**FUNDAMENTALS OF JAVA 02 [Assignment Questions/Answers] 27 JAN’ 23**

ANS 01:

**Statically typed languages:**

A language is statically typed if the type of a variable is known at compile time. For some languages this means that you as the programmer must specify what type each variable is; other languages (e.g.: Java, C, C++) offer some form of type inference, the capability of the type system to deduce the type of a variable (e.g.: OCaml, Haskell, Scala, Kotlin).

The main advantage here is that all kinds of checking can be done by the compiler, and therefore a lot of trivial bugs are caught at a very early stage.

Examples: C, C++, Java, Rust, Go, Scala

int age = 18;

String name = “Hydar Abbas”

int marks = 100;

float avg = 10.5f;

**Dynamically typed languages:**

A language is dynamically typed if the type is associated with run-time values, and not named variables/fields/etc. This means that you as a programmer can write a little quicker because you do not have to specify types every time (unless using a statically-typed language with type inference).

Examples: Perl, Ruby, Python, PHP, JavaScript, Erlang

age = 18;

name “Navin Reddy”;

marks = 100;

ANS 02:

**Java Variables:**

A variable is a container which holds the value while the Java program is executed. A variable is assigned with a data type.

Variable is a name of memory location. There are three types of variables in java:

* Local
* Instance
* static.

**int** [typesOfData] **marks** [variableName] = **99** [value];

ANS 03:

Syntax to assign a value to a variable-

**String** [typesOfData] **name** [variableName] = **“Abbas”** [value];

If we have a variable name ‘name’ and we assign a value ‘Abbas’

ANS 04:

Data types specify the different sizes and values that can be stored in the variable. There are two types of data types in Java:

**Primitive data types:** The primitive data types include

* Boolean
* Char
* Byte
* Short
* Int
* Long
* Float
* Double

ANS 05:

**Identifiers:**

An identifier is a name given to a package, class, interface, method or variables,all identifiers have different names.

public class Test

{

public static void main(String[] args)

{

int a = 20;

}

}

In the above java code, we have 5 identifiers namely :

Test : class name.

main : method name.

String : predefined class name.

args : variable name.

a : variable name.

ANS 06:

**Operators in JAVA:**

* Arithmetic Operator
* Relational Operator
* Logical Operator
* Assignment Operator
* Unary Operator
* Bitwise Operator

ANS 07:

**Increment & Decrement Operators:**

Increment operator is used to increment a value by 1. There are two varieties of increment operator:

Post-Increment: Value is first used for computing the result and then incremented.

Pre-Increment: Value is incremented first and then the result is computed.

EXAMPLE-

//INCREMENTAL OPERATOR

//Let's see what exactly happens when we use these operators with the help of a code.

import java.io.\*;

class GFG {

public static void main (String[] args) {

//1. Post-Increment Operator

int a = 5;

int b = 7;

int c = a++ + b; //Here 'a' will not incremented immediately, a++ will still return value 5.

// c = 5 + 7 and this will evaluate to 12.

System.out.println("Post- Increment \n c = "+ c);

//2. Pre-Increment Operator

int A = 5;

int B = 7;

int C = ++A + B; //Here 'a' will be incremented immediately, and ++a will return value 6.

// C = 6 + 7 and this will evaluate to 13.

System.out.println("Pre- Increment \n C = "+ C);

//Example-

int m = 1, n = 2;

int o = m++ + n + ++m; // It goes like m++ = 1, n = 2, ++m = 1+ incremented 'm' from m++

// = 1 + (1+m) = 1+ (1+1) = 3

// o = 1 + 2 + 3 = 6

System.out.println("Example \n o = "+ o);

}

}

Decrement operator is used for decrementing the value by 1. There are two varieties of decrement operators.

Post-decrement: Value is first used for computing the result and then decremented.

Pre-decrement: Value is decremented first and then the result is computed.

//DECREMETAL OPERATORS

//Let's see what exactly happens when we use these operators with the help of a code.

import java.io.\*;

class GFG {

public static void main (String[] args) {

//1. Post-Decrement Operator

int a = 5;

int b = 7;

int c = a-- + b; //Here 'a' will not decremented immediately, a-- will still return value 5.

// c = 5 + 7 and this will evaluate to 12.

System.out.println("Post- Decrement \n c = "+ c);

//2. Pre-Decrement Operator

int A = 5;

int B = 7;

int C = --A + B; //Here 'a' will be decremented immediately, and --a will return value 4.

// C = 4 + 7 and this will evaluate to 11.

System.out.println("Pre- Decrement \n C = "+ C);

//Example-

int m = 3, n = 2;

int o = m-- + n + --m; // It goes like m-- = 3, n = 2, --m = (decremented 'm' from 'm--') - 1;

// = (m -1) - 1 = (3 -1) -1 = 1

// o = 3 + 2 + 1 = 6

System.out.println("Example \n o = "+ o);

}

}