Object Oriented Programming

NAME = M. Sameed Khan
ID = 6843
Semester = 8th
BS (SE)

Q1 How many variables are being supported by java justify your answer with the help of Java Coded example for each variable?

ANSWER

* JAVA Variables =
Variable are Containers for storing data values.
In Java, There are different types of Variables which are given below:

- String - stores text, such as "Hello". String values are Sorrounded by double quotes
- int - Stores integers (whole numbers), without decimals, Such as 123 or -123
- float - Stores floating point numbers, with decimals Such as 19.99 or -19.99.
- Char - Stores Single characteristics, Such as 'a' or 'B'. Char. Values

are Sorrounded by Single quotes.

- boolen - Store values with two states: true or false.

* Declearing (creating) Variables

To create variable, you must Speciefy the type and assign it a value

- Syntax

type variable = Value;

where type is one of Java's types (such as int or string), and variable is the name of the variable (such as x or name) The equal sign is used to assign values to the variable.

To create a variable that should store text, look at the following example:

* EXAMPLE 1:

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

    String name = "John";
    System.out.println(name);
  }
}
```

Result:

John

* EXAMPLE: 2.

Creat a variable called myNum of type int and assign it the value 15;

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

        int myNum;
        myNum = 15;
        System.out.println(myNum);
        }
}
```

Output:

15

* EXAMIZE 3:

Chang the value of myNum from 15 to 20

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

        int myNum = 15;
        myNum = 20; // myNum is now 20
        System.out.println(myNum);
        }
}
```

Output:

20

* Declare Many Variables:-
To declare more than one variable of the same type, use a comma-seperated list:

* EXAMPLE 3:-
Myclass.java

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

    int x=5, y=6, z=50;
System.out.println(x+y+z);
  }
}
```

Output:-
61

The general rules for constructing names for variables (unique identifiers) are:

- Name can contain letters, digits, underscores, and dollar signs
- Name must begin with a letter
- Name should start with a lowercase letter and it cannot contain whitespace.
- Name can also begin with $ and _ (but we will not use it in this tutorial)
- Name are case sensitive ("myVar" and "myvar" are different variables)
- Reserved words (like Java keywords, such as int or boolen) cannot be used as names.

Q2 Why "if" is used in Java. Justify your answer with the help of Java Coded example and explain it in detail.

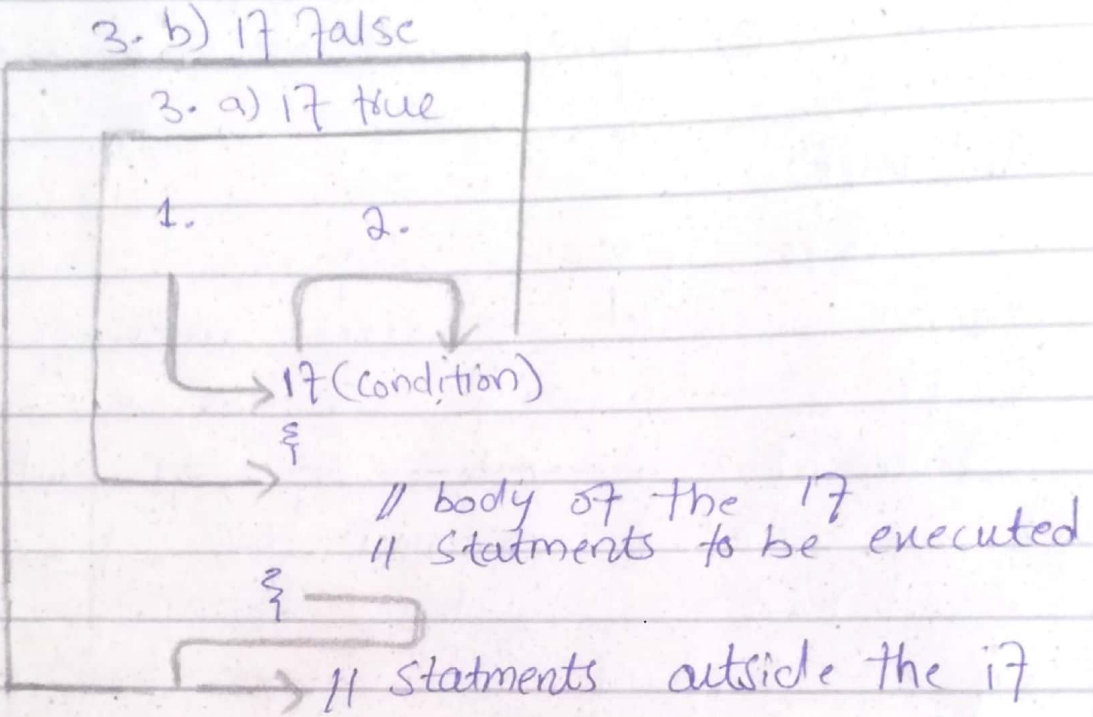
ANSWER

The Java if statment is the most simple decision making statment. it is used to decide whether a certain statment or block of statments will be executed or not ie if a certain Condition is true then a block of statment is executed otherwise not. Control falls into the if block.

Decision Making in Java helps to write decision driven statments and execute a particular set of code based on certain conditions.

The Java if Statment is the most simple decision-making statment. It is used to decide whether a certain statment or block of statments will be executed or not i.e if a certain condition is true than a block statment is executed other wise not.

If statment

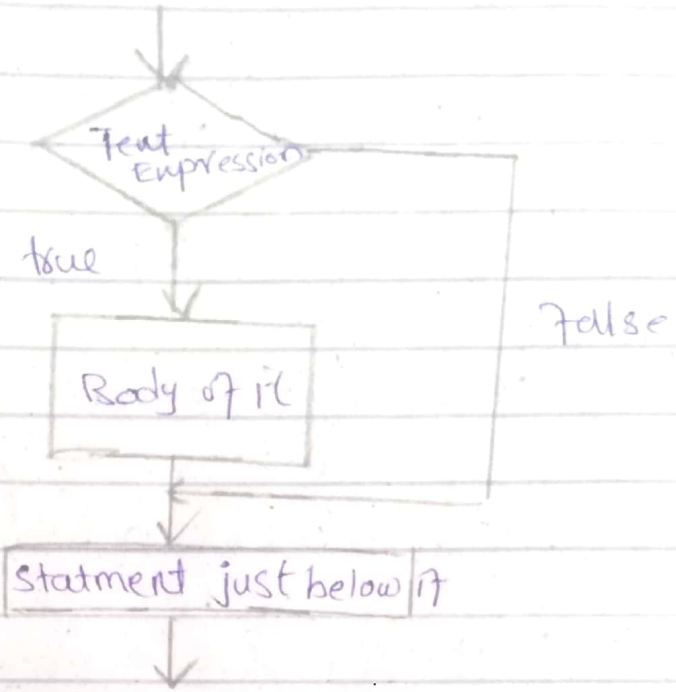


* Syntax:

```
if (condition)
{
    // statments to execute if
    // condition is true
}
```

1. Control falls into the if block
2. The flow jumps to Condition
3. Condition is tested
   a). If Condition yields true, goto step 4.
   b). If Condition yields false, goto step 5.
4. The if-block or the body inside the if is executed
5. Flow steps out of the if block

* Flow chart :



* EXAMPLE 1 :-

```
// Java program to illustrate if
class If Demo {
    public static void main (String args[])
{
  int i = 10
  if (i < 15)
     System.out.println ("10 is less than 15");
// This statment will be executed
// as if considers one statment by default
System.out.println ("Outside if-block");
  }
}
```

Output :

```
10 is less than 15
Out side if-block
```

Q3 Why "if else if" is used in Java justify your answer with the help of Java Coded example and explain in details

ANSWER:

Java if... else (if - then - else) Statment

... The if statment executes a certain section of code if the test expression is evaluated to true.... Statment inside the body of else block are executed if the test expression is evaluated to false. This is known as the if - then - else statment in Java.

- Java if, if... else Statment

In this tutorial, you will learn about Control flow statments in Java using Java if and if... else statments with the help of examples

In Computer programming, it's often desirable to execute a certain section of code based upon whether the specified condition is [true] or [false] (which is known only during the run time). For such cases, Control flow statments are used.

* Java if ... else (if-then-else) Statment

The if statment executes a certain section of code if the test expression is evaluated to true. However, if the test expression is evaluated to false, it does nothing.

In this case, we can use an optional else block. Statments inside the body of else block are executed if the test expression is evaluated to false This is known as if-then-else statment in Java

This Syntax of the if-then-else Statment is

* EXAMPLE:

```
if (expression) {
    // Codes
}
else {
    // Some other Code
}
```

Here, our program will do one task (task inside if block) if the test expression is true and another task (task inside else block) if the test expression is false.

* How if statment works:

Expression is true

```
int test = 5;
if (test < 10)
{
   // body of if
}
else
{
   // body of else
}
```

Expression is false

```
int test = 5;
if (test > 10)
{
   // body of if
}
else
{
   // body of else
}
```

. Working of Java if-else statment

* EXAMPLE 2:-

```
class Ladder {
   public static void main(String[] args {
int number = 0;
// checks if number is greater than 0.
if (number > 0){
   System.out.println("The number is positive.");
}
// checks if number is less than 0.
else if (number < 0) {
   System.out.println("The number is negative.");
}
else {
  System.out.println("The number is 0.");
  }
}
}
```

Output:

The number is 0.

In the above example, we are checking whether the `number` is positive, negative or zero. Here, we have two test expressions:

- `(number > 0)` - checks if the `number` is greater than `0`
- `number < 0` - checks if the `number` is less than `0`.

Here, the value of `number` is `0`. So both the test expression evaluates to `false`. Hence the statment inside the body of `else` is executed.

Q4 What are loops, why they are used in java and how many types of loops are being supported by java explain in detail?

ANSWER:-

The Java for loop is a control flow statment that iterates a part of the programs multiple times.

The Java while loop is a control flow statment that executes a part of the programs repeatedly on

the basis of given boolen Condition.

* Three types of loops in Java.

Loops are used to execute a set of statments repeatedly untill a particular Conditon is satisfied. In Java we have three types of basic loops: for, while and do-while. In this tutorial we will learn how to use "for loop" in Java.

1. For loop
2. while loop
3. Do/while loop

(1) For Loop :

statment 1 is executed (one time) before the execution of code block. Statment 2 defines the Condition for executing the code block. Statment 3 is executed (every time) after the code block has been executed.

(2) while loop :

Java while loop is a Control flow statment that allows code to be executed repeatedly based on a given Boolean Condition evaluates to true then we will execute the body of the loop and go to update expression.

(3) Do/while loop :

The do/while loop is a variant of the while loop The loop will execute the code block once, before checking if the Condition is true, then it will repeat the loop as long as the Condition is true.

EXAMPLE :-

Example of Simple For loop

```
class ForLoopExample {
    public static void main(String args[]) {
        for(int i=10; i>1; i--){
            System.out.println("The value of i is: "+i);
        }
    }
}
```

The Output of this program is :

The value of i is : 10
The value of i is : 9
The value of i is : 8
The value of i is : 7
The value of i is : 6
The value of i is : 5
The value of i is : 4
The value of i is : 3
The value of i is : 2

In the above program
int i=1 is initialization expression
i >1 is Condition (Boolean expression)
i - Decrement operation

Q5: Write 3's table in decremented form in Java which takes input from user write Java coded program and explain in details?

ANSWER:-

Java program to generate multiplication Table

In this program, you'll learn to generate multiplication table of a given number. This is done by using a for and a while loop in Java.

EXAMPLE:-

```
Public class Multiplication table {
  public static void main (string [] args) {
    int num = 3;
   for (int i = 1 ; i <= 10; ++i)
   {
     system.out.printf("%d * %d = %d \n", num, i,
                                 num* i);
   }
  }
}
```

Output

3 * 1 = 3
3 * 2 = 6
3 * 3 = 9
3 * 4 = 12
3 * 5 = 15
3 * 6 = 18
3 * 7 = 21
3 * 8 = 24
3 * 9 = 27
3 * 10 = 30

In the above program, unlike a for loop, we have to increment the value of [i] inside the body of the loop.

Though both programs are for loop technically correct, it is better to use for loop in this case its because the number of Iteration (from 1 to 10) is known.