

TABLE OF CONTENTS



Sl.	Contents	Pages
1	OOP Basic Concepts	03
2	Basic of Java	04
3	Structure of the Java Program	04
4	Foundations of Java Programming: Variables, String, Control Structure, Array	07
5	Java Conceptual Problem Solving : Examples	14
6	Class and Objects	22
7	Constructur in OOP	28
8	Methods	30
9	Understanding the 'Static' Keyword	32
10	Understanding the 'this' Keyword	37
11	Inheritance : Extending Classes for Reusability	38
12	Understanding the 'Super' Keyword	44
13	Final Keyword	50
14	Encapsulation : Data protection Mechanism	53
15	Polymorphism : Versatile Object Behavior	56
16	Abstraction : Simplified for Beginners	59
17	Interfaces : Exploring Contractual Implement	62
18	Exception Handling : Managing Errors Effectively	65
19	Different types of Exception	66
20	References	75

Othera®
Esomeprazole

20 mg Tablet
40 mg Tablet
40 mg IV Injection

1st BI-PHASIC
BIO-EQUIVALENT
Esomeprazole in Bangladesh

Sl.	Contents	Pages
20	<p>Miscellaneous of OOP</p> <ul style="list-style-type: none"> - Upcasting and downcasting - Command Line Arguments - Wrapper class, Inner class, String - Class Relationships - Binding - Object Cloning, Argument Passing - Blocks in Java, Array of objects 	
	I/O in Java	
	Garbage Collection in Java	
	Java JDK, JRE, JVM	
	Collections in Java	
	Generics in Java	
	Java Lambda Expressions	
	Multithreading	
	JDBC	
	References.	

OOP basic concepts

OOPs refers to Object-Oriented Programming. It is the programming paradigm that is defined using objects.

Think about of an object as something from real world - a cat, a person or a book. Each object has characteristics (like color, size, or title) and things it can do (drive, speak, or turn pages).

In OOP, we write code by creating these objects and letting them interact, helping us model real-world scenarios in our programs in a structured and efficient way.

Advantages of OOP

- OOP's is very helpful in solving very complex level of problems.
- Highly complex programs can be created, handled and maintained easily using OOP.
- OOPs, promote code reuse, thereby reducing redundancy.
- OOPs, ~~is based on~~ also helps to hide the unnecessary details with the help of Data Abstraction.
- Polymorphism offers a lot of flexibility in OOPs.

Basics of JAVA



Java is a high level, object-oriented Programming language developed by Sun Microsystems (now owned by Oracle Corporation).

Why Use Java ?

- Java Works on different platforms. (Windows, Mac, Linux..)
- It is fast, Secure and powerful.
- It is easy to learn and Simple to use.
-

Structure of the Java Program:

class Example {

 public static void main (String args [])

{

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

}

}

Explanation of above program.

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

All java applications begin execution from `main()` function. (This is just like C/C++)

⇒ The `public` keyword is an access specifier, which allows the programmers to control the visibility of class members.

⇒ The keyword `static` allows `main()` to be called without having to instantiate a particular instance of the class.

⇒ The keyword ~~static~~ `void` simply tells the compiler that `main()` does not return a value.

⇒ As stated, `main()` is the method called when a java application begins. Keep in mind that java is case sensitive. `Main` is different from `main`.

⇒ `String args []` declares a parameter named `args`, which is an array of instances of the class `String`. (Arrays are ~~simil~~ collections of ~~simil~~ objects). Objects of type `String` store character strings. In this case, `args` receives any command-line arguments present when the program is executed.

Write a program to initialize an integer variable num to 100, then display the value of num & num*2.

Solution:

class Test

```
{   public static void main (String args[])
    {
        int num ;
        num = 100 ;
        System.out.println ("This is num :" + num) ;
        num = num * 2 ;
        System.out.print ("The value of num*2 :") ;
        System.out.println (num) ;
    }
}
```

Output:

This is num : 100

The value of num * 2 : 200.

Foundations of Java Programming

Java Variables

Variables are containers for storing data values.

In Java, there are different types of Variables,
for example -

- String
- int
- float
- char
- boolean

Syntax

type VariableName = Value;

Example

```
public class Main {  
    public static void main (String args[])  
    {  
        int num = 59;  
        float numFloat = 59.29f;  
        char c = 'D';  
        boolean flag = true;  
        String text = "CSE";  
    }  
}
```

Take two variable & print them.

Code:

```
public class Test {
```

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

```
        String first = "Leading";
```

```
        String Second = "University";
```

```
        String fullName = first + " " + Second;
```

```
        System.out.println (first + " " + Second);
```

OR

```
        System.out.println (fullName);
```

}

* One Value to Multiple Variables

```
int x, y, z;
```

```
x = y = z = 59;
```

Identifiers

All Java Variables must be identified with unique names. These unique names are called identifiers.

Example:

```
int minutesPerHour = 60; // Good
```

```
int m = 60; // OK, but not so easy to understand
```

// What m actually is.

Othera®
Esomeprazole
20 mg tablet
50 mg tablet
100 mg tablet

1st

BI-PHASIC
BIO-EQUIVALENT
Esomeprazole 20mg Bangladesh

8

Data Types

Data types divided into two groups:

- * **Primitive data types** - includes byte, short, int, long, float, double, Boolean & char.
- * **Non-primitive data types** - Such as String, Array.

Code :

```
public class Fun{  
    public static void main(String args[]){  
        char var1 = 65, var2 = 66;  
        System.out.println(var1);  
        System.out.print(var2);  
    }  
}
```

⇒ Output ?

Java String

String Text = "LU Cse";

String length : Text.length(); [Text is Variable]

Find char : Text.indexOf("Cse");

String Concatenation : The + operator used between String

More String Methods:

// toUpperCase()

// toLowerCase()

Java Math

The java Math Class has many methods that allows us to perform mathematical tasks on numbers. For example,

Math.Max(x,y)

Math.min(x,y)

Math.Sqrt(49)

Math.abs(x)

Math.random()

Java If... Else

Syntax:

```
if (Condition1){  
    ---  
}  
else if (Condition2){  
    ---  
}  
else {  
    ---  
}
```

Short Hand If... Else

There is also Short-Hand if else, which is known as the "ternary Operator" because it consists of three Operands.

Syntax:

```
Variable = (Condition) ? expressionTrue :  
                    expressionFalse;
```

Code:

```
int age = 21;
```

```
String adult = (age > 18) ? "Yes, adult": "No";
```

```
System.out.println(adult);
```



Java Loop

```
for (int i = 1; i <= 10; i++) {  
    System.out.println(i);  
}
```

Java Break/Continue

```
for (int i = 0; i < 10; i++) {  
    if (i == 5) continue;  
    if (i == 7) break;  
}
```

Othera
Esomeprazole

20 mg Tablets
40 mg Tablets
40 mg N. Injection

1st BI-PHASIC
BIO-EQUIVALENT
Esomeprazole in Bangladesh

12

Java Array

Code :

```
String[] cars = {"BMW", "Ford", "F1"};  
int[] myNum = {45, 50, 55, 59, 60};  
// access the elements.  
System.out.println(cars[0]);  
// Change an array element.  
cars[0] = "Mazda";  
// array length  
cars.length;  
// Loop through an Array  
for (int i=0; i < cars.length; i++){  
    System.out.println(cars[i]);  
}  
// Multidimensional Array  
int[][] numbers = {{1,2,3,4}, {5,6,7}};  
for (int i=0; numbers.length; ++i){  
    for (int j=0; j < numbers[i].length; ++j){  
        System.out.println(numbers[i][j]);  
    }  
}
```

* Java Conceptual Problem Solving



Java Code to Create Pyramid & Pattern

Example-1: Program to print half pyramid using *

```
*
* *
* * *
* * * *
* * * *
```

Code:

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

Ex-2

```
1
1 2
1 2 3
1 2 3 4
1 2 3 4 5
```

Othera
omeprazole

Ex-3

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

1st BI-PHASIC
BIO-EQUIVALENT
Esomeprazole in Bangladesh

Ex-3: Solution.

char last = 'E', alphabet = 'A';

for (int i=1; i <= (last-'A'+1); i++) {

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

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

 }

 System.out.println();

}

Ex-4: Inverted half pyramid Using *.

for (int i=rows; i >= 1; --i) {

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

 System.out.print("* ");

}

 System.out.println();

}

Output:

```
*****
 ****
  ***
   *
  *
```

Task

1 2 3 4 5

1 2 3 4

1 2 3

1 2

1



Example-6 : Program to print full pyramid using *

```
int rows=5, k=0;  
for (int i=1; i<=rows; ++i){  
    k=0;  
    for (int space=1; space<=rows-i; ++space){  
        System.out.print(" ");  
    }  
    while (k!=2*i-1){  
        System.out.print("*");  
        k++;  
    }  
    System.out.println();  
}  
}
```

Output:

```
*  
* * *  
* * * * *  
* * * * * * *  
* * * * * * * *
```

Task

Ex-9: Print Pascal's triangle

```
int rows=6, coef=1;  
for (int i=0; i<rows; i++){  
    for (int space=1; space<rows-i; ++space){  
        System.out.print(" ");  
    }  
    for (int j=0; j<=i; j++){  
        if (j==0 || i==0)  
            coef=1;  
        else  
            coef = coef * (i-j+1)/j;  
        System.out.printf("%4d", coef);  
    }  
    System.out.println();  
}
```

Output:

```
1  
1 1  
1 2 1  
1 3 3 1  
1 4 6 4 1  
1 5 10 10 5 1
```



Example - 8 : Inverted full pyramid using *

Code :

```

int rows = 5;
for (int i = rows; i >= 1; --i) {
    for (int space = 1; space <= rows - i; ++space) {
        System.out.print(" ");
    }
    for (int j = i; j <= 2 * i - 1; ++j) {
        System.out.print("*");
    }
    for (int j = 0; j < i - 1; ++j) {
        System.out.print("*");
    }
    System.out.println();
}

```

Example - 10 : Print Floyd's Triangle.

Code :

```

int rows = 4, num = 1; number = 1;
for (int i = 1; i <= rows; i++) {
    for (int j = 1; j <= i; j++) {
        System.out.print(number + " ");
        number++;
    }
    System.out.println();
}

```

Output :

1
2
3
4
5
6
7
8
9
10

Othera®
Esomeprazole
20 mg Tablet
40 mg Tablet
40 mg for injection

1st BI-PHASIC
BIO-EQUIVALENT
Esomeprazole in Bangladesh

18