

Primitive Types – MCQ Part B

This quiz has 12 questions.

1. Consider the following code segment.

```
int j = 10;
int k = 8;
j += 2;
k += j;
System.out.print(j);
System.out.print(" ");
System.out.println(k);
```

What is printed when the code segment is executed?

- (A) 2 2
- (B) 2 10
- (C) 10 8
- (D) 12 12
- (E) 12 20

2. Consider the following code segment.

```
int x = 0;
x++;
x += 1;
x = x + 1;
x -= -1;
System.out.println(x);
```

What is printed when the code segment has been executed?

- (A) 0
- (B) 1
- (C) 2
- (D) 3
- (E) 4

3. Consider the following code segment.

```
int num = 5;
num *= 2;
num %= 6;
```

What is the value of num after the code segment is executed?

- (A) 1
- (B) 2
- (C) 4
- (D) 6
- (E) 10

4. Consider the following code segment.

```
int x = /* initial value not shown */;
int y = /* initial value not shown */;
int z = x;
z /= y;
z += 2;
```

Which of the following best describes the behavior of the code segment?

- (A) It sets Z to 2.
- (B) It sets Z to x.
- (C) It sets Z to $(1 / y) + 2$.
- (D) It sets Z to $(x / y) + 2$.
- (E) It sets Z to $(x + 2) / y$.

5. Consider the following code segment, which is intended to calculate the average of two quiz scores.

```
double avg = 15 + 20;
avg /= 2;
```

Which of the following best describes the behavior of the code segment?

- (A) The code segment stored 17 in avg because 17 is the result of the integer division of 35 by 2.
- (B) The code segment stores 18 in avg because 18 is the result of the integer division of 35 by 2.
- (C) The code segment stores 17.5 in avg because 17.5 is the result of the floating point division of 35.0 by 2.
- (D) The code segment does not compile because int values cannot be assigned to double values.
- (E) The code segment does not compile because a double value cannot be divided by an int value.

Primitive Types – MCQ Part B

6. Consider the following code segment.

```
int x = 4;
int y = 6;
x -= y;
y += x;
```

Which of the following best describes the behavior of the code segment?

- (A) The value of `x` and the value of `y` have not been changed.
- (B) Both the value of `x` and the value of `y` have been decreased.
- (C) Both the value of `x` and the value of `y` have been increased.
- (D) The value of `x` has been decreased and the value of `y` has been increased.
- (E) The value of `x` has been increased and the value of `y` has been decreased.

7. Consider the following code segment.

```
double d = 0.25;
int i = 3;
double diff = d - i;
System.out.print((int)diff - 0.5);
```

What is printed as a result of executing the code segment?

- (A) -2
- (B) -2.5
- (C) -3
- (D) -3.25
- (E) Nothing is printed because an `int` cannot be subtracted from a `double`.

8. Consider the following code segment.

```
double a = 7;
int b = (int) (a / 2);
double c = (double) b / 2;
System.out.print(b);
System.out.print(" ");
System.out.print(c);
```

What is printed as a result of executing the code segment?

- (A) 3 1
- (B) 3 1.0
- (C) 3 1.5
- (D) 3.5 1.5
- (E) 3.5 1.75

9. Consider the following code segment.

```
double p = 10.6;
double n = -0.2;
System.out.println((int) (p + 0.5));
System.out.print((int) (n - 0.5));
```

What is printed as a result of executing the code segment?

- (A) 10
-1
- (B) 10
0
- (C) 11
-1
- (D) 11
0
- (E) 11
1

10. Consider the following code segment, which is intended to display 0.5.

```
int num1 = 5;
int num2 = 10;
double ans = num1 / num2;
System.out.print(ans);
```

Which of the following best described the error, if any, in the code segment?

- (A) There is no error and the code works as intended.
- (B) The code should have cast the expression `num1 / num2` to `double`.
- (C) The code should have cast either `num1` or `num2` in the expression `num1 / num2` to `double`.
- (D) The code should have declared `ans` as an `int`.
- (E) The code should have initialized `num1` to 5.0 and `num2` to 10.0.

Primitive Types – MCQ Part B

11. A store sells rope only in whole-foot increments.

Given three lengths of rope, in feet, the following code segment is intended to display the minimum length of rope, in feet, that must be purchased so that the rope is long enough to be cut into the three lengths. For example, for lengths of 2.8 feet, 3 feet, and 5 feet, the minimum length of rope that must be purchased is 11 feet. For these inputs, the code segment should display 11. As another example, for lengths of 1.1 feet, 3.2 feet, and 2 feet, the minimum length of rope that must be purchased is 7 feet. For these inputs, the code segment should display 7.

```
double len1;  
double len2;  
double len3;  
double total = len1 + len2 + len3;  
int minLength = (int) (total + 0.5);  
System.out.print(minLength);
```

Which of the following best describes the behavior of the code segment?

- (A) The code segment works as intended for all nonnegative values of `len1`, `len2`, and `len3`.
- (B) The code segment does not work as intended for some values of `len1`, `len2`, and `len3`. The code segment works only when the sum of the three lengths is an integer or the decimal part of the sum of the three lengths is greater than or equal to 0.5.
- (C) The code segment does not work as intended for some values of `len1`, `len2`, and `len3`. It could be corrected by casting `len1`, `len2`, and `len3` to `int` before adding them together and assigning the sum to `minLength`.
- (D) The code segment does not work as intended for some values of `len1`, `len2`, and `len3`. It could be corrected by declaring `minLength` as a `double` and casting it to an `int` in the `System.out.print` statement.
- (E) The code segment does not work as intended for any values of `len1`, `len2`, and `len3`. It returns the sum of 0.5 and the three lengths for all values of `len1`, `len2`, and `len3`.

12. Consider the following code segment, which is intended to display 6.0.

```
double fact1 = 1 / 2;  
double fact2 = 3 * 4;  
double product = fact1 * fact2;  
System.out.println(product);
```

Which of the following best describes the error, if any, in the code segment?

- (A) There is no error and the code works as intended.
- (B) Either the numerator or the denominator of the fraction `1 / 2` should be cast as `double`.
- (C) The expression `fact1 * fact2` should be cast as `double`.
- (D) The expressions `1 / 2` and `3 * 4` should both be cast as `double`.
- (E) The variables `fact1` and `fact2` should both be declared as `int`.