

Nombre:

1) Enumere todos los tipos de datos primitivos.

byte, short, int, ~~long~~ long
float, double
char
boolean

2) Todos los tipos de datos primitivos se representan internamente siempre del mismo modo independientemente de la plataforma que usemos (tipo de hardware+Sistema operativo).

- a) sí b) no c) todos menos uno → **boolean**

3) Que un método tenga tipo de retorno `void` quiere decir que siempre devuelve `null`.

- a) Cierto b) Falso → **no devuelve nada. No es asignable**

4) ¿Cuáles de las siguiente expresiones dan 0.5?

- a) 1/2 b) 1.0/2 c) 1/2.0 d) (double)1/2 e) (double) (1/2) f) ninguna

5) Dos métodos de una misma clase pueden tener el mismo nombre y parámetros idénticos (la misma cantidad y con los mismos tipos en las mismas posiciones)

- a) falso b) sí, si el tipo de retorno es diferente

6) Supongamos que `isPrime` es una variable booleana. ¿Qué expresiones serían válidas para ejecutar algo si tiene el valor <<cierto>> (`true`)

- a) **if (isPrime = true)...** b) **if (isPrime != false)** c) **if (isPrime)**

7) Todas las clases tienen un método constructor por defecto.

- a) sí b) no

8) Una interfaz...

- a) ... puede extender a una clase abstracta .
b) ...es una clase abstracta.
c) ... es una colección de métodos abstractos. d) Todas las afirmaciones son correctas.

9) ¿Cómo deberíamos proceder con los campos de una una clase en desarrollo?

- a) Haciendo todo público y cambiando a privado lo que se pueda cuando el desarrollo termine.
b) Manteniendo siempre todo privado y generando todos los getters y setters.
c) Primando el acceso privado pero considerando la necesidad de hacerlo público o usar getter y/o setter.
d) Manteniendo siempre todo privado y generando los getters y setters cuando se necesiten.

10) ¿Cuál de las siguientes expresiones nos da acceso al anteúltimo elemento de una matriz? (si existe)

- a) **a[a.length-1]** b) **a[a.length-2]** c) **a[a.length()-1]** d) **a[a.length()-2]**

11) ¿Qué expresiones son equivalentes en cuanto al resultado entre las siguientes?

- [1] 3/2 [2] 3<<2 [3] 3*4 [4] 3<2
a) 1 y 2 b) 2 y 3 c) 3 y 4 d) 1 y 4 e) no hay dos equivalentes

- 12) Una clase que extiende a otra abstracta puede ser abstracta. a) Cierto b) Falso
Una clase que extiende a otra no abstracta no puede ser abstracta. a) Cierto b) Falso
Una clase Java puede implementar más de un interfaz. a) Cierto b) Falso
Si una clase implementa un interfaz, debe definir todos sus métodos. a) Cierto b) Falso

13) ¿Qué resultado da este código?

```
int x = 10;
while (x > 0) x-- ;
switch (x) {
    case 0: System.out.print("cual");
    case -1: System.out.print(" es ");
    default: System.out.print("la verdad");
}
```

- a) cual b) cual es la verdad c) la verdad d) es la verdad e) la verdad es cual f) cual es g) es

14) Termine las siguientes frases

Si una clase es final entonces... *otras clases no pueden extenderla*
 Si un campo es final entonces... *cuando tome un valor será definitivo*
 Si un método es final entonces... *no puede reescribirse en subclases*

15) De acuerdo con el siguiente código: `public class A {public A() {}}
class B extends A {}`

- a) B tiene un constructor público b) B no es correcta c) B no tiene método constructor

16) Declare un array de primitivos enteros con los valores 5, 7, 12, 34, 53

int[] miArray = {5, 7, 12, 34, 53}

17) El siguiente código

```
public class A { public A(String s){...} }
class B extends A {}
```

- a) Da un error de compilación en la clase A b) Da un error de compilación en la clase B c) Va como la seda
 d) Da error de ejecución cuando se pretende generar un objeto de clase A
 e) Da error de ejecución cuando se pretende generar un objeto de clase B → *llama a super() y no existe*

18) ¿Puedo definir una clase arrojable (descendiente de **Throwable**) para que se encargue de atender a los eventos de un interfaz gráfico hecho con Swing?

- a) no, los objetos Throwables no pertenecen a Swing.
 b) no, debe ser un Listener en lugar de un Throwable.
 c) sí, si se cumplen los requisitos necesarios, pero no debería hacerse. → *el requisito es que sea "Listener"*

19) Para la variable `double[] miArray`, ¿cuáles de estos ciclos son equivalentes?

```
[1] for (double c:miArray) c=0.0
[2] for (int i=0;i<miArray.length; i++) miArray [i]=0.0;
[3] for (double c:miArray) System.out.println(c);
[4] for (int i=0;i<miArray.length; i++) System.out.println(miArray [i]);
```

- a) 1 y 2 b) 3 y 4 c) a y b son ciertas

20) Los objetos arrojables se atienden con cláusulas catch o declarando que se vuelven a arrojar fuera del método en que nos encontremos, y ...

- a) deben atenderse todos
 b) podemos decidir siempre si los atenderemos o no.
 c) a y b son ciertas
 d) a y b son falsas → *podemos decidir algunos.*

Con otros estamos obligados por el compilador

1) La siguiente sentencia es correcta:

`String char = "Int";`

- a) Totalmente b) No c) Sí, pero es claramente mejorable.

char es palabra reservada

2) Dos métodos de una clase pueden tener el mismo nombre y el mismo número de argumentos:

- a) Si b) No c) sólo si tienen distinto tipo de retorno.

Si los tipos/clases de los argumentos no coinciden.

3) ¿Qué resultado da el siguiente código?

```
int[] v = {8,3,2,6};
int x = v[v[2]];
System.out.println(x);
```

- a) 8 b) 3 c) 2 d) 6 d) El código no es

correcto.

4) ¿Para intercambiar los valores de a y b puede utilizarse el siguiente código?:

```
b=a>b?a(b>a?b:a);
```

- a) Si b) No c) Sí para valores primitivos, pero no para objetos .

5) Cual es la expresión correcta para comprobar si una variable x está en el intervalo [3,10]:

- a) $3 \leq x \leq 10$ b) $(3 \leq x) \ \&\& \ (x \leq 10)$ c) Las dos son correctas d) las dos son incorrectas

6) ¿Cuál es el resultado de la siguiente expresión?: "1 + 2 + 3 + "Java"

- a) 123Java b) 6Java c) La expresión no es correcta.

7) ¿Qué resultado da el siguiente código?

```
boolean x = false;
if (x = true) System.out.print("Not ");
if (x) System.out.print("Done ");
System.out.println("End");
```

- a) Done End b) End c) Not Done End d) ninguna de las anteriores.

8) ¿Qué resultado da el siguiente código?

```
int a = 123;
for(int x=a; x>=0; x++) x--;
System.out.println("x: " + x);
```

- a) x: -2 b) x: -1 c) x: -0 d) ninguna de las anteriores.

no existe fuera del for.

9) ¿Qué resultado da el siguiente código?

```
int x = 1357;
int z = 0;
for(int i=x; i>0; i/=10) z = z + (i%10);
System.out.println(z);
```

- a) 7531 b) 1357 c) 156 d) 16 e) ninguna de las anteriores.

10) desde un método estático...

- a) no se puede ejecutar otro método no estático de la clase.
 b) si se crea un objeto de la misma clase y se pretende acceder a uno de sus campos debe ser estático .
 c) si se crea un objeto de la misma clase y se pretende acceder a uno de sus métodos debe ser estático .
 d) Las dos anteriores opciones son ciertas.

11) ¿Qué puede ir en el espacio subrayado?

```
public class Test {
    static int id;
    public void m1() { ____ .id = 45; }
}
```

- a) this
- b) Test
- c) Las dos cosas.

12) ¿Qué resultado da el siguiente código?

```
public void myMethod(int[] v) {
    int x=v[0];
    for (int i=1; i<=v.length; i++) x = (v[i]>x) ? v[i] : x;
    System.out.println( x);
}
```

sale de rango

- a) El valor mínimo del vector v
- b) El valor máximo del vector v
- c) La suma de los valores en v
- d) Todas las opciones anteriores son incorrectas

13) El siguiente código...

```
class Piece {
    int x;
    // Constructor
    public thePiece() {
        x = 1;
    }
}
```

*error - no es método porque no tiene retorno
- no es constructor porque no es "Piece"*

- a) Se compila y funciona correctamente.
- b) Se compila pero no funciona como se espera.
- c) Genera un error de ejecución.
- d) Ninguna de las opciones anteriores es correcta

14) El siguiente código...

```
class A {} //1
class B extends A {} //2
class C extends A {} //3
A a = new B(); //4
C c = (C) a; //5
```

- a) Dará error de compilación en la línea 4
- b) Dará error de compilación en la línea 5
- c) Dará error de ejecución en la línea 4
- d) Dará error de ejecución en la línea 5

15) El siguiente código...

```
Class MyText {
    String txt = "Hello";
    public static void main(String[] args) {
        MyText o = new MyText();
        System.out.println(MyText.txt);
    }
}
```

no es estática

- a) Producirá un error de ejecución por falta de constructor.
- b) Al ejecutar aparecerá en la salida estándar "Hello"
- c) Da error de compilación.
- d) Todas las opciones anteriores son incorrectas.

1) Un carácter en Java se almacena en...

- a) 1 byte b) 2 bytes c) 3 bytes d) 4 bytes

2) `System.out.println((char)4)` ¿mostrará en pantalla 4 ?

- a) Sí b) No

3) La expresión "Java "+1 +2+ 3 da como resultado:

- a) Java123 b) Java6 c) Java 123 d) java 123 e) Da un error de compilación: Illegal Expression

4) Analiza el siguiente código:

```
boolean par = false;
if (par = true) System.out.println("¡Es par!");
```

- a) El programa tiene un error de compilación. b) El programa tiene un error de ejecución.
c) El programa funciona pero no muestra nada. d) El programa funciona y muestra ¡Es par!.

5) Suponga que tiene que escribir un programa en el que se determina si un usuario puede o no sacar el carnet de conducir. Indique cuales de las siguientes versiones realizan dicha acción.

- a) `if (edad < 16) System.out.println("No puede sacar el carnet de conducir");`
`if (edad >= 16) System.out.println("Puede sacar el carnet de conducir");`
b) `if (edad < 16) System.out.println("No puede sacar el carnet de conducir ");`
`else System.out.println("Puede sacar el carnet de conducir ");`
c) `if (edad < 16) System.out.println("No puede sacar el carnet de conducir ");`
`else if (edad >= 16) System.out.println("Puede sacar el carnet de conducir ");`
d) `if (edad < 16) System.out.println("No puede sacar el carnet de conducir ");`
`else if (edad > 16) System.out.println("Puede sacar el carnet de conducir ");`
`else if (edad == 16) System.out.println("Puede sacar el carnet de conducir ");`

6) Para el caso anterior ¿cuál es la expresión más correcta?

- a) b) c) d)

7) Para el mismo caso anterior ¿Podría hacerlo aún mejor?

(edad < 16 ? "No p" : "p") + "uede sacar el carnet de conducir";

8) Suponiendo que $x=10$ e $y=9$, si ejecutamos $z=y<--x$ ¿Qué es cierto de lo siguiente?

- a) z es 10, y es 10, x es 9 b) z es true c) z es false

9) ¿Cuál de estas expresiones dan un resultado de 0.5 ?

- a) `1 / 2` b) `1.0 / 2` c) `(double) (1 / 2)` d) `(double) 1 / 2` e) `1 / 2.0`

10) Analiza el siguiente código

```
public class Test {
    public static void main(String[] args) {
        double radio;
        final double PI= 3.1415926;
        double area = radio * radio * PI;
        System.out.println("El área es " + area);
    }
}
```

- a) Hay errores de compilación porque la variable radio no está inicializada
b) Hay errores de compilación porque la constante PI está definida dentro de un método
c) No hay errores de compilación pero se producirá un error de ejecución porque la variable radio no está inicializada.
d) El programa se compilará y ejecutará correctamente.

11) ¿Cuáles de las siguientes asignaciones son ilegales ?

- a) `float f = -34;` b) `int t = 23;` c) `short s = 09;`

no valido en octal
↑
octal

cast ilegal
d) int t = (int>false;

necesita cast (valdría 4)
e) int t = 4.5;

12) ¿ Cuales de estas expresiones dan 45.37 ?

a) (int) (45.378 * 100) / 100

b) (int) (45.378 * 100) / 100.0

c) (int) (45.378 * 100 / 100)

d) (int) (45.378) * 100 / 100.0

13) Es correcto poner la palabra reservada `static` como se ha puesto?

```
1 public class Test {
2     private int age;
3     public static int square(int n) {return n * n;}
4     public static int getAge() {}
7 }
```

** Ejercicio anulado
 * falta cuerpo de getAge()
 * sería { return age; }
 * Con eso respuesta a)*

a) en la línea 3 sí

b) en la línea 4 sí

c) en ambas

d) en ninguna

14) Crea un `array` que contenga los siguientes elementos:

- En la 1ª posición: un `Integer` con valor 10
- En la 2ª posición: un `Double` con valor 5.7
- En la 3ª posición: un `Boolean` con valor true
- En la 4ª posición: una `String` con valor "Hello"

Object[] miArray = { 10, 5.7, true, "Hello" };

15) Cuando se invoca un método con un objeto como argumento, se pasa...

a) el contenido del objeto

b) una copia del objeto

c) la referencia del objeto

d) la referencia de una copia del objeto

16) ¿Cual es el valor mostrado para `myCount.count`?

```
public class Test {
    public static void main(String[] args) {
        Count myCount = new Count();
        int times = 0;

        for (int i=0; i<100; i++) increment(myCount, times);

        System.out.println("myCount.count = " + myCount.count);
        System.out.println("times = " + times);
    }

    public static void increment(Count c, int times) {
        c.count++;
        times++;
    }
}

class Count {
    int count;
    Count(int c) {count = c;}
    Count() {count = 1; }
}
```

a) 101

b) 100

c) 99

d) 98

17) Cuales de las siguientes afirmaciones son ciertas?

- a) Se usa la palabra reservada `throws` para declarar excepciones en la cabecera del método**
- b) Un método puede declarar que arroja múltiples excepciones**
- c) Para arrojar una excepción, se usa la palabra reservada `throw`**
- d) Una excepción "atendible" debe ser considerada en un `catch` o declarada como arrojable por el método.**

1) El orden de palabras reservadas que tiene Java en su versión actual es de: 10 40 80 (son 50)

2) Enumere los tipos primitivos de datos *byte short int long float double char boolean*

3) `float f;`
`double g=3.14159;`

Escriba una instrucción para asignar el valor de `g` a `f`.

f = (float) g;

4) Indique cuales de estos identificadores son correctos

- a) pepe b) pepe22 c) 22pepe
d) pepe_ e) pepe_22 f) _22pepe
g) esteIdentificadorEsDemasiadoLargoPorTenerMasDe32Caracteres
h) seriaCorrectoSiNoIncluyeraAcentos i) o_No?

5) Para cada expresión, si es correcta indique el valor final de `x`, `y` y `z`, y si no lo es indique el motivo

x=1, y=2, z=3
a) `int x=1, y=1;`
`int z=x + ++y;`

b) `int x=1, y=1;`
`int z ++x + y++`

c) `int z = x + y++;`
`int x=1, y=1;`

z no tiene valor previo

6) El compilador dice "Incompatible types. Required: boolean. Found: int" corrija el error y diga qué efecto tiene la sentencia. Simplifíquela si es posible.

`if (x == ++y) x = y;`

if (x == ++y) x = y

Simplificación: ++y

7) Reescriba el siguiente código con un "switch"

`if (x+y == 2) z=4;`
`else if (x+y == 3) z=9;`
`else if (x+y == -2) z=4;`
`else if (x+y == -3) z=9;`
`else z=0;`

*switch(x+y){
 case 2:
 z=4; break;
 case 3:
 z=9; break;
 default: z=0; }
}*

*z=x+y;
if(z==2 || z==3) z*=z;
else z=0;*

8) ¿Le parece conveniente el código del ejercicio anterior o lo escribiría usted de otro modo (hágalo en tal caso)?

9) Haga de JVM y ejecute este fragmento de código.

(0,0)(0,1)(0,2)(0,3)(1,0)(1,1)(1,2)(1,3)(2,0)(2,1)(2,2)(3,0)(3,1)
`nivelX:`
`for (int i=0; i<4; i++)`
`for (int j=0; j<4; j++) {`
`if (i*j==6) continue nivelX;`
`System.out.print("(" + i + "." + j + ") ");`
`}`

10) Trate de simplificar (compactar al máximo) el siguiente código:

`int x=0;`
`int f(int a, int b) {`
`if (a>b) x = x + a / b;`
`else if (a == b) x = x + a * b;`
`else x = x + b / a;`
`return x;`
`}`

*int x=0
int f(int a, int b) {
 return x+=a>b?a/b:a==b?a*b:b/a;
}*

11) Dada la definición de clase *tiene un método "especial": el constructor vacío*
 class A {}

Comente la siguiente afirmación: "un objeto de tipo A es un objeto vacío: no tiene ni campos ni métodos".

12) Enumere los tipos de encapsulamientos (ámbitos de acceso) de modo ordenado.

private, package, protected, public (package no es palabra reservada

13) Haga de nuevo de JVM y ejecute este fragmento de código.

y se indica por distinción de las demás)

```
class Pruebas{
    private int i=0;
    private static int j=0;

    public static void main(String[] args) {
        Pruebas p1=new Pruebas(), p2=new Pruebas();
        p1.i++; p1.j++; p2.i--; p2.j--;
        System.out.println(p1.i+" "+p1.j+" "+p2.i+" "+p2.j);
    }
}
```

1 0 -1 0

14) Complete las frases:

- a) que una clase sea final implica que... *otras clases no pueden extenderla*
 b) que un campo sea final implica que... *cualdo tome un valor será definitivo*
 c) que un método sea final implica que... *no puede reescribirse en subclases*

15) Verdadero o falso

- a) todas las clases tienen un constructor sin parámetros. *FALSO*
 b) en la escritura de un constructor se debe explicitar un tipo de retorno void *FALSO*
 c) puede haber más de un constructor y no hay límites al número de ellos. *CIERTO*

16) Verdadero o falso

- a) Una clase que extiende otra clase abstracta puede ser abstracta. *CIERTO*
 b) Una clase que extiende otra clase no abstracta no puede ser abstracta *FALSO*

17) Verdadero o falso

- a) Hay lenguajes que admiten interfaces múltiples, pero no es el caso de Java *FALSO*
 b) Una clase que implementa un interfaz debe tener implementados todos los métodos de la misma. *FALSO*

18) Verdadero o falso

- a) Referenciar a un objeto a través de una clase/interfaz u otra no cambia la naturaleza del objeto. *CIERTO*
 b) Referenciar a un objeto a través de una clase/interfaz u otra nos permite ver a los objetos como algo que no son. *FALSO*

- 19) a) Declare e instancie un array de dimensión 3x2 para objetos de la clase Pájaro. *Pájaro [3][2] p=new Pájaro [3][2]*
 b) Declare e instancie un array lineal de enteros con los valores 1,3,5,7,11 *int[] i={1,3,5,7,11}*

20) Escriba la enumeración PiezaDeAjedrez

enum PiezaDeAjedrez { PEON, TORRE, CABALLO, ALFIL, REY, REINA; }