

1. Which of the following are legal entry point methods that can be run from the command line? (Choose all that apply.)

- A. `private static void main(String[] args)`
- B. `public static final main(String[] args)`
- C. `public void main(String[] args)`
- D. `public static final void main(String[] args)`
- E. `public static void main(String[] args)`
- F. `public static main(String[] args)`

A, C, D, E

La opción E es la firma canónica del método `main()`. Necesitas memorizarla. La opción D es una forma alternativa con el final redundante.

La opción A es incorrecta porque el método `main()` debe ser público. Las opciones B y F son incorrectas porque el método `main()` debe tener un tipo de retorno `void`. La opción C es incorrecta porque el método `main()` debe ser estático.

2. Which answer options represent the order in which the following statements can be assembled into a program that will compile successfully? (Choose all that apply.)

X: `class Rabbit {}`
Y: `import java.util.*;`
Z: `package animals;`

- A. X, Y, Z
- B. Y, Z, X
- C. Z, Y, X
- D. Y, X
- E. Z, X
- F. X, Z
- G. None of the above

C, D, E

Las sentencias `package` e `import` son ambas opcionales. Si ambas están presentes, el orden debe ser `package`, luego `import`, y luego `class`. La opción A es incorrecta porque `class` está antes que `package` e `import`. La opción B es incorrecta porque `import` está antes que `package`. La opción F es incorrecta porque `class` está antes que `package`.

3. Which of the following are true? (Choose all that apply.)

```
public class Bunny {  
    public static void main(String[] x) {  
        Bunny bun = new Bunny();  
    }  
}
```

- A. Bunny is a class.
- B. bun is a class.
- C. main is a class.
- D. Bunny is a reference to an object.
- E. bun is a reference to an object.
- F. main is a reference to an object.
- G. The main() method doesn't run because the parameter name is incorrect.

A, E

Bunny es una clase, lo que se puede ver en la declaración: public class Bunny. La variable bun es una referencia a un objeto. El método main() es el punto de entrada estándar a un programa. La opción G es incorrecta porque el tipo de parámetro importa, no el nombre del parámetro.

4. Which of the following are valid Java identifiers? (Choose all that apply.)

- A. _
- B. _helloWorld\$
- C. true
- D. java.lang
- E. Public
- F. 1980_s
- G. _Q2_

B, E, G

La opción A no es válida porque no se permite un solo guion bajo. La opción C no es un identificador válido porque true es una palabra reservada de Java. La opción D no es válida porque no se permite un punto (.) en los identificadores. La opción F no es válida porque el primer carácter no es una letra, un signo de dólar (\$) o un guion bajo (_). Las opciones B, E y G son válidas porque contienen solo caracteres válidos.

5. Which statements about the following program are correct? (Choose all that apply.)

```
2: public class Bear {  
3:   private Bear pandaBear;  
4:   private void roar(Bear b) {  
5:     System.out.println("Roar!");  
6:     pandaBear = b;  
7:   }  
8:   public static void main(String[] args) {  
9:     Bear brownBear = new Bear();  
10:    Bear polarBear = new Bear();  
11:    brownBear.roar(polarBear);  
12:    polarBear = null;  
13:    brownBear = null;  
14:    System.gc(); } }
```

- A. The object created on line 9 is eligible for garbage collection after line 13.
- B. The object created on line 9 is eligible for garbage collection after line 14.
- C. The object created on line 10 is eligible for garbage collection after line 12.
- D. The object created on line 10 is eligible for garbage collection after line 13.
- E. Garbage collection is guaranteed to run.
- F. Garbage collection might or might not run.
- G. The code does not compile.

A, D, F

La recolección de basura nunca está garantizada, lo que hace que la opción F sea correcta y la opción E incorrecta. A continuación, la clase compila y se ejecuta sin problemas, por lo que la opción G es incorrecta. El objeto Bear creado en la línea 9 es accesible hasta la línea 13 a través de la variable de referencia brownBear, que es la opción A. El objeto Bear creado en la línea 10 es accesible a través de la referencia polarBear y la referencia brownBear.pandaBear. Después de la línea 12, el objeto sigue siendo accesible a través de brownBear.pandaBear. Sin embargo, después de la línea 13, ya no es accesible porque brownBear ya no es accesible, lo que convierte a la opción D en la respuesta final.

6. Assuming the following class compiles, how many variables defined in the class or method are in scope on the line marked on line 14?

```
1: public class Camel {  
2:   { int hairs = 3_000_0; }  
3:   long water, air=2;  
4:   boolean twoHumps = true;  
5:   public void spit(float distance) {  
6:     var path = "";  
7:     { double teeth = 32 + distance++; }  
8:     while(water> 0){  
9:       int age = twoHumps ? 1 : 2;  
10:      short i=-1;  
11:      for(i=0; i<10; i++) {  
12:        var Private = 2;  
13:      }  
14:      // SCOPE  
15:    }  
16:  }  
17: }
```

A. 2

B. 3

C. 4

D. 5

E. 6

F. 7

G. None of the above

F

Para resolver este problema, necesitas rastrear las llaves {} y ver cuándo las variables entran y salen de alcance. Las variables en las líneas 2 y 7 solo están en alcance para un bloque de una sola línea. La variable en la línea 12 solo está en alcance para el bucle for. Ninguna de estas está en alcance en la línea 14. Por el contrario, las tres variables de instancia en las líneas 3 y 4 están disponibles en todos los métodos de instancia. Además, las variables en las líneas 6, 9 y 10 están disponibles ya que el método y el bucle while todavía están en alcance. Esto es un total de 7 variables, que es la opción F.

7. Which are true about this code? (Choose all that apply.)

```
public class KitchenSink {  
    private int numForks;  
  
    public static void main(String[] args) {  
        int numKnives;  
        System.out.print("""  
            "# forks = " + numForks +  
            " # knives = " + numKnives +  
            "# cups = 0""");  
    }  
}
```

- A. The output includes: # forks = 0.
- B. The output includes: # knives = 0.
- C. The output includes: # cups = 0.
- D. The output includes a blank line.
- E. The output includes one or more lines that begin with whitespace.
- F. The code does not compile.

C, E

Lo primero que hay que reconocer es que esto es un bloque de texto y el código dentro de """" es solo texto. Las opciones A y B son incorrectas porque las variables numForks y numKnives no se utilizan. Esto es conveniente ya que numKnives no está inicializada y no compilaría si se hiciera referencia a ella. La opción C es correcta ya que coincide con el texto. La opción D es incorrecta porque el bloque de texto no tiene una línea en blanco al final. Finalmente, la opción E también es una respuesta ya que "# knives" tiene sangría.

8. Which of the following code snippets about var compile without issue when used in a method? (Choose all that apply.)

- A. var spring = null;
- B. var fall = "leaves";
- C. var evening = 2; evening = null;
- D. var night = Integer.valueOf(3);
- E. var day = 1/0;
- F. var winter = 12, cold;
- G. var fall = 2, autumn = 2;
- H. var morning = ""; morning = null;

B, D, E, H

Una var no puede ser inicializada con un valor nulo sin un tipo, pero se le puede asignar un valor nulo más tarde si el tipo subyacente no es un primitivo. Por estas razones, la opción H es correcta, pero las opciones A y C son incorrectas. Las opciones B y D son correctas ya que los tipos subyacentes son String e Integer, respectivamente. La opción E es correcta ya que esta es una expresión numérica válida. Puede que sepas que dividir entre cero produce una excepción en tiempo de ejecución, pero la pregunta era solo sobre si el código compilaba. Finalmente, las opciones F y G son incorrectas ya que var no se puede usar en una asignación de múltiples variables.

9. Which of the following are correct? (Choose all that apply.)

- A. An instance variable of type float defaults to 0.
- B. An instance variable of type char defaults to null.
- C. A local variable of type double defaults to 0.0.
- D. A local variable of type int defaults to null.
- E. A class variable of type String defaults to null.
- F. A class variable of type String defaults to the empty string "".
- G. None of the above.

E

Las opciones C y D son incorrectas porque las variables locales no tienen valores por defecto. La opción A es incorrecta porque float debe tener un punto decimal. La opción B es incorrecta porque los primitivos no tienen como valor por defecto nulo. La opción E es correcta y la opción F incorrecta porque los tipos de referencia en las variables de clase tienen como valor por defecto nulo.

10. Which of the following expressions, when inserted independently into the blank line, allow the code to compile? (Choose all that apply.)

```
public void printMagicData() {  
    var magic = _____;  
    System.out.println(magic);  
}
```

A. 3_1

B. 1_329_.0

C. 3_13.0_

D. 5_291._2

E. 2_234.0_0

F. 9___6

G. _1_3_5_0

A, E, F

Un guion bajo (_) se puede colocar en cualquier literal numérico, siempre que no esté al principio, al final o junto a un punto decimal (.). Los guiones bajos incluso se pueden colocar uno junto al otro. Por estas razones, las opciones A, E y F son correctas. Las opciones B y D son incorrectas ya que el guion bajo (_) está junto a un punto decimal (.). Las opciones C y G son incorrectas porque un guion bajo (_) no se puede colocar al principio o al final del literal.

11. Given the following two class files, what is the maximum number of imports that can be removed and have the code still compile?

// Water.java

```
package aquarium;  
public class Water { }
```

// Tank.java

```
package aquarium;  
import java.lang.*;  
import java.lang.System;  
import aquarium.Water;  
import aquarium.*;  
public class Tank {  
    public void print(Water water) {  
        System.out.println(water); } }
```

- A. 0
- B. 1
- C. 2
- D. 3
- E. 4
- F. Does not compile

E

Las dos primeras importaciones se pueden eliminar porque java.lang se importa automáticamente. Las dos importaciones siguientes se pueden eliminar porque Tank y Water están en el mismo paquete, lo que hace que la opción correcta sea la E.

12. Which statements about the following class are correct? (Choose all that apply.)

```
1: public class ClownFish {  
2:   int gills = 0, double weight=2;  
3:   { int fins = gills; }  
4:   void print(int length = 3) {  
5:     System.out.println(gills);  
6:     System.out.println(weight);  
7:     System.out.println(fins);  
8:     System.out.println(length);  
9:   } }
```

- A. Line 2 generates a compiler error.
- B. Line 3 generates a compiler error.
- C. Line 4 generates a compiler error.
- D. Line 7 generates a compiler error.
- E. The code prints 0.
- F. The code prints 2.0.
- G. The code prints 2.
- H. The code prints 3.

A, C, D

La línea 2 no compila ya que solo se debe especificar un tipo, lo que hace que la opción A sea correcta. La línea 3 compila sin problemas ya que declara una variable local dentro de un inicializador de instancia que nunca se usa. La línea 4 no compila porque Java no admite la configuración de valores de parámetros de método predeterminados, lo que hace que la opción C sea correcta. Finalmente, la línea 7 no compila porque fins está en alcance y es accesible solo dentro del inicializador de instancia en la línea 3, lo que hace que la opción D sea correcta.

13. Given the following classes, which of the following snippets can independently be inserted in place of INSERT IMPORTS HERE and have the code compile? (Choose all that apply.)

```
package aquarium;  
public class Water {  
    boolean salty = false;  
}
```

```
package aquarium.jellies;  
public class Water {  
    boolean salty = true;  
}
```

```
package employee;  
INSERT IMPORTS HERE  
public class WaterFiller {  
    Water water;  
}
```

A. import aquarium.*;

B. import aquarium.Water;

import aquarium.jellies.*;

C. import aquarium.*;

import aquarium.jellies.Water;

D. import aquarium.*;

import aquarium.jellies.*;

E. import aquarium.Water;

import aquarium.jellies.Water;

F. None of these imports can make the code compile.

A, B, C

La opción A es correcta porque importa todas las clases en el paquete aquarium incluyendo aquarium.Water. Las opciones B y C son correctas porque importan Water por nombre de clase. Dado que importar por nombre de clase tiene prioridad sobre los comodines, estos compilan. La opción D es incorrecta porque Java no sabe cuál de las dos clases Water con comodín usar. La opción E es incorrecta porque no se puede especificar el mismo nombre de clase en dos importaciones.

14. Which of the following statements about the code snippet are true?

(Choose all that apply.)

```
3: short numPets = 5L;  
4: int numGrains = 2.0;  
5: String name = "Scruffy";  
6: int d = numPets.length();  
7: int e = numGrains.length;  
8: int f = name.length();
```

- A. Line 3 generates a compiler error.
- B. Line 4 generates a compiler error.
- C. Line 5 generates a compiler error.
- D. Line 6 generates a compiler error.
- E. Line 7 generates a compiler error.
- F. Line 8 generates a compiler error.

A, B, D, E

La línea 3 no compila porque el sufijo L hace que el valor literal sea un long, que no se puede almacenar dentro de un short directamente, lo que hace que la opción A sea correcta. La línea 4 no compila porque int es un tipo integral, pero 2.0 es un valor literal double, lo que hace que la opción B sea correcta. La línea 5 compila sin problemas. Las líneas 6 y 7 no compilan porque numPets y numGrains son ambas primitivas, y se pueden llamar métodos solo en tipos de referencia, no en valores primitivos, lo que hace que las opciones D y E sean correctas, respectivamente. Finalmente, la línea 8 compila porque hay un método length() definido en String.

15. Which of the following statements about garbage collection are correct?

(Choose all that apply.)

- A. Calling `System.gc()` is guaranteed to free up memory by destroying objects eligible for garbage collection.
- B. Garbage collection runs on a set schedule.
- C. Garbage collection allows the JVM to reclaim memory for other objects.
- D. Garbage collection runs when your program has used up half the available memory.
- E. An object may be eligible for garbage collection but never removed from the heap.
- F. An object is eligible for garbage collection once no references to it are accessible in the program.
- G. Marking a variable `final` means its associated object will never be garbage collected.

C, E, F

En Java, no hay garantías sobre cuándo se ejecutará la recolección de basura. La JVM es libre de ignorar las llamadas a `System.gc()`. Por esta razón, las opciones A, B y D son incorrectas. La opción C es correcta ya que el propósito de la recolección de basura es reclamar la memoria utilizada. La opción E también es correcta porque un objeto puede que nunca se recolecte como basura, como si el programa finaliza antes de que se ejecute la recolección de basura. La opción F es correcta y es el medio principal por el cual los algoritmos de recolección de basura determinan si un objeto es elegible para la recolección de basura. Finalmente, la opción G es incorrecta ya que marcar una variable como `final` significa que es constante dentro de su propio alcance. Por ejemplo, una variable local marcada como `final` será elegible para la recolección de basura después de que finalice el método, suponiendo que no haya otras referencias al objeto que existan fuera del método.

16. Which are true about this code? (Choose all that apply.)

```
var blocky = ""  
  squirrel \s  
  pigeon \  
  termite"";  
System.out.print(blocky);
```

- A. It outputs two lines.
- B. It outputs three lines.
- C. It outputs four lines.
- D. There is one line with trailing whitespace.
- E. There are two lines with trailing whitespace.
- F. If we indented each line five characters, it would change the output.

A, D

La opción A es correcta. Hay dos líneas. Una comienza con `squirrel` y la otra comienza con `pigeon`. Recuerda que una barra invertida significa omitir el salto de línea. La opción D también es correcta ya que `\s` significa mantener el espacio en blanco. En un bloque de texto, se ignora la sangría incidental, lo que hace que la opción F sea incorrecta.

17. What lines are printed by the following program? (Choose all that apply.)

```
1: public class WaterBottle {  
2:   private String brand;  
3:   private boolean empty;  
4:   public static float code;  
5:   public static void main(String[] args) {  
6:     WaterBottle wb = new WaterBottle();  
7:     System.out.println("Empty = " + wb.empty);  
8:     System.out.println("Brand = " + wb.brand);  
9:     System.out.println("Code = " + code);  
10:  }}
```

A. Line 8 generates a compiler error.

B. Line 9 generates a compiler error.

C. Empty =

D. Empty = false

E. Brand =

F. Brand = null

G. Code = 0.0

H. Code = 0f

D, F, G

El código compila y se ejecuta sin problemas, por lo que las opciones A y B son incorrectas. Un campo booleano se inicializa a false, lo que hace que la opción D sea correcta con Empty = false siendo impreso. Las referencias a objetos se inicializan a null, no a la cadena vacía, por lo que la opción F es correcta con Brand = null siendo impreso. Finalmente, el valor por defecto de los números de punto flotante es 0.0. Aunque los valores float se pueden declarar con un sufijo f, no se imprimen con un sufijo f. Por estas razones, la opción G es correcta y Code = 0.0 es impreso.

18. Which of the following statements about var are true? (Choose all that apply.)

- A. A var can be used as a constructor parameter.
- B. The type of a var is known at compile time.
- C. A var cannot be used as an instance variable.
- D. A var can be used in a multiple variable assignment statement.
- E. The value of a var cannot change at runtime.
- F. The type of a var cannot change at runtime.
- G. The word var is a reserved word in Java.

B, C, F

A var no se puede usar para un constructor o parámetro de método o para una instancia o variable de clase, lo que hace que la opción A sea incorrecta y la opción C correcta. El tipo de una var se conoce en tiempo de compilación, y el tipo no se puede cambiar en tiempo de ejecución, aunque su valor puede cambiar en tiempo de ejecución. Por estas razones, las opciones B y F son correctas, y la opción E es incorrecta. La opción D es incorrecta, ya que var no está permitido en declaraciones de variables múltiples. Finalmente, la opción G es incorrecta, ya que var no es una palabra reservada en Java.

19. Which are true about the following code? (Choose all that apply.)

```
var num1 = Long.parseLong("100");  
var num2 = Long.valueOf("100");  
System.out.println(Long.max(num1, num2));
```

- A. The output is 100.
- B. The output is 200.
- C. The code does not compile.
- D. num1 is a primitive.
- E. num2 is a primitive.

A, D

Las dos primeras líneas proporcionan una forma de convertir un String en un número. La primera es un primitivo long y la segunda es un objeto de referencia Long, lo que hace que la opción D sea una de las respuestas. El código es correcto y el máximo es 100, que es la opción A.

20. Which statements about the following class are correct? (Choose all that apply.)

```
1: public class PoliceBox {  
2:   String color;  
3:   long age;  
4:   public void PoliceBox() {  
5:     color = "blue";  
6:     age = 1200;  
7:   }  
8:   public static void main(String []time) {  
9:     var p = new PoliceBox();  
10:    var q = new PoliceBox();  
11:    p.color = "green";  
12:    p.age = 1400;  
13:    p = q;  
14:    System.out.println("Q1="+q.color);  
15:    System.out.println("Q2="+q.age);  
16:    System.out.println("P1="+p.color);  
17:    System.out.println("P2="+p.age);  
18:  }}
```

- A. It prints Q1=blue.
- B. It prints Q2=1200.
- C. It prints P1=null.
- D. It prints P2=1400.
- E. Line 4 does not compile.
- F. Line 12 does not compile.
- G. Line 13 does not compile.
- H. None of the above.

C

La clave para notar es que la línea 4 no define un constructor, sino un método llamado PoliceBox(), ya que tiene un tipo de retorno void. Este método nunca se ejecuta durante la ejecución del programa, y color y age se les asignan los valores por defecto null y 0L, respectivamente. Las líneas 11 y 12 cambian los valores para un objeto asociado con p, pero luego, en la línea 13, la variable p se cambia para apuntar al objeto asociado con q, que todavía tiene los valores por defecto. Por esta razón, el programa imprime Q1=null, Q2=0, P1=null, y P2=0, lo que hace que la opción C sea la única respuesta correcta.

21. What is the output of executing the following class?

```
1: public class Salmon {  
2:   int count;  
3:   { System.out.print(count+"-");}  
4:   { count++;}  
5:   public Salmon() {  
6:     count = 4;  
7:     System.out.print(2+"-");  
8:   }  
9:   public static void main(String[] args) {  
10:    System.out.print(7+"-");  
11:    var s = new Salmon();  
12:    System.out.print(s.count+"-"); } }
```

A. 7-0-2-1-

B. 7-0-1-

C. 0-7-2-1-

D. 7-0-2-4-

E. 0-7-1-

F. The class does not compile because of line 3.

G. The class does not compile because of line 4.

H. None of the above.

D

Comenzamos con el método main(), que imprime 7- en la línea 10. A continuación, se crea una nueva instancia de Salmon en la línea 11. Esto llama a los dos inicializadores de instancia en las líneas 3 y 4 para que se ejecuten en orden. El valor por defecto de una variable de instancia de tipo int es 0, por lo que 0- se imprime a continuación y a count se le asigna un valor de 1. A continuación, se llama al constructor. Esto asigna un valor de 4 a count e imprime 2-. Finalmente, la línea 12 imprime 4-, ya que ese es el valor de count. Poniéndolo todo junto, tenemos 7-0-2-4-, lo que hace que la opción D sea la respuesta correcta.

22. Given the following class, which of the following lines of code can independently replace INSERT CODE HERE to make the code compile? (Choose all that apply.)

```
public class Price {  
    public void admission() {  
        INSERT CODE HERE  
        System.out.print(amount);  
    }  
}
```

A. int Amount = 0b11;

B. int amount = 9L;

C. int amount = 0xE;

D. int amount = 1_2.0;

E. double amount = 1_0_.0;

F. int amount = 0b101;

G. double amount = 9_2.1_2;

H. double amount = 1_2_.0_0;

C, F, G

Primero, Ob es el prefijo para un valor binario, y 0x es el prefijo para un valor hexadecimal. Estos valores se pueden asignar a muchos tipos primitivos, incluyendo int y double, lo que hace que las opciones C y F sean correctas. La opción A es incorrecta porque nombrar la variable Amount hará que la llamada System.out.print(amount) en la siguiente línea no compile. La opción B es incorrecta porque 9L es un valor long. Si el tipo se cambiara a long amount = 9L, entonces compilaría. La opción D es incorrecta porque 1_2.0 es un valor double. Si el tipo se cambiara a double amount = 1_2.0, entonces compilaría. Las opciones E y H son incorrectas porque el guion bajo (_) aparece junto al punto decimal (.), lo cual no está permitido. Finalmente, la opción G es correcta, y el uso del guion bajo y la asignación son válidos.

23. Which statements about the following class are true? (Choose all that apply.)

```
1: public class River {  
2:   int Depth = 1;  
3:   float temp = 50.0;  
4:   public void flow() {  
5:     for (int i = 0; i < 1; i++) {  
6:       int depth = 2;  
7:       depth++;  
8:       temp--;  
9:     }  
10:    System.out.println(depth);  
11:    System.out.println(temp); }  
12: public static void main(String... s) {  
13:   new River().flow();  
14: }}
```

- A. Line 3 generates a compiler error.
- B. Line 6 generates a compiler error.
- C. Line 7 generates a compiler error.
- D. Line 10 generates a compiler error.
- E. The program prints 3 on line 10.
- F. The program prints 4 on line 10.
- G. The program prints 50.0 on line 11.
- H. The program prints 49.0 on line 11.

A, D

El primer error de compilación está en la línea 3. La variable temp se declara como un float, pero el valor asignado es 50.0, que es un double sin el sufijo F/f. Dado que un double no cabe dentro de un float, la línea 3 no compila. A continuación, depth se declara dentro del bucle for y solo tiene alcance dentro de este bucle. Por lo tanto, leer el valor en la línea 10 desencadena un error de compilación. Por estas razones, las opciones A y D son las respuestas correctas.