



DATOS DEL ALUMNO			FIRMA
APELLIDOS:			
Nombre:	D.N.I./ N.I.E.	Fecha:	
Código del ciclo: (1) IFCS02	Denominación completa del título: (1) Técnico Superior en Desarrollo de Aplicaciones Multiplataforma		
Clave o código del módulo: (1) 0487	Denominación completa del módulo profesional: (1) Entornos de Desarrollo		

RA1: Reconoce los elementos y herramientas que intervienen en el desarrollo de un programa informático, analizando sus características y las fases en las que actúan hasta llegar a su puesta en funcionamiento.

1 EDRA1 Cuando un programa ya compilado se ejecuta en un ordenador, ¿qué papel desempeña el sistema operativo?

- a) Traduce el código fuente a lenguaje máquina durante la ejecución.
- b) Sustituye al procesador en la ejecución de instrucciones.
- c) Carga el programa, gestiona recursos y controla el acceso a los dispositivos.
- d) Permite que el programa use directamente toda la memoria y los periféricos sin intermediación.

2 ED RA1 En un modelo en cascada, el cliente pide un cambio importante cuando ya ha terminado la fase de diseño. ¿Qué consecuencia es la más propia de este modelo?

- a) El cambio se incorpora sin afectar a la planificación, porque el modelo prevé cambios continuos.
- b) Suele obligar a volver a fases ya cerradas, con más coste y retrasos.
- c) Solo afecta a la documentación final, no al desarrollo técnico.
- d) No afecta al proyecto si la implementación aún no ha terminado.

3 ED RA1 ¿Cuál de estas secuencias describe correctamente el proceso habitual en Java?

- a) Código fuente .java → bytecode .class → ejecución en la JVM.
- b) Código fuente .java → ejecutable nativo universal → JVM.
- c) Código fuente .java → código objeto del sistema operativo → procesador.
- d) Código fuente .java → enlazado con el kernel → ejecución directa)

4 ED RA1 ¿Cuál de estas afirmaciones es correcta sobre la clasificación de los lenguajes de programación?

- a) Todo lenguaje compilado es orientado a objetos.
- b) Todo lenguaje interpretado es estructurado.
- c) Un lenguaje orientado a objetos no puede incluir funciones.
- d) El paradigma y la forma de ejecución son clasificaciones distintas.



DATOS DEL ALUMNO			FIRMA
APELLIDOS:			
Nombre:	D.N.I./ N.I.E.	Fecha:	
Código del ciclo: (1) IFCS02	Denominación completa del título: (1) Técnico Superior en Desarrollo de Aplicaciones Multiplataforma		
Clave o código del módulo: (1) 0487	Denominación completa del módulo profesional: (1) Entornos de Desarrollo		

5 ED RA1 ¿Cuál es la ventaja principal de un IDE frente a un editor de texto simple?

- a) Genera software final sin necesidad de compilar ni depurar.
- b) Reúne edición, compilación, depuración y otras ayudas en una misma herramienta)
- c) Evita por sí solo los errores lógicos del programa)
- d) Convierte cualquier aplicación en multiplataforma automáticamente.

6 ED RA1 En Scrum, si hay que lanzar un MVP en pocos sprints, ¿qué actuación encaja mejor con el rol adecuado?

- a) El Scrum Master decide qué funcionalidades entran en el MVP y reparte tareas.
- b) El equipo de desarrollo bloquea el backlog hasta terminar todo lo previsto.
- c) El Product Owner prioriza el backlog para entregar cuanto antes el mayor valor posible.
- d) El cliente valida el producto solo al final del proyecto.

7 ED RA1 En compilación nativa, ¿qué afirmación es correcta?

- a) El código fuente se ejecuta sin traducción previa.
- b) El ejecutable resultante suele depender de la plataforma destino.
- c) Toda compilación genera bytecode portable.
- d) Compilar solo resuelve llamadas a bibliotecas.

8 ED RA1 En la metodología Scrum, cada sprint forma parte del proceso de convertir el product backlog en incrementos de producto dentro del roadmap general. ¿Cuál de las siguientes afirmaciones describe mejor la naturaleza de un sprint?

- a) Es una fase flexible sin límites de tiempo, que se adapta según la complejidad del backlog.
- b) Es un periodo de duración definida en el que el equipo desarrolla un incremento potencialmente entregable del producto.
- c) Es la etapa previa a la planificación del roadmap que permite definir los requisitos del backlog.
- d) Es la última iteración del proyecto en la que se entrega el producto final completo al cliente.

9 ED RA1 En el ciclo de vida del software, la fase de pruebas tiene un objetivo principal. ¿Cuál de las siguientes opciones lo expresa con mayor precisión?

- a) Comprobar que toda la documentación, el diseño y las funcionalidades se ajustan correctamente al plan definido.
- b) Probar que la planificación de la siguiente versión del software se mantiene coherente con los objetivos del proyecto y los requisitos del usuario.



DATOS DEL ALUMNO			FIRMA
APELLIDOS:			
Nombre:	D.N.I./ N.I.E.	Fecha:	
Código del ciclo: (1) IFCS02	Denominación completa del título: (1) Técnico Superior en Desarrollo de Aplicaciones Multiplataforma		
Clave o código del módulo: (1) 0487	Denominación completa del módulo profesional: (1) Entornos de Desarrollo		

- c) Verificar que el producto completo cumple los requisitos establecidos y funciona correctamente en los escenarios previstos antes de la entrega.
- d) Analizar aspectos internos del código y la interacción de los módulos, centrándose solo en detalles técnicos que no afectan a la validación frente a requisitos.



DATOS DEL ALUMNO			FIRMA
APELLIDOS:			
Nombre:	D.N.I./ N.I.E.	Fecha:	
Código del ciclo: (1) IFCS02	Denominación completa del título: (1) Técnico Superior en Desarrollo de Aplicaciones Multiplataforma		
Clave o código del módulo: (1) 0487	Denominación completa del módulo profesional: (1) Entornos de Desarrollo		

ED-RA2: Evalúa entornos integrados de desarrollo analizando sus características para editar código fuente y generar ejecutables.

10 ED RA2 Al instalar un entorno de desarrollo para trabajar con Java, sea libre o propietario, ¿qué requisito es realmente imprescindible para poder crear, compilar y ejecutar proyectos?

- a) Tener instalado únicamente el JRE, porque el IDE ya incorpora el compilador.
- b) Instalar antes un plugin de depuración, aunque todavía no exista ningún proyecto.
- c) Que el entorno reconozca un JDK válido y lo tenga configurado para el proyecto o el workspace.
- d) Definir manualmente el bytecode de destino en cada archivo fuente.

11 ED RA2 En un IDE extensible, ¿cuál de las siguientes afirmaciones describe correctamente la gestión de módulos y del sistema de actualización?

- a) Un módulo puede añadirse, pero no eliminarse, salvo reinstalando por completo el entorno.
- b) Los módulos se gestionan desde el sistema de extensiones del entorno, y las actualizaciones del IDE pueden configurarse mediante canales o repositorios.
- c) Al actualizar el IDE, el JDK del sistema se sustituye automáticamente por una versión compatible.
- d) El sistema operativo decide qué módulos del IDE quedan activos en cada proyecto.

12 ED RA2 ¿Cuál de estas acciones representa una personalización y automatización real del entorno orientada a mejorar la productividad?

- a) Desactivar el autocompletado y el formateo para mantener un control totalmente manual.
- b) Cambiar la carpeta de trabajo en cada proyecto para forzar una reindexación completa)
- c) Mantener siempre la configuración por defecto, aunque no se adapte al equipo ni al proyecto.
- d) Definir plantillas, autoformato, atajos y tareas automáticas de prueba o análisis al guardar.

13 ED RA2 Un mismo entorno de desarrollo puede trabajar con varios lenguajes y generar la salida ejecutable correspondiente a cada uno cuando:

- a) integra, para cada lenguaje, las herramientas necesarias de compilación o ejecución.
- b) todos los lenguajes usan exactamente el mismo compilador.
- c) los proyectos comparten la misma estructura de carpetas y nombres de archivos.
- d) el sistema operativo traduce automáticamente cualquier lenguaje al abrir el proyecto.



DATOS DEL ALUMNO			FIRMA
APELLIDOS:			
Nombre:	D.N.I./ N.I.E.	Fecha:	
Código del ciclo: (1) IFCS02	Denominación completa del título: (1) Técnico Superior en Desarrollo de Aplicaciones Multiplataforma		
Clave o código del módulo: (1) 0487	Denominación completa del módulo profesional: (1) Entornos de Desarrollo		

14 ED RA2 Un mismo proyecto Java basado en Maven se abre en dos IDE distintos. Ambos usan la misma versión de JDK y la misma configuración de construcción. ¿Qué resultado es el esperable al compilar?

- a) Cada IDE generará necesariamente bytecode distinto, porque usa su propio editor y su propio índice interno.
- b) El proyecto solo podrá compilarse correctamente en el IDE con el que fue creado por primera vez.
- c) Los artefactos generados deberían ser equivalentes, porque la construcción depende de la configuración de build y del JDK, no del IDE en sí.
- d) El IDE determina la semántica final del programa más que el compilador y el sistema de construcción.

15 ED RA2 ¿Cuál de estas opciones distingue mejor entre características comunes y características específicas de distintos entornos de desarrollo?

- a) Si dos entornos soportan el mismo lenguaje, deben ofrecer exactamente las mismas funciones.
- b) Edición, navegación, ejecución y depuración son rasgos comunes; el ecosistema de plugins, ciertos asistentes o herramientas integradas pueden ser específicos.
- c) La diferencia esencial entre entornos está solo en el tema visual y en el color del resaltado de sintaxis.
- d) Las herramientas de control de versiones no forman parte de las funciones habituales de un entorno moderno.

16 ED RA2 Respecto a los entornos de desarrollo libres y propietarios, ¿qué afirmación es correcta?

- a) Un IDE propietario no admite extensiones ni módulos adicionales.
- b) Un IDE libre funciona con cualquier lenguaje sin configuración previa.
- c) La licencia los diferencia, pero ambos pueden ofrecer edición, compilación y depuración.
- d) Un IDE libre no puede usarse para generar ejecutables en un entorno profesional.



DATOS DEL ALUMNO			FIRMA
APELLIDOS:			
Nombre:	D.N.I./ N.I.E.	Fecha:	
Código del ciclo: (1) IFCS02	Denominación completa del título: (1) Técnico Superior en Desarrollo de Aplicaciones Multiplataforma		
Clave o código del módulo: (1) 0487	Denominación completa del módulo profesional: (1) Entornos de Desarrollo		

ED-RA3: Verifica el funcionamiento de programas diseñando y realizando pruebas.

17 ED RA3 En un proyecto se realizan estas tres comprobaciones:

1. Se prueba una clase aislada sustituyendo sus dependencias por dobles de prueba)
2. Se verifica después que dos módulos reales intercambian correctamente la información.
3. Finalmente, el usuario comprueba que el flujo completo satisface lo que se pidió.

¿Qué secuencia clasifica correctamente esas comprobaciones?

- a) Integración, sistema, aceptación.
- b) Unitarias, integración, aceptación.
- c) Caja blanca, regresión, validación.
- d) Unitarias, sistema, regresión.

18 ED RA3 Un campo entero solo admite valores del 1 al 12, ambos incluidos.

Si se quiere diseñar un conjunto de pruebas con la técnica de clases de equivalencia, ¿qué conjunto es el más adecuado?

- a) 0, 1, 12, 13
- b) 1, 2, 11, 12
- c) 0, 6, 13
- d) 2, 6, 11

19 ED RA3 Observa esta clase de prueba en JUnit:

```
class CuentaTest {
    Cuenta c;
    @BeforeEach
    void preparar() {
        c = new Cuenta(100);
    }
    @Test
    void retirar20() {
        c.retirar(20);
        assertEquals(80, c.getSaldo());
    }
    @Test
    void saldoInicial() {
        assertEquals(100, c.getSaldo());
    }
}
```



DATOS DEL ALUMNO			FIRMA
APELLIDOS:			
Nombre:	D.N.I./ N.I.E.	Fecha:	
Código del ciclo: (1) IFCS02	Denominación completa del título: (1) Técnico Superior en Desarrollo de Aplicaciones Multiplataforma		
Clave o código del módulo: (1) 0487	Denominación completa del módulo profesional: (1) Entornos de Desarrollo		

¿Cuál de las siguientes afirmaciones es correcta?

- a) saldoInicial() puede fallar si retirar20() se ejecuta antes, porque ambos tests comparten el mismo objeto c)
- b) @BeforeEach se ejecuta una sola vez al principio de la clase, así que c conserva los cambios entre pruebas.
- c) retirar20() verifica integración, mientras que saldoInicial() verifica aceptación.
- d) Cada test parte de una nueva Cuenta(100), por lo que el cambio de una prueba no afecta a la otra)

20 ED RA3 Durante una depuración en Eclipse, la ejecución está detenida dentro del método validarPedido(). Ya has inspeccionado las líneas relevantes y ahora quieres terminar el resto del método actual y volver al punto desde el que fue llamado, quedándote allí de nuevo en pausa)

¿Qué acción es la más adecuada?

- a) Resume
- b) Step Into
- c) Step Over
- d) Step Return

21 ED RA3 En un test con Mockito, se ha creado un mock llamado repo que simula un repositorio de descuentos.

Después de ejecutar un método de la aplicación, aparece esta línea en el test:

```
verify(repo, times(1)).discountPercentFor("C1");
```

¿Qué se está comprobando con esa instrucción?

- a) Que el descuento aplicado al cliente "C1" ha sido del 1%.
- b) Que el método discountPercentFor("C1") del mock repo se ha llamado exactamente una vez.
- c) Que el mock repo devuelve siempre el mismo valor para cualquier cliente.
- d) Que durante la ejecución no se ha producido ninguna excepción.

22 ED RA3 A efectos de trazabilidad y cierre correcto de incidencias, ¿cuál de los siguientes registros es el más sólido?

a) Título: "Error al guardar pedido". Versión: última) Pasos: crear pedido y guardar. Esperado: que funcione. Obtenido: error. Evidencia: captura) Estado: cerrada)



DATOS DEL ALUMNO			FIRMA
APELLIDOS:			
Nombre:	D.N.I./ N.I.E.	Fecha:	
Código del ciclo: (1) IFCS02	Denominación completa del título: (1) Técnico Superior en Desarrollo de Aplicaciones Multiplataforma		
Clave o código del módulo: (1) 0487	Denominación completa del módulo profesional: (1) Entornos de Desarrollo		

b) Título: "NullPointerException al guardar pedido sin líneas". Versión: 1.4.2. Precondición: usuario autenticado. Pasos: crear pedido vacío y pulsar Guardar. Esperado: mensaje de validación. Obtenido: excepción con traza adjunta) Fix: commit referenciado. Estado: cerrada)

c) Título: "NullPointerException al guardar pedido sin líneas". Versión: 1.4.2. Precondición: usuario autenticado. Pasos: crear pedido vacío y pulsar Guardar. Esperado: mensaje de validación. Obtenido: excepción con traza adjunta) Evidencia: test automatizado fallando. Fix: commit referenciado. Validación tras el fix: mismo test en verde y batería relacionada ejecutada) Estado: cerrada)

d) Título: "Fallo de persistencia en pedidos". Versión: 1.4.2. Evidencia: captura y commit del arreglo. Validación: test en verde. Estado: cerrada)

23 ED RA3 En el ejemplo de cubrimiento con tres condiciones booleanas, ¿qué enfoque reduce el número de pruebas frente a probar las 8 combinaciones posibles, manteniendo una intención de cobertura razonable?

- a) Repetir varias veces un caso en que el resultado final sea verdadero.
- b) Diseñar casos que hagan tomar, al menos una vez, cada rama verdadera y falsa de las decisiones.
- c) Probar solo la combinación en la que las tres condiciones sean falsas.
- d) Evitar la cobertura de decisiones porque exige siempre todas las combinaciones.

24 ED RA3 Según la teoría del tema, ¿cuál es la afirmación más correcta sobre el papel de las pruebas en el desarrollo?

- a) Si todos los tests pasan, el programa ya está libre de errores.
- b) Las pruebas exhaustivas suelen ser inviables y conviene planificarlas.
- c) Las pruebas solo deben hacerse al final, cuando el programa ya esté completo.
- d) Las pruebas unitarias sustituyen completamente a las de integración.

25 ED RA3 En una integración ascendente, los módulos superiores no disponibles se sustituyen por:

- a) Stubs
- b) Drivers
- c) Mocks
- d) Fixtures



DATOS DEL ALUMNO			FIRMA
APELLIDOS:			
Nombre:	D.N.I./ N.I.E.	Fecha:	
Código del ciclo: (1) IFCS02	Denominación completa del título: (1) Técnico Superior en Desarrollo de Aplicaciones Multiplataforma		
Clave o código del módulo: (1) 0487	Denominación completa del módulo profesional: (1) Entornos de Desarrollo		

ED-RA4: Optimiza código empleando las herramientas disponibles en el entorno de desarrollo.

26 ED RA4 Analiza este código pensando en mantenibilidad y en los tipos de avisos que podría señalar SonarQube, ¿qué opción describe mejor el problema principal y la mejora más adecuada?

```
public double calculateFinalPrice(double basePrice, String country) {  
    if (country.equals("ES")) {  
        return basePrice * 1.21;  
    } else if (country.equals("US")) {  
        return basePrice * 1.3;  
    } else if (country.equals("FR")) {  
        return basePrice * 1.21;  
    } else {  
        return basePrice * 1.20;  
    }  
}
```

- a) El método tiene demasiados parámetros; la solución sería Introduce Parameter Object.
- b) Hay duplicación y valores fijos en el código.
- c) El método debería convertirse en static para que el análisis sea más simple.
- d) El principal problema es que faltan comentarios, así que lo mejor sería documentarlo mejor.

27 ED RA4 Este código funciona, pero quieres mejorarlo para que el diseño sea más claro y las responsabilidades queden mejor repartidas. ¿Qué refactorización resulta más adecuada?

```
class Customer {  
    private String name;  
    private String email;  
    private String phone;  
  
    public String getName() { return name; }  
    public String getEmail() { return email; }  
    public String getPhone() { return phone; }  
}  
  
class Order {  
    private Customer customer;  
    private String orderId;  
  
    public String getCustomerSummary() {  
        return customer.getName() + " - " +  
            customer.getEmail() + " - " +  
            customer.getPhone();  
    }  
}
```



DATOS DEL ALUMNO			FIRMA
APELLIDOS:			
Nombre:	D.N.I./ N.I.E.	Fecha:	
Código del ciclo: (1) IFCS02	Denominación completa del título: (1) Técnico Superior en Desarrollo de Aplicaciones Multiplataforma		
Clave o código del módulo: (1) 0487	Denominación completa del módulo profesional: (1) Entornos de Desarrollo		

- a) Extraer un método dentro de Order para concatenar las tres cadenas.
- b) Mover el método `getCustomerSummary()` a la clase Customer.
- c) Encapsular el campo `orderId` con getter y setter.
- d) Añadir comentarios explicando qué representa el resumen del cliente.

28 ED RA4 Observa el siguiente fragmento de documentación del método "dividir":

```
/**
 * Divide un número entre otro.
 *
 * <p>Utiliza división entera; los decimales se truncan.</p>
 *
 * @param a Dividendo
 * @param b Divisor
 * @return Cociente entero de dividir {@code a} entre {@code b}
 * @throws IllegalArgumentException si {@code b} es 0
 */
```

¿Cuál de las siguientes afirmaciones es correcta según un uso adecuado de JavaDoc como herramienta del entorno de desarrollo?

- a) El bloque JavaDoc permite que el compilador impida automáticamente en tiempo de compilación las llamadas en las que `b` vale 0.
- b) Al documentarse `IllegalArgumentException` con `@throws`, el método queda obligado a declarar esa excepción en su cabecera con `throws`.
- c) El comentario mejora la legibilidad del código, pero no puede ser utilizado por el IDE ni por herramientas automáticas para mostrar ayuda o generar documentación.
- d) La documentación añade información semántica que no se deduce por completo de la firma del método, como la truncación de decimales y la condición de error asociada al divisor.

29 ED R4 ¿Cuál de las siguientes afirmaciones es correcta según un uso adecuado de JavaDoc como herramienta del entorno de desarrollo?

- a) La etiqueta `@since` se utiliza para indicar que un método ha sido incorporado en una versión posterior del código respecto a la versión inicial de la clase.
- b) El uso de `@since` es necesario para que los métodos aparezcan correctamente en la documentación generada por JavaDoc)
- c) `@since` indica la versión del lenguaje Java utilizada al implementar el método dentro del proyecto.



DATOS DEL ALUMNO			FIRMA
APELLIDOS:			
Nombre:	D.N.I./ N.I.E.	Fecha:	
Código del ciclo: (1) IFCS02	Denominación completa del título: (1) Técnico Superior en Desarrollo de Aplicaciones Multiplataforma		
Clave o código del módulo: (1) 0487	Denominación completa del módulo profesional: (1) Entornos de Desarrollo		

d) La etiqueta @since sustituye a los comentarios del código cuando se documentan cambios o mejoras realizadas en un método.

30 ED RA4 Dos alumnos trabajan sobre el mismo repositorio remoto. Uno de ellos ya ha subido sus cambios y el otro, al hacer push, recibe un rechazo porque en el remoto hay cambios que él no tiene en local. ¿Qué debería hacer este segundo alumno?

- a) Ejecutar push --force para reemplazar el contenido del repositorio remoto.
- b) Eliminar su repositorio local y volver a clonarlo desde cero.
- c) Actualizar su repositorio local con git pull, resolver los conflictos si aparecen y volver a intentar el push.
- d) Esperar a que el otro alumno elimine su último commit.

31 ED RA4 ¿Qué diferencia básica existe entre SVN y Git?

- a) SVN sigue un modelo centralizado, mientras que Git utiliza un modelo distribuido en el que cada usuario dispone de un repositorio local completo.
- b) SVN permite crear ramas y Git no.
- c) Git solo sirve para proyectos pequeños y SVN para proyectos grandes.
- d) SVN no guarda historial de cambios y Git sí.

32 ED RA4 En un entorno de trabajo con SVN, ¿cuál de estas descripciones refleja mejor su funcionamiento habitual?

- a) El desarrollador realiza commits en un repositorio local y más tarde decide si los comparte.
- b) El desarrollador modifica los archivos en una copia de trabajo local y después envía los cambios al repositorio central.
- c) El desarrollador modifica directamente los archivos del servidor remoto sin copia local.
- d) Cada desarrollador dispone del historial completo del proyecto igual que en Git distribuido.

33 ED RA4 Una clase LibraryManager contiene estos métodos: addBook, removeBook, findBookByIsbn, calculateFine, hasPendingFines, saveBook y deleteBook. La aplicación sigue funcionando, pero el equipo quiere refactorizarla para que los cambios futuros sean más localizados y el diseño quede más claro. ¿Qué propuesta de refactorización resulta más adecuada?

- a) Mantener una sola clase, pero reorganizar internamente sus métodos y atributos para que queden agrupados por bloques de funcionalidad)
- b) Dividir la funcionalidad en varias clases con responsabilidades distintas, por ejemplo gestión de libros, cálculo de multas y persistencia)



DATOS DEL ALUMNO			FIRMA
APELLIDOS:			
Nombre:	D.N.I./ N.I.E.	Fecha:	
Código del ciclo: (1) IFCS02	Denominación completa del título: (1) Técnico Superior en Desarrollo de Aplicaciones Multiplataforma		
Clave o código del módulo: (1) 0487	Denominación completa del módulo profesional: (1) Entornos de Desarrollo		

- c) Convertir los métodos en static y centralizar la lógica en una única clase utilitaria, para que desde otras partes del programa se pueda acceder más fácilmente a todas las operaciones relacionadas con la biblioteca)
- d) Mantener la clase actual, pero mejorar la legibilidad con nombres más descriptivos, comentarios y una organización interna más clara del código.

34 ED RA4 Sobre la utilización de la herramienta JavaDoc integrada en el entorno de desarrollo Eclipse para generar la documentación del proyecto.

¿Cuál de las siguientes afirmaciones es correcta respecto al resultado de esa generación?

- a) La herramienta genera automáticamente un archivo en formato PDF que contiene toda la documentación del proyecto de forma estructurada)
- b) Al generar la documentación, se crea un conjunto de archivos HTML que permiten consultar la información utilizando un navegador web)
- c) La documentación generada solo puede visualizarse dentro del propio entorno de desarrollo y no es accesible desde otros programas.
- d) JavaDoc genera un único archivo de texto plano que recoge los comentarios del código sin estructura adicional.



DATOS DEL ALUMNO			FIRMA
APELLIDOS:			
Nombre:	D.N.I./ N.I.E.	Fecha:	
Código del ciclo: (1) IFCS02	Denominación completa del título: (1) Técnico Superior en Desarrollo de Aplicaciones Multiplataforma		
Clave o código del módulo: (1) 0487	Denominación completa del módulo profesional: (1) Entornos de Desarrollo		

ED-RA5: Genera diagramas de clases valorando su importancia en el desarrollo de aplicaciones y empleando herramientas específicas.

35 ED RA5 ¿Cuál de las siguientes opciones recoge exclusivamente los cuatro pilares de la programación orientada a objetos?

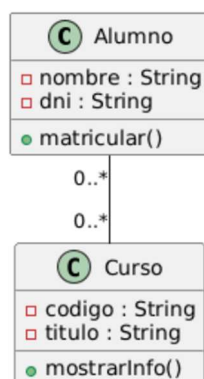
- a) Encapsulación, herencia, modularidad y reutilización.
- b) Clases, objetos, métodos y atributos.
- c) Abstracción, instanciación, herencia y sobrecarga)
- d) Abstracción, encapsulación, herencia y polimorfismo.

36 ED RA5 Una variable de un tipo base puede referenciar objetos de distintas clases derivadas y ejecutar el mismo método obteniendo comportamientos distintos.

¿Qué concepto de la programación orientada a objetos describe esta situación?

- a) Herencia, porque permite reutilizar código entre clases relacionadas.
- b) Abstracción, porque se define una interfaz común para varias clases.
- c) Polimorfismo, porque un mismo método se ejecuta de forma distinta según el objeto.
- d) Encapsulación, porque los detalles de implementación están ocultos.

37 ED RA5 ¿Cuál de las siguientes afirmaciones es correcta según la interpretación del diagrama?

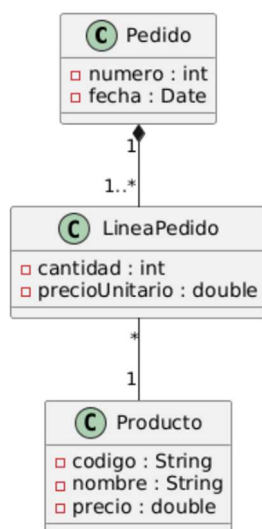




DATOS DEL ALUMNO			FIRMA
APELLIDOS:			
Nombre:	D.N.I./ N.I.E.	Fecha:	
Código del ciclo: (1) IFCS02	Denominación completa del título: (1) Técnico Superior en Desarrollo de Aplicaciones Multiplataforma		
Clave o código del módulo: (1) 0487	Denominación completa del módulo profesional: (1) Entornos de Desarrollo		

- a) Alumno y Curso son objetos concretos que solo existen cuando se ejecuta el programa)
- b) Alumno y Curso representan clases con atributos y métodos definidos en el diseño.
- c) La relación entre Alumno y Curso indica que uno hereda del otro.
- d) Los métodos del diagrama se ejecutan automáticamente al crear los objetos.

38 ED RA5 ¿Cuál de las siguientes afirmaciones interpreta correctamente las relaciones representadas en el diagrama?

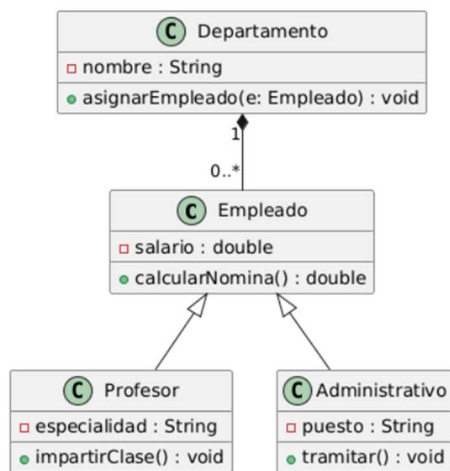


- a) La composición entre Pedido y LineaPedido indica que una línea de pedido no puede existir sin un pedido asociado, mientras que la asociación con Producto permite que un mismo producto sea referenciado por múltiples líneas de pedido sin depender de su ciclo de vida)
- b) El diagrama indica que Producto forma parte estructural de LineaPedido, por lo que al eliminar una línea de pedido también debe eliminarse el producto asociado para mantener la consistencia del modelo.
- c) La relación entre LineaPedido y Producto implica que un producto solo puede aparecer en una única línea de pedido, ya que la navegabilidad restringe la reutilización del producto en otros pedidos.
- d) La composición establece que Pedido hereda el comportamiento de LineaPedido, mientras que la asociación con Producto representa una especialización del producto dentro del contexto del pedido.



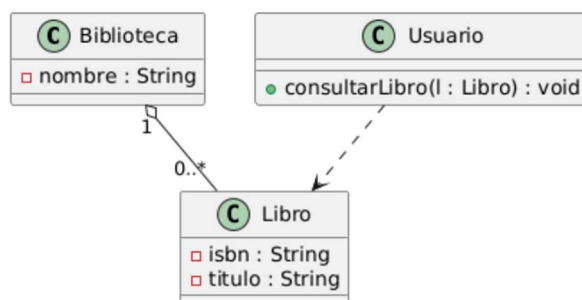
DATOS DEL ALUMNO			FIRMA
APELLIDOS:			
Nombre:	D.N.I./ N.I.E.	Fecha:	
Código del ciclo: (1) IFCS02	Denominación completa del título: (1) Técnico Superior en Desarrollo de Aplicaciones Multiplataforma		
Clave o código del módulo: (1) 0487	Denominación completa del módulo profesional: (1) Entornos de Desarrollo		

39 ED RA5 ¿Cuál de las siguientes afirmaciones describe correctamente el significado del diagrama?



- a) Profesor y Administrativo comparten estructura con Empleado, y un Departamento puede tener de 0 a muchos empleados.
- b) Un empleado pertenece a un Departamento y puede existir independientemente de este.
- c) Profesor y Administrativo son especializaciones de Empleado; además, Empleado no existe en el sistema si no existe el Departamento.
- d) El diagrama indica que Departamento hereda de Empleado, ya que ambos se relacionan con multiplicidad y operaciones.

40 ED RA5 ¿Cuál de las siguientes afirmaciones interpreta correctamente ambas relaciones representadas en el diagrama?





DATOS DEL ALUMNO			FIRMA
APELLIDOS:			
Nombre:	D.N.I./ N.I.E.	Fecha:	
Código del ciclo: (1) IFCS02	Denominación completa del título: (1) Técnico Superior en Desarrollo de Aplicaciones Multiplataforma		
Clave o código del módulo: (1) 0487	Denominación completa del módulo profesional: (1) Entornos de Desarrollo		

- a) La agregación indica una relación de herencia entre Biblioteca y Libro, mientras que la dependencia representa una asociación bidireccional entre Usuario y Libro.
- b) La relación entre Biblioteca y Libro es una composición, por lo que los libros no pueden existir sin la biblioteca, y la dependencia implica que Usuario hereda de Libro.
- c) El diagrama muestra que Libro controla el ciclo de vida de Biblioteca, y que la dependencia obliga a que Usuario almacene una colección de libros.
- d) La agregación indica que Biblioteca agrupa libros que pueden existir independientemente de ella, mientras que la dependencia refleja que Usuario utiliza Libro de forma puntual sin mantenerlo como atributo permanente.

41 ED RA5 Indica que es cierto acerca de los métodos:

- a) Se añaden como notas a una clase
- b) Son las acciones que podrá desempeñar una instancia de una clase
- c) No forman parte de los diagramas de clases
- d) Se muestran siempre con un valor entero

42 ED RA5 La diferencia entre composición y agregación es:

- a) La fuerza de la unión entre los elementos que lo componen
- b) El nivel de visibilidad entre los elementos que lo componen
- c) El tipo de valor de atributo de las clases
- d) El empleo de los diferentes niveles de visibilidad



DATOS DEL ALUMNO			FIRMA
APELLIDOS:			
Nombre:	D.N.I./ N.I.E.	Fecha:	
Código del ciclo: (1) IFCS02	Denominación completa del título: (1) Técnico Superior en Desarrollo de Aplicaciones Multiplataforma		
Clave o código del módulo: (1) 0487	Denominación completa del módulo profesional: (1) Entornos de Desarrollo		

ED-RA6: Genera diagramas de comportamiento valorando su importancia en el desarrollo de aplicaciones y empleando herramientas específicas.

43 ED R6 ¿Qué tipo de diagrama es el más adecuado para representar cómo cambia un objeto a lo largo del tiempo entre estados como Disponible, Prestado o Reservado?

- a) Diagrama de casos de uso
- b) Diagrama de secuencia
- c) Diagrama de estados
- d) Diagrama de actividades

44 ED R6 En UML, ¿qué diagrama de comportamiento resulta más adecuado cuando se quiere representar qué participantes intercambian mensajes entre sí, dando más importancia a sus relaciones que al paso del tiempo línea a línea?

- a) Diagrama de secuencia, porque muestra con claridad el orden temporal exacto de cada mensaje.
- b) Diagrama de actividades, porque representa el flujo general de tareas entre varios elementos.
- c) Diagrama de comunicación o colaboración, porque se centra en participantes y mensajes.
- d) Diagrama de estados, porque muestra cómo cambian los objetos al enviarse mensajes.

45 ED R6 En un parking inteligente, un conductor puede:

- consultar plazas libres,
- iniciar el estacionamiento,
- pagar al salir.

Además, un operario puede registrar una incidencia en el sistema)

¿Qué opción representa mejor el diagrama de casos de uso?

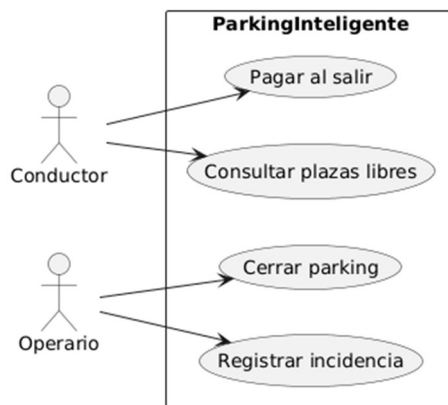


DATOS DEL ALUMNO			FIRMA
APELLIDOS:			
Nombre:	D.N.I./ N.I.E.	Fecha:	
Código del ciclo: (1) IFCS02	Denominación completa del título: (1) Técnico Superior en Desarrollo de Aplicaciones Multiplataforma		
Clave o código del módulo: (1) 0487	Denominación completa del módulo profesional: (1) Entornos de Desarrollo		



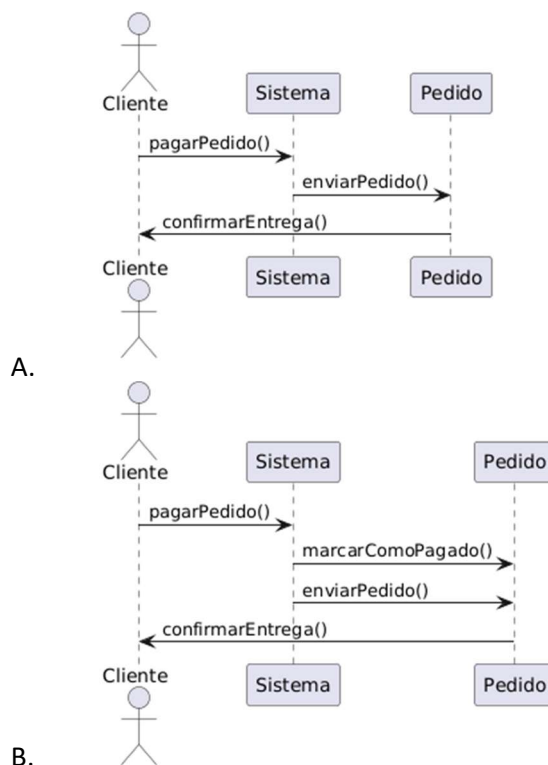


DATOS DEL ALUMNO			FIRMA
APELLIDOS:			
Nombre:	D.N.I./ N.I.E.	Fecha:	
Código del ciclo: (1) IFCS02	Denominación completa del título: (1) Técnico Superior en Desarrollo de Aplicaciones Multiplataforma		
Clave o código del módulo: (1) 0487	Denominación completa del módulo profesional: (1) Entornos de Desarrollo		



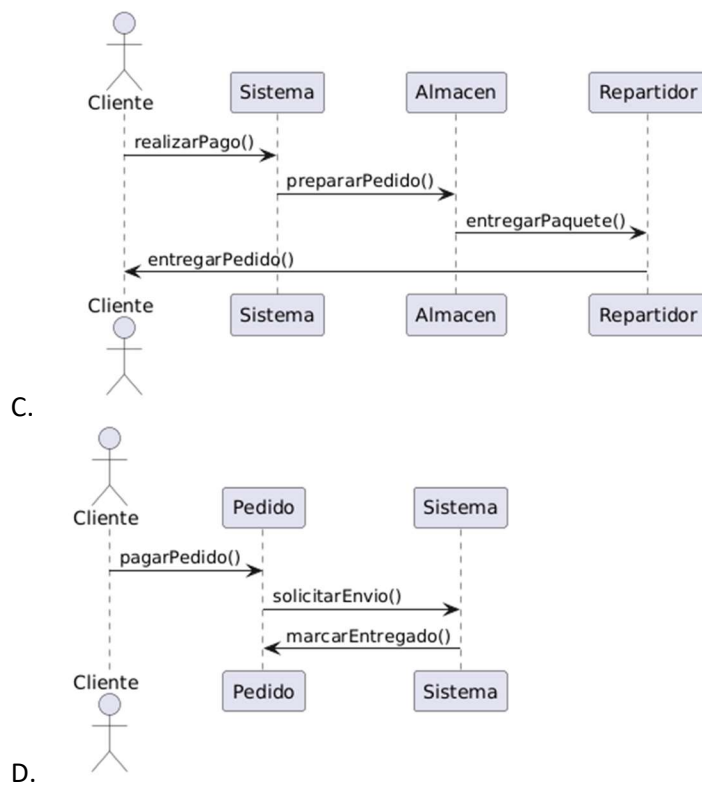
D.

46 ED R6 En una tienda online, un cliente realiza el pago, el sistema ordena preparar el pedido, el almacén lo entrega al repartidor y finalmente el pedido se entrega al cliente.
¿Qué opción representa mejor el diagrama de secuencia de este proceso?





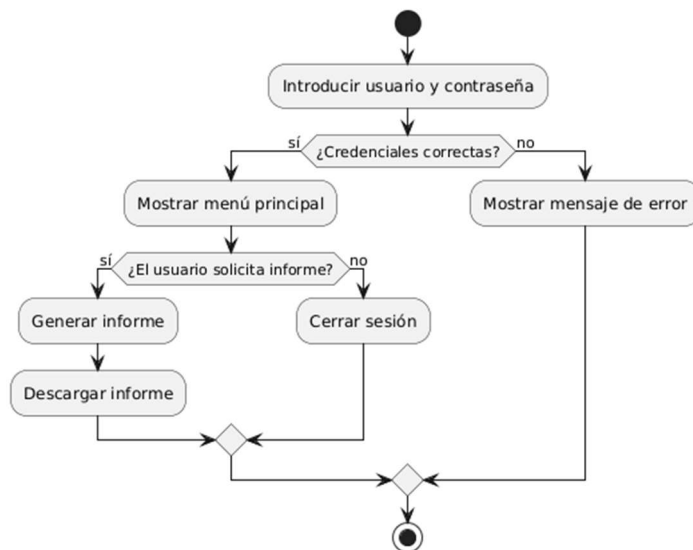
DATOS DEL ALUMNO			FIRMA
APELLIDOS:			
Nombre:	D.N.I./ N.I.E.	Fecha:	
Código del ciclo: (1) IFCS02	Denominación completa del título: (1) Técnico Superior en Desarrollo de Aplicaciones Multiplataforma		
Clave o código del módulo: (1) 0487	Denominación completa del módulo profesional: (1) Entornos de Desarrollo		





DATOS DEL ALUMNO			FIRMA
APELLIDOS:			
Nombre:	D.N.I./ N.I.E.	Fecha:	
Código del ciclo: (1) IFCS02	Denominación completa del título: (1) Técnico Superior en Desarrollo de Aplicaciones Multiplataforma		
Clave o código del módulo: (1) 0487	Denominación completa del módulo profesional: (1) Entornos de Desarrollo		

47 ED R6 Dado el siguiente diagrama:



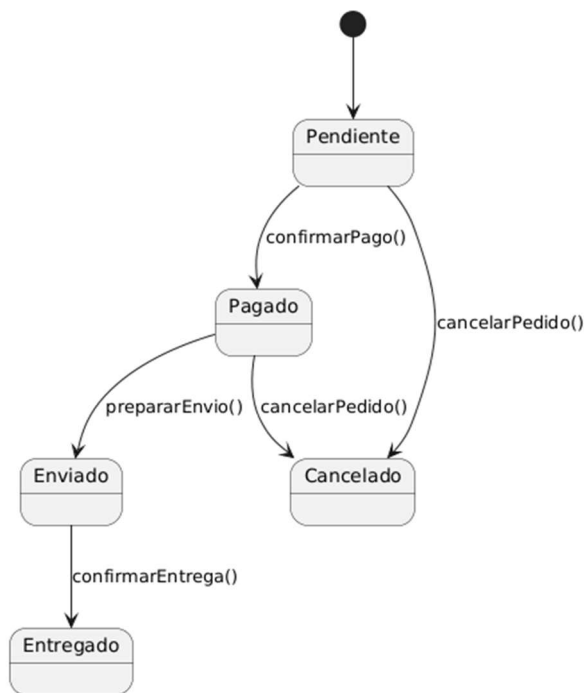
¿Qué enunciado describe correctamente el comportamiento representado en el diagrama?

- a) El usuario introduce sus credenciales y, si son correctas, accede al menú principal. Desde ahí el sistema genera siempre un informe, aunque después el usuario puede decidir si lo descarga o si cierra la sesión sin guardarlo. Si las credenciales no son válidas, el sistema vuelve al inicio.
- b) El usuario introduce sus credenciales y el sistema decide si mostrar el menú principal o cerrar la sesión. Solo después de cerrar sesión, y si el acceso era correcto, puede solicitarse la generación y descarga de un informe. Si las credenciales fallan, el proceso termina sin mostrar ningún mensaje.
- c) El usuario introduce sus credenciales. Si son correctas, accede al menú principal; desde ahí puede solicitar un informe, que se genera y descarga, o bien cerrar sesión sin generar nada) Si las credenciales son incorrectas, se muestra un mensaje de error.
- d) El usuario introduce sus credenciales y, si son correctas, accede al menú principal. En ese momento puede elegir entre descargar un informe ya existente o cerrar sesión. Si las credenciales son incorrectas, el sistema muestra un mensaje de error y, a continuación, genera automáticamente un informe de incidencia)



DATOS DEL ALUMNO			FIRMA
APELLIDOS:			
Nombre:	D.N.I./ N.I.E.	Fecha:	
Código del ciclo: (1) IFCS02	Denominación completa del título: (1) Técnico Superior en Desarrollo de Aplicaciones Multiplataforma		
Clave o código del módulo: (1) 0487	Denominación completa del módulo profesional: (1) Entornos de Desarrollo		

48 ED R6 Dado el siguiente diagrama sobre los Pedidos de una tienda online:



¿Qué enunciado describe correctamente el comportamiento representado en el diagrama?

- a) Un pedido comienza en estado Pendiente. Si se confirma el pago, pasa a Pagado y después puede enviarse hasta quedar Entregado. Además, el pedido puede cancelarse tanto antes del pago como después de haberse pagado, pero no una vez enviado.
- b) Un pedido comienza en estado Pendiente. Desde ese estado solo puede pasar a Pagado, y una vez pagado puede enviarse, entregarse o cancelarse en cualquier momento, incluso después de haber sido enviado al cliente.
- c) Un pedido comienza en estado Pendiente. Si el cliente lo cancela, pasa primero a Pagado y después a Cancelado. Si no se cancela, el sistema lo marca como Entregado directamente después de confirmar el pago.
- d) Un pedido comienza en estado Pendiente. Después de confirmarse el pago, el pedido puede volver a Pendiente si todavía no se ha enviado, o pasar directamente a Entregado si el cliente confirma que desea recibirlo sin fase de envío.



DATOS DEL ALUMNO			FIRMA
APELLIDOS:			
Nombre:	D.N.I./ N.I.E.	Fecha:	
Código del ciclo: (1) IFCS02	Denominación completa del título: (1) Técnico Superior en Desarrollo de Aplicaciones Multiplataforma		
Clave o código del módulo: (1) 0487	Denominación completa del módulo profesional: (1) Entornos de Desarrollo		

49 ED R6 En UML, se quiere representar el comportamiento dinámico de un sistema: interacciones entre actores y sistema, intercambio de mensajes y cambios de estado de un objeto. ¿Qué tipo de diagrama quedaría fuera de ese grupo por pertenecer a otra perspectiva del modelado?

- a) Diagrama de casos de uso
- b) Diagrama de estados
- c) Diagrama de despliegue
- d) Diagrama de secuencia

50 ED R6 En UML, si se quiere representar una interacción entre objetos destacando principalmente quién se comunica con quién, en lugar del detalle temporal de cada mensaje, ¿qué afirmación es correcta?

- a) Usa líneas de vida verticales para mostrar el tiempo con detalle.
- b) Puede resultar más compacto y resalta los enlaces entre participantes.
- c) Describe los estados internos de un objeto ante distintos eventos.
- d) Se centra en los actores externos y en los objetivos del sistema.