

1.- Which of the following Java operators can be used with boolean variables? (Choose all that apply.)

Opciones: 1. ==

2. +

3. -

4. !

5. %

6. <=

7. Cast with (boolean)

Respuesta: Opciones 1, 4 y 7.

Explicación: El == puede ser usado tanto en operaciones de primitivos como en referencias a objetos, El ! solo puede ser usado por booleanos y si se puede hacer un cast para que el valor que regrese sea un booleano.

2.- What data type (or types) will allow the following code snippet to compile? (Choose all that apply.)

```
byte apples = 5;  
short oranges = 10;  
_____ bananas = apples + oranges;
```

Opciones: 1. int

2. long

3. boolean

4. double

5. short

6. Byte.

Respuesta: Opciones 1, 2 y 4.

Explicación: int(32), long(64) y double(64) son primitivos que tienen un mayor valor que short(16) y byte(8) por lo tanto pueden ser usados.

3.- What change, when applied independently, would allow the following code snippet to compile? (Choose all that apply.)

```
long ear = 10;  
int hearing = 2 * ear;
```

- Opciones:
1. No change; it compiles as is.
 2. Cast ear on line 4 to int.
 3. Change the data type of ear on line 3 to short.
 4. Cast 2 * ear on line 4 to int.
 5. Change the data type of hearing on line 4 to short.
 6. Change the data type of hearing on line 4 to long.

Respuesta: Opciones 2, 3, 4 y 6.

Explicación: 2 si se hace un cast ear a int funcionaria por que el tipo de dato de hearing es int, 3 esta caso funcionaria ya que el short ear sería menor que int por lo que el resultado de la operación si entraría en el int, 4 como en el caso 2 hacer el cast funcionaria, 4 funcionaria ya que hearing tendría el mismo tipo de dato que ear.

4.- What is the output of the following program?

```
1: public class CandyCounter {  
2:     static long addCandy(double fruit, float vegetables) {  
3:         return (int)fruit+vegetables;  
4:     }  
5:  
6: public static void main(String[] args) {  
7:     System.out.print(addCandy(1.4, 2.4f) + "- ");  
8:     System.out.print(addCandy(1.9, (float)4) + "-");  
9:     System.out.print(addCandy((long)(int) (short)2, (float)4)); } }
```

Opciones:

1. 4-6-6.0
2. 3-5-6
3. 3-6-6

Academia Java Xideral
Respuestas Examen Semana 01
Nombre: Luis Wonen Olvera Vásquez
Fecha: 25/11/2022

4. 4-5-6

5. The code does not compile because of line 9.

6. None of the above

Respuesta: Opción 6.

Explicación: El código no compilaría ya que, aunque en la línea 3 tenemos un cast a int, vegetables sigue siendo float por lo tanto, el tipo de dato que se mantiene es float.

5. What are the unique outputs of the following code snippet? (Choose all that apply.)

```
int a = 2, b = 4, c = 2;  
System.out.println(a > 2 ? --c : b++);  
System.out.println(b = (a!=c ? a : b++));  
System.out.println(a > b ? b < c ? b : 2 : 1);
```

Opciones: 1. 1

2. 2

3. 3

4. 4

5. 5

6. 6

7. The code does not compile

Respuesta: Opciones 1, 4 y 5.

Explicación: Primero imprime 4 ya que $a > 2$ es falso, pero, aunque b aumenta no lo hará hasta la siguiente línea ya que uso un b++. Después imprime 5 ya que a y c son iguales nos dará un false por lo tanto b++, y como b aumento a 5 en la línea pasada se asignará 5 y se cancelara el b++ de esa línea. Finalmente, $a=2$, $c=2$ y $b=5$ por lo tanto imprime 1 ya que b no es menor que c y a no es mayor que b.

6. Given the following code snippet, what is the value of the variables after it is executed? (Choose all that apply.)

```
int ticketsTaken = 1;  
int ticketsSold = 3;  
ticketsSold += 1 + ticketsTaken++;  
ticketsTaken *= 2;  
ticketsSold += (long)1;
```

- Opciones:
1. ticketsSold is 8
 2. ticketsTaken is 2
 3. ticketsSold is 6
 4. ticketsTaken is 6
 5. ticketsSold is 7
 6. ticketsTaken is 4
 7. The code does not compile.

Respuesta: Opciones 3 y 6.

Explicación: En la línea 3 ticketsTaken aumenta a 2, por lo que a ticketsSold se le suman 3 por lo tanto su valor es 6, en la línea 4 ticketsTaken es multiplicado por 2 por lo que $(2 * 2) = 4$.

7. What is the output of the following code snippet? (Choose all that apply.)

```
3: int temperature = 4;  
4: long humidity = -temperature + temperature * 3;  
5: if (temperature >= 4)  
6:     if (humidity < 6) System.out.println("Too Low");  
7:     else System.out.println("Just Right");  
8: else System.out.println("Too High");
```

- Opciones:
1. Too Low
 2. Just Right
 3. Too High

Academia Java Xideral
Respuestas Examen Semana 01
Nombre: Luis Wonen Olvera Vásquez
Fecha: 25/11/2022

4. A NullPointerException is thrown at runtime.
5. The code will not compile because of line 7.
6. The code will not compile because of line 8.

Respuesta: Opción 2.

Explicación: En las primeras dos líneas el valor de temperature es 4 mientras que humidity es 8 por precedencia de operaciones, por lo tanto, cumple con el primer if en cuando el segundo no lo cumple, ya que 6 no es menor que 8 por lo que imprime Just Right.

8. Which statements, when inserted independently into the following blank, will cause the code to print 2 at runtime? (Choose all that apply.)

```
int count = 0;
BUNNY: for(int row = 1; row <=3; row++)
    RABBIT: for(int col = 0; col <3 ; col++) {
        if((col + row) % 2 == 0)
            _____;
        count++;
    }
System.out.println(count);
```

- Opciones:
1. break BUNNY
 2. break RABBIT
 3. continue BUNNY
 4. continue RABBIT
 5. break
 6. continue
 7. None of the above, as the code contains a compiler error

Respuesta: Opciones 2, 3 y 5.

Explicación: La opción que yo ocuparía en este caso es usar el break ya que dentro count aumentaría a 1 y volvería a entrar ya que el valor de row=2 y col=0, por lo tanto count aumentaría a 2, por lo que row= 3 y col=0, por lo que entraría en el for de nuevo ya que aún cumple con que sea <= 3, como count ya vale 2 lo mejor sería

usar el `break` para que conserve el valor consideramos que `break rabbit` es equivalente por lo que también es una respuesta correcta y al final `continue bunny` no nos afectaría ya que en ese ciclo `for` no aumenta `count`.

9. What is the output of the following code snippet?

```
2: boolean keepGoing = true;
3: int result = 15, meters = 10;
4: do {
5:     meters--;
6:     if(meters==8) keepGoing = false;
7:     result -= 2;
8: } while keepGoing;
9: System.out.println(result);
```

Opciones: 1. 7

2. 9

3. 10

4. 11

5. 15

6. The code will not compile because of line 6.

7. The code does not compile for a different reason.

Respuesta: Opción 7.

Explicación: en la línea 8 `keepGoing` no tiene los paréntesis por lo que no hace la comparación de un booleano.

10. What is the output of the following code snippet? (Choose all that apply.)

```
9: int w = 0, r = 1;
10: String name = "";
11: while(w < 2) {
12:     name += "A";
13:     do {
14:         name += "B";
```

Academia Java Xideral
Respuestas Examen Semana 01
Nombre: Luis Wonen Olvera Vásquez
Fecha: 25/11/2022

```
15:         if(name.length()>0) name += "C";  
16:         else break;  
17:     } while (r <=1);  
18:     r++; w++; }  
19:     System.out.println(name);
```

- Opciones:
1. ABC
 2. ABCABC
 3. ABCABCABC
 4. Line 15 contains a compilation error.
 5. Line 18 contains a compilation error.
 6. The code compiles but never terminates at runtime.
 7. The code compiles but throws a NullPointerException at runtime.

Opciones: Opción 6.

Explicación: En la línea 18 nos podemos percatar que r++, es aumentada afuera del ciclo do while por lo tanto el ciclo se seguirá ejecutando ya que no aumentará r.

11. What is output by the following code? (Choose all that apply.)

```
1: public class Fish {  
2:     public static void main(String[] args) {  
3:         int numFish = 4;  
4:         String fishType = "tuna";  
5:         String anotherFish = numFish + 1;  
6:         System.out.println(anotherFish + " " + fishType);  
7:         System.out.println(numFish + " " + 1);  
8:     }}
```

- Opciones:
1. 4 1
 2. 5
 3. 5 tuna
 4. 5tuna
 5. 51tuna
 6. The code does not compile.

Respuesta: Opción 6.

Explicación: En la línea 5 nos encontramos con esto, `String anotherFish = numFish + 1;`, podemos notar que esta queriendo asignar un `int` a un `String` y esto no se puede por lo tanto, el código no compilara.

12. What is the result of the following code?

```
7: StringBuilder sb = new StringBuilder();  
8: sb.append("aaa").insert(1, "bb").insert(4, "ccc");  
9: System.out.println(sb);
```

- Opciones:
1. abbaaccc
 2. abbaccca
 3. bbaaaccc
 4. bbaaccca
 5. An empty line
 6. The code does not compile.

Respuesta: Opción 2.

Explicación: Primero "aaa" es agregado a sb, después en la posición 1 después de la primera "a" se agrega "bb", por lo tanto, quedaría "abbaa", finalmente en la posición 4 después de la segunda "a" agregamos "ccc" por lo que al final imprimiría "abbaccca".

13. What is the result of the following code?

```
12: int count = 0;  
13: String s1 = "java";  
14: String s2 = "java";  
15: StringBuilder s3 = new StringBuilder("java");  
16: if (s1 == s2) count++;  
17: if (s1.equals(s2)) count++;  
18: if (s1 == s3) count++;  
19: if (s1.equals(s3)) count++;  
20: System.out.println(count);
```


Academia Java Xideral
Respuestas Examen Semana 01
Nombre: Luis Wonen Olvera Vásquez
Fecha: 25/11/2022

- Opciones:
1. 0
 2. 1
 3. 2
 4. 3
 5. 4
 6. An exception is thrown.
 7. The code does not compile.

Respuesta: Opción 7.

Explicación: Todo va bien hasta la línea 18, en esta podemos ver el uso de una comparación entre s1 que es un String y s3 que es un StringBuilder, algo que no se permite por lo que el código no compilará.

14. What is the result of the following code?

```
public class Lion {  
    public void roar(String roar1, StringBuilder roar2) {  
        roar1.concat("!!!");  
        roar2.append("!!!");  
    }  
    public static void main(String[] args) {  
        String roar1 = "roar";  
        StringBuilder roar2 = new StringBuilder("roar");  
        new Lion().roar(roar1, roar2);  
        System.out.println(roar1 + " " + roar2);  
    }  
}
```

- Opciones:
1. roar roar
 2. roar roar!!!
 3. roar!!! roar
 4. roar!!! roar!!!
 5. An exception is thrown.
 6. The code does not compile.

Respuesta: Opción 2.

Explicación: El problema de este bloque de código es que cuando se llama el método `roar`, como `roar1` es un `String` no le concatena ya que `String` es inmutable por lo que no se modifica.

15. Which of the following can replace line 4 to print "avaJ"? (Choose all that apply.)

```
3: var puzzle = new StringBuilder("Java");  
4: // INSERT CODE HERE  
5: System.out.println(puzzle);
```

Opciones: 1. `puzzle.reverse();`
2. `puzzle.append("vaJ$").substring(0, 4);`
3. `puzzle.append("vaJ$").delete(0, 3).deleteCharAt(puzzle.length() - 1);`
4. `puzzle.append("vaJ$").delete(0, 3).deleteCharAt(puzzle.length());`
5. None of the above

Respuesta: Opciones 1 y 3.

Explicación: Al ver el código pensé que solo la opción 1 sería correcta pero en la opción 3 dice que agregara "vaJ\$" y borrará los primeros 3, por lo que tendríamos "avaJ\$" y luego borra el ultimo carácter por lo tanto al final también obtendríamos "avaJ".