Assignment 2 Solution

Theory Questions with Answers:

1. What are JAVA tokens? Name the types of them.
Ans: JAVA Token are the smallest individual entity used in a program.
It is divided into five (5) categories given below:

- Keywords
- Identifiers (or Variables)
- Literals (or Constants)
- Separators (or Punctuators)
- Operators

2. What are literals? Name different types of literals.
Ans: The quantity which does not change its value during the program execution is called Literal or constant. JAVA allows several kinds of literals: -

- Integer Literal
- Floating Literals
- Boolean Literals
- Character-Literal
- String-Literals
- Null Literal

3. What are identifiers? State the rules for declaring identifiers.
Ans: Those quantities which change their values during the program execution are called identifiers or variables. It is the name of memory area used to store data values.
Rules for assigning Identifier:

- Keyword cannot be used as an Identifier or variable or data member.
- Spacing and special characters cannot be used, rather use underscore (_).
- Uppercase and lowercase are distinct in Java environment.

4. What is an operand, operator and expression?
Ans: Operand: - These are the values on which certain operations are performed. It can be a variable or a constant.
Operator: - Operators are the symbol which specifies the type of operation to be performed on the operands.
Expression: - It is a valid combination of operators & operands.

5. What is Type Conversion?
Ans: Typecasting, or type conversion, is a method of changing a variable from one data type to another.

6. Differentiate between implicit and explicit type conversion.
Ans:

Implicit type conversion	***Explicit Type Conversion***
• Converting a lower data type into a higher one is called **implicit** type conversion.	• Converting a higher data type into a lower one is called **explicit** type casting
• It is also known as **widening type casting** or **casting down**.	• It is also known as **narrowing type casting** or **casting up**.
• It is done automatically	• It is done manually by the programmer.

7. Differentiate between unary and binary operator?

Ans:

Unary Operators	***Binary Operators***
• The operators which require only one operand to work upon are called unary operators.	• The operators which require two operands to work upon are called binary operators.
• There are two unary operators:- Increment and decrement operators.	• There are mainly three types of binary operators- Arithmetic, Relational and Logical Operators.

8. Differentiate between % and / operators.

Ans:

% operator	***/ operator***
• This operator is called modulus operator.	• This operator is called division operator.
• This operator is used to find modulus remainder of the two numbers.	• This operator is used to find quotient of the two numbers.

9. Differentiate between = and == operator.

Ans:

= operator	***== operator***
• This is an assignment operator.	• This is a relational operator.
• This operator is used to assign the value to the variable.	• This operator is used to compares two values.

10. Explain the ternary operator with its syntax and example.

Ans: The operator which uses three operands to work upon are called ternary operators.

Syntax: variable= condition? Values True: False;

Example: m= a>b? a: b;

11. Differentiate between Arithmetic and Relational operator.

Ans:

Arithmetic Operator	***Relational Operator***
• The operators which perform mathematical operations are called arithmetic operators.	• The operators which are used to compare two operands are called relational operators or comparison operators.
• They return the operated value.	• They always return Boolean values.
• Examples: +, -, *, /, %	• Examples: >, <, <=, >= , ==, !=

12. What are escape sequences?

Ans: Some characters cannot be typed directly and must be written as "escape-sequences".

Example: \t -> Tab or **\n-> Newline.**

13. Using ternary operator, print "Hello" when g is 10 otherwise print "Bye".

```
System.out.println(g==10?"Hello": "Bye");
```

14. If x=10, find the value of y :-

i. y= ++x + x-- - ++x + --x => **y = 11+11+11+10 y=> 43**

ii. y= --x - x-- + x-- => **y = 9 -9+8 => 8**

iii. y= x++ + ++x + --x => **y = 10 + 12 + 11 => 33**