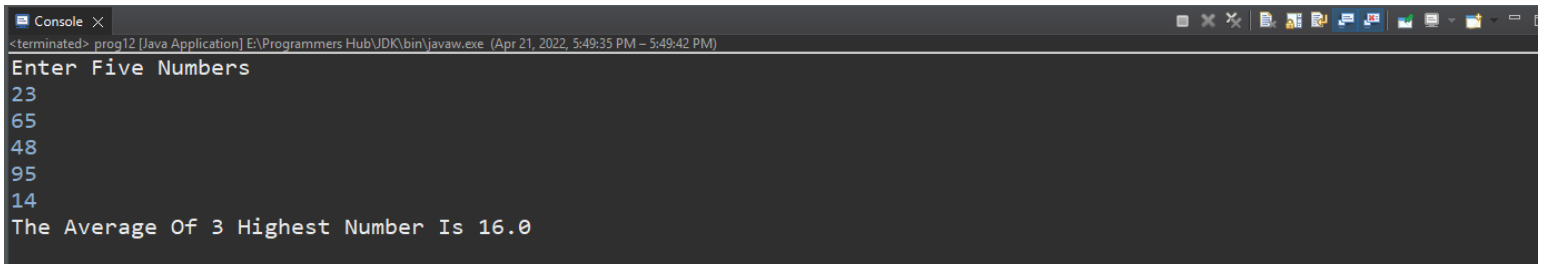


Problem 12- WAP to input 5 numbers and find out the average of the 3 highest numbers among them

```java
package Questions;

import java.util.Arrays;
import java.util.Scanner;

public class prog12 {

    public static void main(String[] args) {
        Scanner sc = new Scanner(System.in);
        System.out.println("Enter Five Numbers ");
        int i; int [] numArray = new int[5];
```

```
        for(i=0;i<5;i++)
            numArray[i]=sc.nextInt();
        Arrays.sort(numArray);
        float sum=0;

        for(i=0;i<3;i++);
        sum+=numArray[numArray.length-i];

        System.out.println("The Average Of 3 Highest Number Is "+(sum/3));

    }

}
```

Output



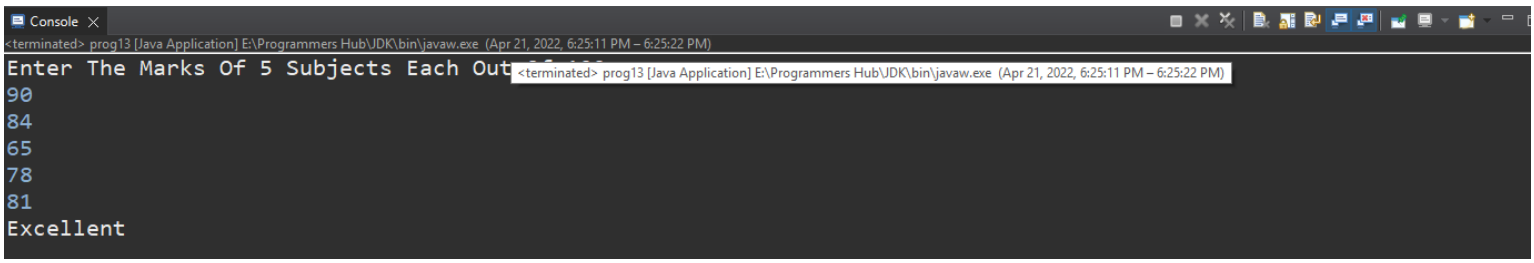
Problem 13- WAP to input marks of 5 subjects and calculate the grade on total. The total marks for each subject is 100.

Sl No	Percentage	Result
1	>=90%	Outstanding
2	>=75% and <90%	Excellent
3	>=60% and <75%	Good
4	>=50% and <60%	Average
5	<50%	Disqualified

```
package Questions;
import java.util.Scanner;
public class prog13 {


    public static String getswitchinput(float result)
    {

        if(result>100 || result<0)
            return "Invalid Inputs.Enter Numbers Again";
        else
```

```
        {
            if(result>=90)
                return "Outstanding";
            if(result>=75 && result <90)
                return "Excellent";
            if(result>=60 && result <75)
                return "Good";
            if(result>=50 && result <60)
                return "Average";
            if(result<50)
                return "Disqualified";

        }
        return null;
    }


    public static void main(String[] args) {

        Scanner sc = new Scanner(System.in);

        System.out.println("Enter The Marks Of 5 Subjects Each Out Of 100 ");
        float [] array = new float[5];

        int i; float sum=0;

        for(i=0;i<array.length;i++)
        {
            array[i]=sc.nextFloat();
            sum+=array[i];

        }
        System.out.println(getswitchinput(sum/5));

    }
}
```

Output



Problem 15- WAP to calculate the consumed Electricity Units and calculate the bill as per the data provided by West Bengal State Electricity Distribution Corporation Limited.

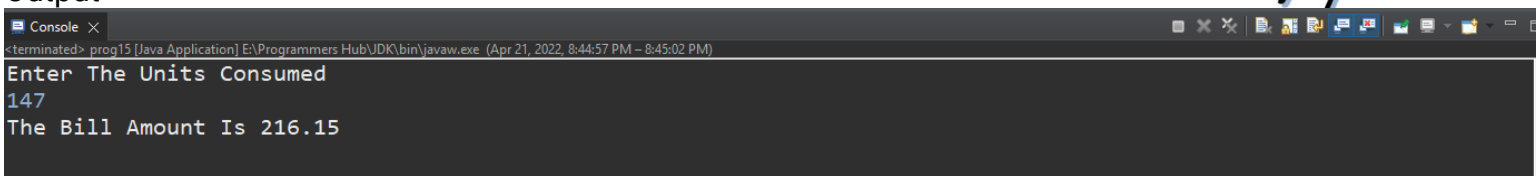
Sl No	Consumption	Base Price	Additional / Unit
1	Below 50		₹ 6.5
2	50 -80		₹ 8.5
3	81 - 120	₹ 100	₹ 0.35

4	121 - 150	₹ 150	₹ 0.45
5	Above 150		₹ 12.5


```java
package Questions;
import java.util.Scanner;
public class prog15 {

    public static int getBills(int units)
    {
        if(units < 50)
            return 1;
        if(units >=50 && units<=80)
            return 2;
        if(units >80 && units<120)
            return 3;
        if(units >120 && units <=150)
            return 4;
        if(units >150)
            return 5;
        return 0;
    }

    public static void main(String[] args)
    {
        Scanner sc = new Scanner(System.in);
        System.out.println("Enter The Units Consumed");
        int units = sc.nextInt();

        switch(getBills(units))
        {
        case 1:
        {
            System.out.println("The Bill Amount Is "+(units*6.5));
            break;
        }
        case 2:
        {
            System.out.println("The Bill Amount Is "+(units*8.5));
            break;
        }
        case 3:
        {
            System.out.println("The Bill Amount Is "+(100+(units*0.35)));
            break;
        }
        case 4:
        {
            System.out.println("The Bill Amount Is "+(150+(units*0.45)));
            break;
        }
        case 5:
        {
            System.out.println("The Bill Amount Is "+(units*12.5));
            break;
        }
        default:
        {
            System.out.println("Failed To Generate Bill");
        }
        }
    }
```

```
}
```

Output

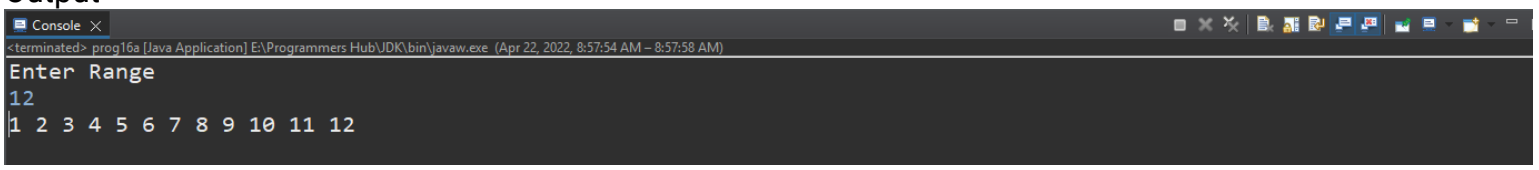


Problem 16 :- Write Programs To Print The Following Series

a. 1,2,3,4,5,6

```
package Questions;
import java.util.Scanner;
public class prog16a {

    public static void main(String[] args) {

        Scanner sc = new Scanner(System.in);
        System.out.println("Enter Range");
        int range =sc.nextInt();
        for(int i=1;i<=range;i++)
            System.out.print(i+" ");
      }
}
```

Output

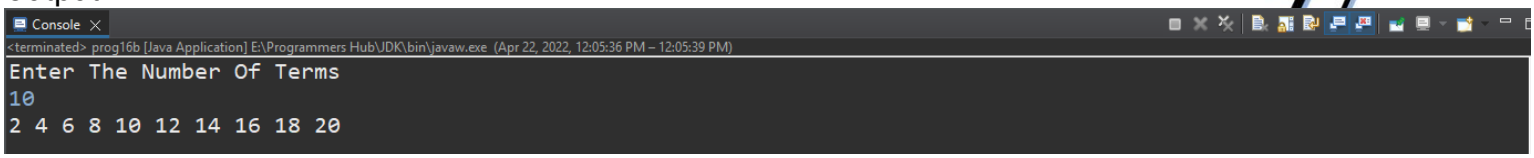


b. 2,4,6,8

```
package Questions;
import java.util.Scanner;
public class prog16b {

    public static void main(String[] args) {

        Scanner sc = new Scanner(System.in);
        System.out.println("Enter The Number Of Terms");
        int range =sc.nextInt();

        for(int i=1;i<=range;i++)
        {
            System.out.print((2*i)+" ");
        }


    }
```

```
}
```

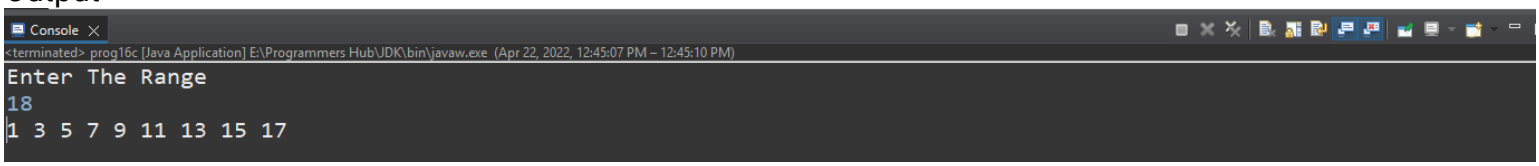
Output



c. 1,3,5,7

```
package Questions;
import java.util.Scanner;
public class prog16c {

    public static void main(String[] args) {
        Scanner sc = new Scanner(System.in);
        System.out.println("Enter The Range");
        int range =sc.nextInt();
        for(int i=1;i<=range;i+=2)
            System.out.print(i+" ");

    }
}
```

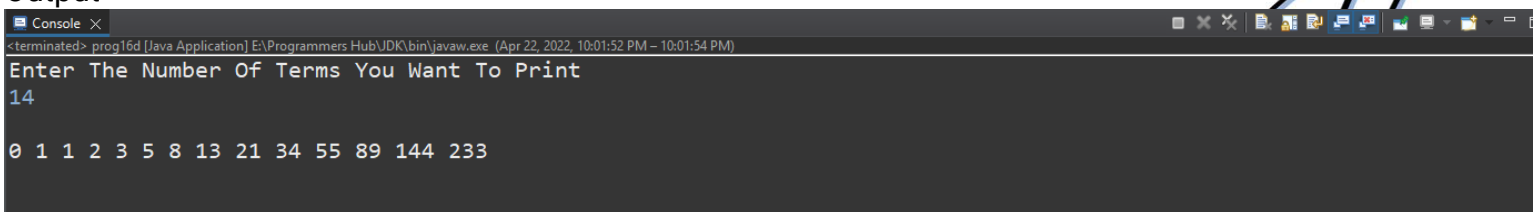
Output



d. Fibonacci Series

```
package Questions;
import java.util.Scanner;
public class prog16d {

    public static void main(String[] args) {

        Scanner sc = new Scanner(System.in);
        System.out.println("Enter The Number Of Terms You Want To Print");
        int terms = sc.nextInt();
        System.out.println();
        int a=0,b=1;
        System.out.print(a+" "+b+" ");

        for(int i=0;i<terms-2;i++)
        {
            int c=a+b;
            System.out.print(c+" ");
            a=b;
            b=c;
        }

    }

}
```

Output



e. 0,1,3,6,10,15,21,28,36,

```
package Questions;
import java.util.Scanner;
public class prog16e {

    public static void main(String[] args) {

        Scanner sc = new Scanner(System.in);
        System.out.println("Enter The Number Of Terms");
        int terms = sc.nextInt();
        int pos=1;
        int a=0;
        System.out.print(0+" ");
        for(int i=0;i<terms-1;i++)
        {
            int sum=a+pos;
            System.out.print(sum+" ");
            a=sum;
            pos++;
        }
     }

}
```

Output

f. 1, 2, 0, 3, -1, 4, -2, 5, -3, 6, -4

```
package Questions;

public class prog16f {

    public static void main(String[] args) {
```

```
        for(int i=1,j=2;j<=7;i--,j++)
        {
            System.out.print(i+" "+j+" ");
        }
    }
}
```

Output

```
Console
<terminated> prog16f [Java Application] E:\Programmers Hub\JDK\bin\javaw.exe (Apr 22, 2022, 11:42:36 PM – 11:42:36 PM)
1 2 0 3 -1 4 -2 5 -3 6 -4 7
```

g. 1, 5, 2, 4, 3, 3, 4, 2, 5, 1

```
package Questions;
public class prog16f {

    public static void main(String[] args) {
        for(int i=1,j=2;j<=7;i--,j++)
        {
            System.out.print(i+" "+j+" ");
        }
    }
}
```

Output

```
Console
<terminated> prog16g [Java Application] E:\Programmers Hub\JDK\bin\javaw.exe (Apr 22, 2022, 11:46:36 PM – 11:46:36 PM)
1 5 2 4 3 3 4 2 5 1
```

h. 1, 5, 2, 6, 3, 7, 4, 8, 5, 9

```
package Questions;
public class prog16g {

    public static void main(String[] args) {
       for(int i=1,j=5;i<=5;i++,j++)
            System.out.print(i+" "+j+" ");
      }
}
```

Output

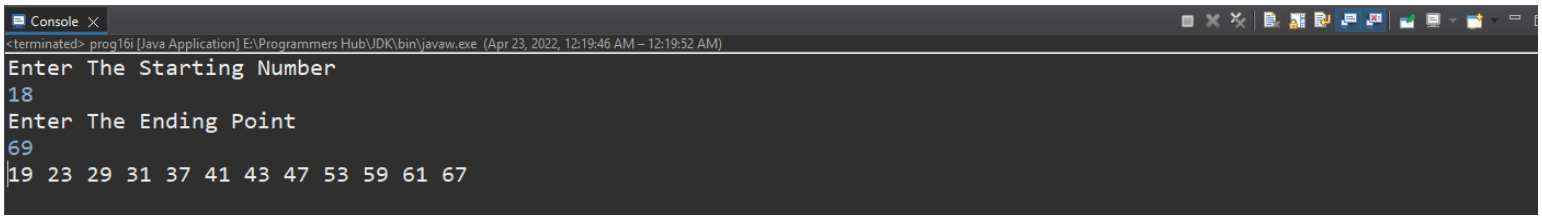
```
Console
<terminated> prog16g [Java Application] E:\Programmers Hub\JDK\bin\javaw.exe (Apr 22, 2022, 11:50:23 PM – 11:50:24 PM)
1 5 2 6 3 7 4 8 5 9
```

i. Print Prime Numbers Between The Given Range

```
package Questions;
import java.util.Scanner;
```

```java
public class prog16i {

    public static void main(String[] args) {
        Scanner sc = new Scanner(System.in);
        System.out.println("Enter The Starting Number");
        int start = sc.nextInt();
        System.out.println("Enter The Ending Point");
        int end = sc.nextInt();

        int i,j;

        for(i=start;i<=end;i++)
        {
            boolean isPrime=true;

            for(j=2;j<i;j++)
            {
                if(i%j==0)
                {
                    isPrime=false;
                    break;
                }
            }

            if(isPrime)
                System.out.print(i+" ");
        }
    }

}
```

Output



Problem 17 - WAP to check if a number is a strong number or not

```java
package Questions;
import java.util.Scanner;
public class prog17 {



    public static int factorialFinder(int dig)
    {
        int prod=1;
        while(dig!=0)
        {
            prod*=dig;
            dig--;
        }
        return prod;

    }
```

```
    public static void main(String[] args) {

        Scanner sc = new Scanner(System.in);
        System.out.println("Enter The Number For Strong Number Check");
        int num = sc.nextInt();
        int copy=num;
        int sum=0;
        while(num!=0)
        {
            int dig = num%10;
            sum+=factorialFinder(dig);
            num/=10;
        }

        if(sum==copy)
            System.out.println("The Number Is A Strong Number");
        else
            System.out.println("The Number Is Not A Strong Number");
     }

}
```

Output

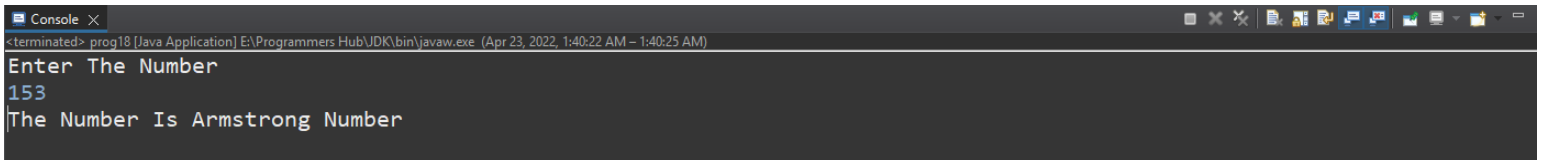
```
Console
<terminated> prog17 [Java Application] E:\Programmers Hub\JDK\bin\javaw.exe (Apr 23, 2022, 1:23:31 AM – 1:23:33 AM)
Enter The Number For Strong Number Check
145
The Number Is A Strong Number
```

Problem 18 - Write a program to check if a number is an armstrong number or not

```
package Questions;
import java.util.Scanner;
public class prog18 {

    public static int cubeRootFinder(int dig)
    {
        return  dig*dig*dig;

    }

    public static void main(String[] args) {

        Scanner sc = new Scanner(System.in);
        System.out.println("Enter The Number");
        int num = sc.nextInt();

        int copy=num;
        int sum=0;

        while(num!=0)
        {
            int dig = num%10;
            sum+=cubeRootFinder(dig);
            num/=10;
        }

        if(sum==copy)
            System.out.println("The Number Is Armstrong Number");
        else
            System.out.println("The Number Is Not Armstrong Number");
```

```
    }

}
```

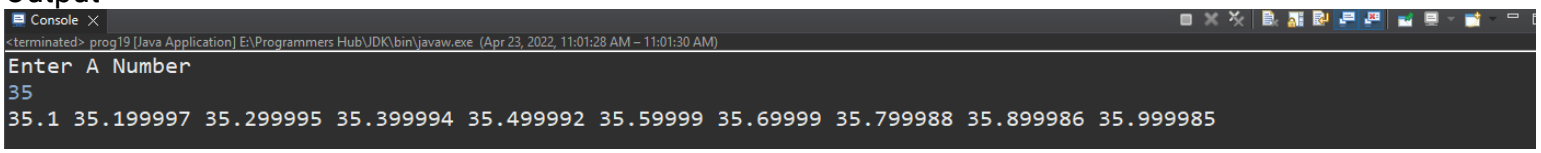
Output



Problem 19 - Write a program to input a number and increase it 10 times by 0.1

```
package Questions;
import java.util.Scanner;
public class prog19 {
   public static void main(String[] args) {

        Scanner sc = new Scanner(System.in);
        System.out.println("Enter A Number");
        float num=sc.nextFloat();

        for(int i=0;i<10;i++)
        {
            num+=0.1;
            System.out.print(num+" ");
        }
     }
}
```

Output



Problem 20 - Write a program to print 1+2+3+4+5+6+7+8+9+10=55

```
package Questions;
public class prog20 {

    public static void main(String[] args) {

        int sum=0;

        for(int i=1;i<=10;i++)
        {
            sum+=i;
            System.out.print(i);

            if(i==10)
            {
```

```
                break;
            }
            System.out.print(" + ");
        }
        System.out.print(" = "+sum);
    }
}
```

Output

```
Console
<terminated> prog20 [Java Application] E:\Programmers Hub\JDK\bin\javaw.exe (Apr 23, 2022, 11:09:31 AM – 11:09:31 AM)
1 + 2 + 3 + 4 + 5 + 6 + 7 + 8 + 9 + 10 = 55
```

Problem 21 - WAP to print the following pattern

```
x
xx
xxx
xxxx
xxxxx
```

```
package Questions;
import java.util.Scanner;
public class prog21 {

    public static void main(String[] args) {

        Scanner sc = new Scanner(System.in);
        System.out.println("Enter The Number Of Rows");
        int rows=sc.nextInt();
        int i,j;
        for(i=1;i<=rows;i++)
        {
            for(j=0;j<i;j++)
            {
                System.out.print("x"+" ");
            }
            System.out.println();
        }
    }

}
```

Output

```
Console
<terminated> prog21 [Java Application] E:\Programmers Hub\JDK\bin\javaw.exe (Apr 23, 2022, 11:55:37 AM – 11:55:39 AM)
Enter The Number Of Rows
5
x
x x
x x x
x x x x
x x x x x
```

Problem 22 - WAP to print the following pattern

```
    x
   xx
  xxx
 xxxx
xxxxx
```

```
package Questions;
import java.util.Scanner;
public class prog21 {

    public static void main(String[] args) {

        Scanner sc = new Scanner(System.in);
        System.out.println("Enter The Number Of Rows");
        int rows=sc.nextInt();
        int i,j,k;
        for(i=1;i<=rows;i++)
        {
            for(j=0;j<rows-i;j++)
            {
                System.out.print("  ");
            }
            for(k=0;k<i;k++)
            {
                System.out.print(" *");
            }
            System.out.println();
        }

    }

}
```

Output

```
Console
<terminated> prog21 [Java Application] E:\Programmers Hub\JDK\bin\javaw.exe (Apr 23, 2022, 12:02:16 PM – 12:02:18 PM)
Enter The Number Of Rows
5
        *
      * *
    * * *
  * * * *
* * * * *
```

Problem 23 - WAP to print the following pattern

```
xxxxx
xxxxx
xxxxx
xxxxx
xxxxx
```

```
package Questions;
import java.util.Scanner;
public class prog21 {

    public static void main(String[] args) {
```

```
        Scanner sc = new Scanner(System.in);
        System.out.println("Enter The Number Of Rows");
        int rows=sc.nextInt();
        int i,j,k;
        for(i=1;i<=rows;i++)
        {
            for(j=0;j<=rows-i;j++)
            {
                System.out.print("  ");
            }
            for(k=0;k<i;k++)
            {
                System.out.print(" *");
            }
            System.out.println();
        }
     }

}
```

Output

```
Console
<terminated> prog21 [Java Application] E:\Programmers Hub\JDK\bin\javaw.exe  (Apr 23, 2022, 12:06:41 PM – 12:06:44 PM)
Enter The Number Of Rows
6
Enter The Number Of Cols
4
x x x x
x x x x
x x x x
x x x x
x x x x
x x x x
```

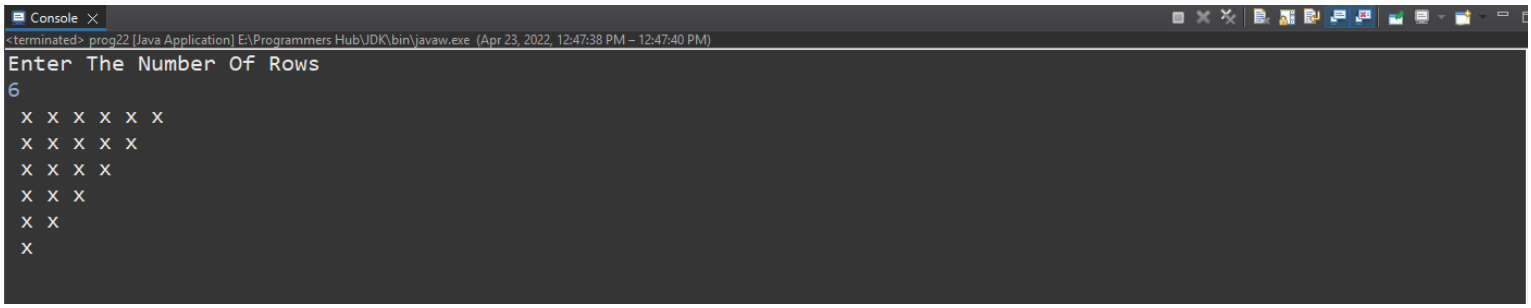
Problem 24 - WAP to print the following pattern

```
xxxxx
xxxx
xxx
xx
x
```

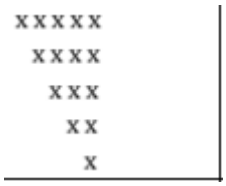
```
package Questions;
import java.util.Scanner;
public class prog22 {

    public static void main(String[] args) {

        Scanner sc = new Scanner(System.in);
        System.out.println("Enter The Number Of Rows");
        int rows = sc.nextInt();
        int i,j;

        for(i=0;i<rows;i++)
        {
            for(j=0;j<rows-i;j++)
            System.out.print(" x");
            System.out.println();
        }
```

```
    }

}
```

Output

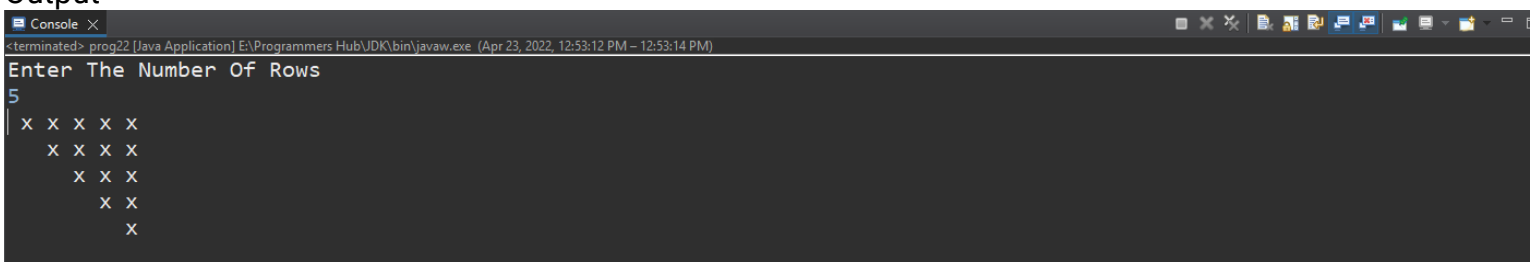


Problem 25 - WAP to print the following pattern




```
package Questions;
import java.util.Scanner;
public class prog22 {

    public static void main(String[] args) {

        Scanner sc = new Scanner(System.in);
        System.out.println("Enter The Number Of Rows");
        int rows = sc.nextInt();
        int i,j,k;

        for(i=0;i<rows;i++)
        {
            for(k=0;k<i;k++)
                System.out.print("  ");
            for(j=0;j<rows-i;j++)
                System.out.print(" x");
            System.out.println();
        }
     }

}
```

Output



Problem 26 - WAP to print the following pattern

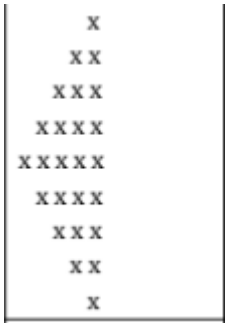
```
x
xx
xxx
xxxx
xxxxx
xxxx
xxx
xx
x
```

```
package Questions;
import java.util.Scanner;
public class prog22 {

    public static void main(String[] args) {

        Scanner sc = new Scanner(System.in);
        System.out.println("Enter The Number Of Rows");
        int rows = sc.nextInt();
        int i,j,k;

        for(i=0;i<rows;i++)
        {
            for(k=0;k<i;k++)
                System.out.print("  ");
            for(j=0;j<rows-i;j++)
                System.out.print(" x");
            System.out.println();
        }
    }

}
```

Output

```
Console
<terminated> prog24 [Java Application] E:\Programmers Hub\JDK\bin\javaw.exe (Apr 23, 2022, 1:01:30 PM – 1:01:31 PM)
Enter The Peak
5
x
x x
x x x
x x x x
x x x x x
x x x x
x x x
x x
x
```

Problem 27 - WAP to print the following pattern




```java
package Questions;
import java.util.Scanner;
public class prog24 {

    public static void main(String[] args) {

        Scanner sc = new Scanner(System.in);
        System.out.println("Enter The Peak");
        int peak = sc.nextInt();

        int i,j,k;
        for(i=0;i<peak;i++)
        {
            for(j=0;j<peak-i;j++)
                System.out.print("  ");
            for(k=0;k<i;k++)
            System.out.print(" X");

            System.out.println();

        }

         for(i=0;i<peak;i++)
        {
            for(j=0;j<i;j++)
                System.out.print("  ");
            for(k=0;k<peak-i;k++)
            System.out.print(" X");

            System.out.println();

        }

    }

}
```

Output

```
Console
<terminated> prog24 [Java Application] E:\Programmers Hub\JDK\bin\javaw.exe (Apr 23, 2022, 1:09:08 PM – 1:09:10 PM)
Enter The Peak
5

        X
      X X
    X X X
  X X X X
X X X X X
  X X X X
    X X X
      X X
        X
```

Problem 28- Write a program to print the following pattern

```
    x
   xxx
  xxxxx
 xxxxxxx
xxxxxxxxx
```

```
package Questions;
import java.util.Scanner;
public class prog25 {

    public static void main(String[] args) {

        Scanner sc = new Scanner(System.in);
        System.out.println("Enter The Height ");
        int height=sc.nextInt();
        int i,j,k,l,m;

        for(i=1;i<=height;i++)
        {
            for(j=0;j<height-i;j++)
                System.out.print("  ");
            for(k=0;k<i;k++)
                System.out.print("X ");
            for(l=1;l<i;l++)
                System.out.print("X ");

            System.out.println();

        }
        }
    }

}
```

Output

```
Console
<terminated> prog25 [Java Application] E:\Programmers Hub\JDK\bin\javaw.exe (Apr 23, 2022, 3:09:33 PM – 3:09:34 PM)
Enter The Height
5
        X
      X X X
    X X X X X
  X X X X X X X
X X X X X X X X X
```

Problem 29- Write a program to print the following pattern

```
xxxxxxxxx
 xxxxxxx
  xxxxx
   xxx
    x
```

```
package Questions;
import java.util.Scanner;
public class prog25 {

    public static void main(String[] args) {
```

```
        Scanner sc = new Scanner(System.in);
        System.out.println("Enter The Height ");
        int height=sc.nextInt();
        int i,j,k,l,m;
        for(i=1;i<=height;i++)
        {
            for(l=0;l<i;l++)
                System.out.print("  ");
            for(j=0;j<=height-i;j++)
                System.out.print("X ");
            for(k=0;k<height-i;k++)
                System.out.print("X ");
            System.out.println();

        }
     }

}
```

Output

```
Enter The Height
5
  X X X X X X X X X
    X X X X X X X
      X X X X X
        X X X
          X
```

Problem 30- Write a program to print the following pattern

```
      x
     xxx
    xxxxx
   xxxxxxx
  xxxxxxxxx
   xxxxxxx
    xxxxx
     xxx
      x
```

```
package Questions;
import java.util.Scanner;
public class prog25 {

    public static void main(String[] args) {
        Scanner sc = new Scanner(System.in);
        System.out.println("Enter The Height ");
        int height=sc.nextInt();
        int i,j,k,l,m;

        for(i=1;i<=height;i++)
        {
            for(j=0;j<height-i;j++)
                System.out.print("  ");
            for(k=0;k<i;k++)
                System.out.print("X ");
            for(l=1;l<i;l++)
                System.out.print("X ");

            System.out.println();

        }
        for(i=1;i<=height-1;i++)
        {
            for(l=0;l<i;l++)
```

```
                System.out.print("  ");
            for(j=0;j<=height-1-i;j++)
                System.out.print("X ");
            for(k=0;k<height-1-i;k++)
                System.out.print("X ");

           System.out.println();

        }

      }

}
```

Output