**Chapter 2.1:**

**1.** { }

**2.** if

**3.** //

**4.** Space, tab, and newline

**5.** keywords

**6.** main

**7.** console.log, console.warn and console.error

**2.2**

**a)** False.\* Comments do not cause the computer to print the text after the // on the screen; they are ignored by the JavaScript engine during execution.

**b)** False.\* In JavaScript, variables do not need to be given a type when declared; JavaScript is dynamically typed, and the type is determined by the value assigned.

**c)** False.\* JavaScript considers number and NuMbEr to be different variables because variable names are case-sensitive.

**d)** False.\* The remainder operator (%) can be used with both integer and floating-point operands in JavaScript, not only with integers.

**e)** False.\* The arithmetic operators \*, /, and % have higher precedence than + and –, so they do not all have the same level of precedence.

**2.3** **Write Statements**

**a)** int c, thisIsAVariable, q76354, number;

**b)** System.out.print("Enter an integer: ");

**c)** int value = input.nextInt();

**d)** System.out.println("This is a Java program");

**e)** System.out.printf("%s%n%s%n", "This is a Java", "program");

**f)** if (number != 7) System.out.println("The variable number is not equal to 7");

**2.4**

**a)** Error: Semicolon after if causes the condition to be empty. Corrected: if (c < 7) System.out.println("c is less than 7");

**b)** Error: => should be >= for "greater than or equal to". Corrected: if (c >= 7) System.out.println("c is equal to or greater than 7");

**2.5**

**a)** // This program calculates the product of three integers

**b)** Scanner input = new Scanner(System.in);

**c)** int x, y, z, result;

**d)** System.out.print("Enter the first integer: ");

**e)** x = input.nextInt();

**f)** System.out.print("Enter the second integer: ");

**g)** y = input.nextInt();

**h)** System.out.print("Enter the third integer: ");

**i)** z = input.nextInt();

**j)** result = x \* y \* z;

**k)** System.out.printf("Product is %d%n", result);

**2.7 Fill in the blanks**

**a)** Comments

**b)** if statement

**c)** assignment

**d)** division (/) and remainder (%)

**e)** innermost

**f)** variable

**2.8 Write Java Statements**

a) System.out.print("Enter an integer: ");

b) a = b \* c;

c) // This program performs a sample payroll calculation

**2.9**

**a)** False - Java operators are evaluated based on precedence, not strictly left to right.

**b)** True

**c)** False - Without parentheses, precedence rules apply, not strictly left-to-right evaluation.

**d)** True

**2.10 Output**

Assuming x = 2, y = 3:

**a)** x = 2

**b)** Value of 2 + 2 is 4

**c)** x = (no newline)

**d)** 5 = 5

**2.11** **Modified Variables**

**a)** Yes - p is modified

**b)** No - Just printing

**c)** No - Just printing

**d)** Yes - value is modified

**2.12** **Correct Statements for y = ax³ + 7**

a) Yes - y = a \* x \* x \* x + 7;

b) No - Incorrect: multiplies x by (x + 7)

c) No - Incorrect: multiplies x by (x + 7)

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

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

f) No - Incorrect: multiplies x by (x² + 7)

**2.13 Order of Evaluation**

**a)** x = 7 + 3 \* 6 / 2 - 1

**.** / (18 / 2 = 9)

**.** (7 + 9 = 16)

**.** (16 - 1 = 15)

**.** x = 15

**b)** x = 2 % 2 + 2 \* 2 - 2 / 2

**.** % (2 % 2 = 0)

**.** (2 \* 2 = 4)

**.** / (2 / 2 = 1)

**.** (0 + 4 = 4)

**.** (4 - 1 = 3)

**.** x = 3

**c)** x = (3 \* 9 \* (3 + (9 \* 3 / (3))))

**.** Innermost / (9 \* 3 = 27, 27 / 3 = 9)

**.** (3 + 9 = 12)

**.** (9 \* 12 = 108)

**.** (3 \* 108 = 324)

**.** x = 324

**2.14**

**a)** System.out.println("1 2 3 4");