

Test Autoevaluación. UF2. Construcción de software

- Fecha de entrega No hay fecha de entrega
- Puntos 10
- Preguntas 10
- Disponible 9 de sep a las 0:00 - 29 de jun de 2025 a las 23:59
- Límite de tiempo Ninguno
- Intentos permitidos Ilimitado

Instrucciones



Este test incluye un total de **21 preguntas** sobre los diferentes contenidos aprendidos en esta unidad.

¡Adelante!

[Volver a hacer la prueba](#)

Historial de intentos

	Intento	Hora	Puntuación
MANTENER	Intento 2	4 minutos	8,5 de 10
MÁS RECIENTE	Intento 2	4 minutos	8,5 de 10
	Intento 1	2 minutos	7 de 10

Puntuación para este intento: 8,5 de 10

Entregado 19 de nov a las 12:14

Este intento ha tenido una duración de 4 minutos.



Pregunta 1

0,5 / 1 puntos

Cuando creamos un script para una tarea concreta:

¡Correcto!

☒ Siempre debemos usar buenas prácticas de codificación y escribir tests.

No se debe hacer siempre, lo fundamental es que hagan bien su trabajo.

☐ Siempre lo podremos ejecutar en cualquier sistema.

Respuesta correcta

- ☐ Lo almacenaremos en un repositorio distinto al código al que da soporte
- ☐ Siempre se escribe usando Bash.



Pregunta 2

1 / 1 puntos

¿Consideras completo el test de la función X?

```
int totalValidos(List items = null) {  
    int l = 0;  
    for (int i=0; i<items.size() ; i++) {  
        if (items[i] == 'ok') {  
            l+=1;  
        }  
    }  
    return l;  
}  
  
void test_totalValidos() {  
    assert totalValidos([]) == 0;  
    assert totalValidos(['ok']) == 1;  
    assert totalValidos([1]) == 0;  
}
```

- ☐ Si, se prueban todos los casos.

¡Correcto!

- ☒ No, porque no prueba la entrada nula.

Exacto: Si el parámetro es nulo o no se especifica, el código fallará y no existe ningún test que lo verifique.

- ☐ No, porque no prueba la entrada vacía.
- ☐ No, porque no prueba el caso en que la lista no contenga cadenas.



Pregunta 3

0,5 / 1 puntos

¿Cuál de los siguientes lenguajes es un lenguaje *compilado*?

Respuesta correcta

- ☐ Java 1.0.

¡Correcto!

- ☒ Java 1.8.

Correcto: Java es un lenguaje que necesita compilar el código fuente a bytecodes para su ejecución en una JVM, desde la primera versión hasta la última.

- ☐ Python.
- ☐ JavaScript.



Pregunta 4

1 / 1 puntos

¿A qué nos referimos con 'ciclos rápidos de desarrollo'?

☐ Los ciclos rápidos de desarrollo garantizan que cada semana se tendrá un entregable listo para desplegar.

¡Correcto!

☒ Los ciclos rápidos de desarrollo garantizan que se tendrá feedback rápidamente de posibles errores.

¡Exacto! Esto permite arreglarlos y seguir iterando, hasta llegar a resolver el problema que nos habíamos propuesto.

☐ Los ciclos rápidos de desarrollo se basan en generar tests antes que el código que los hacer funcionar.

☐ Los ciclos rápidos de desarrollo se basan en entregar funcionalidad nueva lo más rápido posible.



Pregunta 5

0,5 / 1 puntos

¿Cuál de los siguientes es un lenguaje *interpretado*?

☐ Java.

Respuesta correcta

☐ JavaScript

¡Correcto!

☒ Python.

¡Correcto!

☐ Ninguno de los anteriores.



Pregunta 6

1 / 1 puntos

¿Cuál de los siguientes es un lenguaje *tipado*?

¡Correcto!

☒ Java.

¡Correcto! Java es un lenguaje tipado.

☐ JavaScript.

☐ Python.

☐ Ninguno de los anteriores.



Pregunta 7

1 / 1 puntos

¿Qué hace el siguiente bloque de código funcional en Java 8?

```
new Random().ints().limit(100).filter( i -> i < 30 ).map( i -> i^2 ).sum();
```

☐ Calcula la suma de los cuadrados de 100 enteros aleatorios.

☐ Filtra 100 valores aleatorios y calcula la suma de sus cuadrados.

☐ Calcula la suma de los cuadrados de 100 enteros aleatorios menores de 30.

¡Correcto!

- ☒ Filtra 100 números enteros aleatorios y calcula la suma de los cuadrados de los menores de 30.

¡Exacto!



Pregunta 8

1 / 1 puntos

Un test que comprueba que mi aplicación se conecta correctamente a una base de datos MySQL es:

- ☐ Un test unitario.
- ☐ Un test end-to-end.

¡Correcto!

- ☒ Un test de integración.

¡Exacto! Podría ser un test funcional también, pero es el tipo de integración que interesa poder probar con Mock Servers o incluso con contenedores.

- ☐ Un smoke test.



Pregunta 9

1 / 1 puntos

Considera el siguiente algoritmo, que elimina duplicados de una lista de la siguiente forma:

```
List eliminar Duplicados(List A) {  
    B = List();  
    for (item in A) {  
        if (item in B) {  
            continue;  
        }  
        B.append(item);  
    }  
    return B  
}
```

- ☐ Su complejidad temporal en el mejor caso es $O(n)$.
- ☐ Su complejidad temporal en el mejor caso es $O(n \cdot \log(n))$.

¡Correcto!

- ☒ Su complejidad temporal en el peor caso es $O(n^2)$.

Correcto: en este algoritmo no hay mejor o peor caso, la complejidad es siempre la misma.

- ☐ Su complejidad temporal en el peor caso es $O(n \cdot \log(n))$.



Pregunta 10

1 / 1 puntos

Un lenguaje orientado a objetos

- ☐ Nunca puede pasar funciones como argumentos o valores de retorno.
- ☐ Nunca es de tipo procedimental.

- ☐ Es un tipo de lenguaje declarativo basado en el concepto de 'Objetos' y 'Clases'.

¡Correcto!

- ☒ Es un tipo de lenguaje imperativo basado en el concepto de 'Objetos' y 'Clases'.

¡Exacto! Es un paradigma dentro de los lenguajes imperativos en el que los Objetos son entidades que almacenan datos y estado.

Puntuación de la prueba: 8,5 de 10