**2.1 Fill in the blanks**

a) A(n) **left brace {** begins the body of every method, and a(n) **right brace }** ends the body of every method.  
b) You can use the **if** statement to make decisions.  
c) **//** begins an end-of-line comment.  
d) **Spaces, tabs, and newlines** are called white space.  
e) **Keywords (or reserved words)** are reserved for use by Java.  
f) Java applications begin execution at method **main**.  
g) Methods **System.out.print**, **System.out.println**, and **System.out.printf** display information in a command window.

**2.2 True/False**

a) **False** – Comments are ignored by the compiler and do not appear in program output.  
b) **True** – Every variable in Java must have a type when declared.  
c) **False** – Java is case-sensitive, so number and NuMbEr are different variables.  
d) **False** – The remainder operator % can be used with both integer and floating-point operands.  
e) **False** – The operators \*, /, and % have higher precedence than + and -.

**2.7 Fill in the blanks**

a) **Comments** are used to document a program and improve its readability.  
b) A decision can be made in a Java program with a(n) **if statement**.  
c) Calculations are normally performed by **assignment** statements.  
d) The arithmetic operators with the same precedence as multiplication are **division (/)** and **modulus (%)**.  
e) When parentheses in an arithmetic expression are nested, the **innermost** set of parentheses is evaluated first.  
f) A location in the computer’s memory that may contain different values at various times throughout the execution of a program is called a(n) **variable**.

**2.9 True/False**

a) **False** – Operator precedence determines order, not left-to-right.  
b) **True** – All listed variable names are valid.  
c) **False** – Operator precedence affects evaluation order.  
d) **True** – Variable names cannot start with numbers.

### ****2.10 Output of given statements****

a) x = 2  
b) Value of 2 + 2 is 4  
c) x = (prints "x =" without newline)  
d) 5 = 5

### ****2.11 Modified Variables****

a) **Yes** – p = i + j + k + 7; modifies p.  
b) **No** – System.out.println(...) does not modify variables.  
c) **No** – Just prints text.  
d) **Yes** – value = input.nextInt(); assigns user input to value.

### ****2.12 Correct Java statements****

Correct ones:

* **a) y = a \* x \* x \* x + 7;**
* **d) y = (a \* x) \* x \* x + 7;**
* **e) y = a \* (x \* x \* x) + 7;**

### ****2.13 Operator Precedence****

a) 7 + 3 \* 6 / 2 - 1 = 7 + 18 / 2 - 1 = 7 + 9 - 1 = 15  
b) 2 % 2 + 2 \* 2 - 2 / 2 = 0 + 4 - 1 = 3  
c) 3 \* 9 \* (3 + (9 \* 3 / 3)) = 3 \* 9 \* (3 + 9) = 3 \* 9 \* 12 = 324