




Nombre y Apellidos

DNI

FIRMA

Programación (código: 0485)

Bienvenido. Va a iniciar los distintos test que componen el examen.

- El examen se compone de 50 PREGUNTAS EN TOTAL. REPARTIDAS en 9 test.
- El examen se realizará en bolígrafo azul y se permite el uso de Tipp-Ex
- La respuesta correcta se redondeará de la siguiente forma:

- Cualquier respuesta que no esté marcada como se indica en el apartado anterior, NO SERÁ CONTABILIZADA y se considerará como NO RESPONDIDA.
- Cada test está relacionado con un Resultado de Aprendizaje distinto.
- Se ha de superar cada test con nota superior o igual a 5 para poder hacer media y superar el módulo.
- La nota de cada test es de 10 puntos que se reparten equitativamente entre el número de preguntas que lo componen. (Ejemplo si un test tiene 5 preguntas cada pregunta correcta puntúa 2 puntos. Si el test tiene 4 preguntas cada pregunta correcta puntúa 2.5 puntos)
- En cada test las respuestas incorrectas restan -33% de lo que puntúan la pregunta en dicho test.
- Las preguntas no contestadas no suman ni restan puntuación.
- En caso de no superar un resultado de aprendizaje la nota se calculará restando el número de resultados de aprendizaje no superados a 5. Siendo 0 si resulta un negativo.
- No se podrá salir del examen hasta que no hayan transcurrido los primeros 30 minutos. Asimismo, no se podrá acceder al examen una vez hayan transcurrido los primeros 30 minutos.

Número de preguntas por test y porcentaje para el cálculo de la nota final si los distintos resultados de aprendizaje son superados.

RA1 (4 preguntas) 8%
RA2 (4 preguntas) 8%
RA3 (7 preguntas) 14%
RA4 (7 preguntas) 14%
RA5 (5 preguntas) 10%
RA6 (9 preguntas) 18%
RA7 (8 preguntas) 16%
RA8 (3 preguntas) 6%
RA9 (3 preguntas) 6%

PONGA SU NOMBRE Y FIRME TODAS LAS HOJAS



Nombre y Apellidos

DNI

FIRMA

PROG-RA1: Reconoce la estructura de un programa informático, identificando y relacionando los elementos propios del lenguaje de programación utilizado.

1 PRRA1 ¿Cuál de las siguientes instrucciones sería CORRECTA?

A) byte b = 255;

B) byte b = 128;

C) byte b = -255;

D) byte b = -128;

2 PRRA1 ¿Cuál es la salida del siguiente programa en Java?

```
PUBLIC CLASS CONVERSIONTEST {  
    PUBLIC STATIC VOID MAIN(STRING[] ARGS) {  
        INT X = 5;  
        DOUBLE Y = X / 2;  
        SYSTEM.OUT.PRINTLN(Y);  
    }  
}
```

A) 2.5

B) 2.0

C) Error de compilación

D) 2

3 PRRA1 ¿Cuál es la salida del siguiente programa en Java?

```
PUBLIC CLASS MAIN {  
    PUBLIC STATIC VOID MAIN(STRING[] ARGS) {  
        INT A = 5, B = 10;  
        IF ( !(A > B) && (A + 5 == B) )  
            {  
                SYSTEM.OUT.PRINT("A");  
            }  
        ELSE  
            {  
                SYSTEM.OUT.PRINT("B");  
            }  
        SYSTEM.OUT.PRINT("C");  
    }  
}
```

A) A

B) B

C) BC

D) Ninguna de las otras respuestas

4 PRRA1 Indica cuál de los siguientes pares de instrucciones son INCORRECTOS.

A) int b2 = 1; byte i2 = b2;

B) byte b1 = 1; int i1 = b1;

C) float b3 = 1F; double i3 = b3;

D) long b4 = 1; float i4 = b4;



Nombre y Apellidos

DNI

FIRMA

PROG-RA2: Escribe y prueba programas sencillos, reconociendo y aplicando los fundamentos de la programación orientada a objetos.

1 PRRA2 Dado el siguiente fragmento de código, ¿cuál es el valor final de resultado?

```
PUBLIC CLASS PROGRAMA {  
    PUBLIC STATIC VOID MAIN(STRING[] ARGS) {  
        STRING MENSAJE = "PROGRAMAR";  
        INT RESULTADO = MENSAJE.INDEXOF('G') + MENSAJE.LENGTH();  
        SYSTEM.OUT.PRINTLN(RESULTADO);  
    }  
}
```

A) 12

B) 10

C) 9

D) 11

2 PRRA2 ¿Qué afirmación sobre constructores en Java es cierta

A) Cada clase puede tener un solo constructor

B) Si no se define ningún constructor, Java genera uno que recibe parámetros

C) El constructor se llama automáticamente al instanciar un objeto

D) Los constructores pueden devolver valores de cualquier tipo

3 PRRA2 ¿Qué ocurre al ejecutar el siguiente código?

```
IMPORT JAVA.UTIL.SCANNER;  
PUBLIC CLASS LECTURA {  
    PUBLIC STATIC VOID MAIN(STRING[] ARGS)  
    {  
        SCANNER SC = NEW SCANNER(SYSTEM.IN);  
        INT EDAD = SC.NEXTINT();  
        SYSTEM.OUT.PRINTLN("EDAD INTRODUCIDA: " + EDAD);  
    }  
}
```

A) El programa lanza error de compilación si no se introduce nada por teclado

B) El programa muestra siempre "Edad introducida: 0" si se pulsa Enter sin escribir

C) El programa compila, pero lanza excepción en tiempo de ejecución si se introduce texto no numérico

D) El programa convierte automáticamente cualquier entrada en número



Nombre y Apellidos

DNI

FIRMA

4 PRRA2 ¿Qué salida produce el siguiente código en Java?

```
PUBLIC CLASS TEST {  
PUBLIC STATIC VOID MAIN(STRING[] ARGS)  
{  
    STRING S = NEW STRING("HOLA");  
    STRING T = "HOLA";  
    SYSTEM.OUT.PRINTLN(S == T);  
}}
```

- A) true
- B) false
- C) null
- D) Error en tiempo de ejecución



Nombre y Apellidos

DNI

FIRMA

PROG-RA3: Escribe y depura código, analizando y utilizando las estructuras de control del lenguaje.

1 PRRA3 ¿Cuál de las siguientes opciones lanza una excepción personalizada correctamente?

- A) throw new Exception("Error personalizado");
- B) throw "Error personalizado";
- C) Exception e = new Exception();e.throw();
- D) throw Exception("Error personalizado");

2 PRRA3 ¿Cuál es la salida del siguiente código?

```
INT X = 10;  
IF (X > 5) IF (X < 15)  SYSTEM.OUT.PRINTLN("A");  
ELSE  SYSTEM.OUT.PRINTLN("B");
```

- A) A
- B) B
- C) No se imprime nada
- D) Error de compilación

3 PRRA3 ¿Qué hace el siguiente fragmento de código?

```
INT I = 0;DO {  ++I;  } WHILE (I < 0);SYSTEM.OUT.PRINTLN(I);
```

- A) Imprime 0
- B) Imprime 1
- C) No imprime nada
- D) Error en tiempo de ejecución

4 PRRA3 ¿Qué muestra el siguiente código?

```
INT X=2;STRING LETRA="";  
SWITCH (X) {  
    CASE 1: LETRA += "A";BREAK;  
    CASE 2: LETRA += "B";  
    CASE 3: LETRA += "C"; BREAK;  
    DEFAULT: LETRA += "D";  }  
SYSTEM.OUT.PRINTLN(LETRA);
```

- A) B
- B) ABC
- C) BD
- D) BC



Nombre y Apellidos

DNI

FIRMA

5 PRRA3 ¿Qué ocurre con el siguiente código?

```
FOR (INT I = 0; I < 5; I++) { IF (I == 3) BREAK; SYSTEM.OUT.PRINT(I + " ");}
```

- A) Imprime 0 1 2 3
- B) Imprime 0 1 2
- C) Imprime 0 1 2 3 4
- D) No imprime nada

6 PRRA3 ¿Qué ocurre con este código?

```
TRY { INT X = 10 / 0; }  
CATCH (ARITHMETICEXCEPTION E)  
{ SYSTEM.OUT.PRINTLN("ERROR: DIVISIÓN POR CERO");}
```

- A) No se imprime nada
- B) El programa lanza una excepción no controlada
- C) Se imprime "Error: división por cero"
- D) El programa entra en bucle infinito

7 PRRA3 ¿Qué salida produce este código?

```
INT[] ARR = {1, 2, 3, 4, 5};  
FOR (INT I : ARR)  
{ IF (I == 3) CONTINUE; SYSTEM.OUT.PRINT(I + " ");}
```

- A) 1 2 3 4 5
- B) 1 2 4 5
- C) 1 2 3 4
- D) 1 2 5



Nombre y Apellidos

DNI

FIRMA

PROG-RA4: Desarrolla programas organizados en clases analizando y aplicando los principios de la programación orientada a objetos.

1 PRRA4 ¿Qué hace el modificador private en un atributo?

- A) Lo hace visible solo dentro del paquete
- B) Permite el acceso solo desde clases hijas
- C) Restringe el acceso solo a la clase donde se define
- D) Lo vuelve constante

2 PRRA4 ¿Qué hace el siguiente código?

```
CLASS UTIL {  
    STATIC VOID MOSTRARMENSAJE() {  
        SYSTEM.OUT.PRINTLN("HOLA");  }}
```

- A) Define un método que solo se puede llamar desde objetos
- B) Declara un atributo privado
- C) Crea un método que se puede llamar sin instanciar la clase
- D) Crea una excepción

3 PRRA4 ¿Qué hace este constructor?

```
CLASS PERSONA {  
    STRING NOMBRE;  
    PERSONA(STRING N)  
    {    NOMBRE = N;  }}
```

- A) Crea una clase abstracta
- B) Asigna un valor inicial a nombre
- C) Declara un método estático
- D) Sobrescribe el método toString()

4 PRRA4 ¿Qué imprime este código?

```
CLASS ANIMAL {  
    STRING NOMBRE = "PERRO";  
    PUBLIC STRING TOSTRING() {  
        RETURN "GUAU";  }}  
PUBLIC CLASS MAIN {  
    PUBLIC STATIC VOID MAIN(STRING[] ARGS) {  
        ANIMAL A = NEW ANIMAL();  
        SYSTEM.OUT.PRINTLN(A);  
    }}
```



Nombre y Apellidos

DNI

FIRMA

- A) a
- B) GUAU
- C) Animal
- D) Perro

5 PRRA4 ¿Qué imprime este código?

```
CLASS A {  
    STATIC INT CONTADOR = 0;  
    A() { CONTADOR++; }  
    PUBLIC CLASS MAIN {    PUBLIC STATIC VOID MAIN(STRING[] ARGS)  
    {    NEW A();    NEW A();    SYSTEM.OUT.PRINTLN(A.CONTADOR);  
    }  
}
```

- A) 1
- B) 2
- C) 0
- D) Nada. Da un error de compilación

6 PRRA4 ¿Qué ocurre con este código?

```
CLASS FIGURA {  
    VOID DIBUJAR() { SYSTEM.OUT.PRINTLN("FIGURA"); }  
    CLASS CIRCULO EXTENDS FIGURA {  
        VOID DIBUJAR() { SYSTEM.OUT.PRINTLN("CÍRCULO"); }  
    }
```

- A) Circulo hereda el método dibujar() sin modificarlo
- B) Se produce una sobrecarga del método
- C) El método dibujar() de Figura es sobrescrito
- D) Figura es una clase abstracta

7 PRRA4 ¿Qué permite el uso de paquetes (packages) en Java?

- A) Ejecutar clases sin compilarlas
- B) Reutilizar librerías y organizar clases
- C) Crear clases abstractas automáticamente
- D) Evitar el uso de métodos estáticos



Nombre y Apellidos

DNI

FIRMA

PROG-RA5: Realiza operaciones de entrada y salida de información, utilizando procedimientos específicos del lenguaje y librerías de clases.

1 PRRA5 ¿Cuál es la función del siguiente código?

```
FILEWRITER FW = NEW FILEWRITER("DATOS.TXT", TRUE);  
FW.WRITE("HOLA\n");  
FW.CLOSE();
```

- A) Sobrescribe el archivo con el texto
- B) Borra el archivo datos.txt
- C) Añade el texto al final del archivo
- D) Solo crea el archivo sin escribir

2 PRRA5 ¿Qué habría que sustituir por los tres puntos (entre paréntesis) para que saliera por consola "CLIC" al pulsar el botón (usando JAVA FX)

```
BUTTON B = NEW BUTTON; B.SETONACTION( ... );
```

- A) new EventHandler<>() {public void handle(ActionEvent e) {System.out.println("CLIC");}}
- B) System.out.println("CLIC")
- C) () -> System.out.println("CLIC")
- D) Ningún código es correcto.

3 PRRA5 ¿Qué hace este código?

```
SCANNER SC = NEW SCANNER(System.in); INT EDAD = SC.NEXTINT();
```

- A) Lee un número entero desde la consola
- B) Lee una línea de texto
- C) Imprime un número en pantalla
- D) Lee un archivo de texto

4 PRRA5 ¿Qué imprime este código? SYSTEM.OUT.PRINTF("PRECIO: %.2F €", 12.3456);

- A) Precio: 12.3 €
- B) Precio: 12.34 €
- C) Precio: 12.35 €
- D) Precio: 12.3456 €



Nombre y Apellidos

DNI

FIRMA

5 PRRA5 ¿Qué ocurre si se omite el método LAUNCH(ARGS) en una aplicación JavaFX?

```
PUBLIC CLASS APP EXTENDS APPLICATION {
```

```
@OVERRIDE PUBLIC VOID START(STAGE PRIMARYSTAGE) {  
SYSTEM.OUT.PRINTLN("INICIANDO...");
```

```
}}
```

A) El método start() se ejecuta automáticamente

B) La aplicación no compila

C) La interfaz gráfica no se inicia

D) Se crea un Scene vacío por defecto



Nombre y Apellidos

DNI

FIRMA

PROG-RA6: Escribe programas que manipulen información seleccionando y utilizando tipos avanzados de datos.

1 PRRA6 Indica cuál de las siguientes instrucciones es INCORRECTA.

- A) `int[] x4 = new int[1];`
- B) `int[] x1,x2={1};`
- C) `int[] x3 = [1];`
- D) `int x5=10;int[] x6=new int[x5];`

2 PRRA6 ¿Qué clase se usa para procesar documentos en formato JSON en Java?

- A) `DocumentBuilder`
- B) `Gson`
- C) `PrintWriter`
- D) `TextField`

3 PRRA6 ¿Qué colección permite almacenar elementos duplicados y mantiene el orden de inserción?

- A) `HashSet`
- B) `TreeSet`
- C) `LinkedList`
- D) `HashMap`

4 PRRA6 ¿Qué muestra el siguiente código?

```
LIST<INTEGER> L = LIST.OF(1, 2, 3, 4);  
INT R = L.STREAM().FILTER(X -> X % 2 != 0).REDUCE(1, (A, B) -> A * B);  
SYSTEM.OUT.PRINTLN(R);
```

- A) 0
- B) 3
- C) 8
- D) 24

5 PRRA6 ¿Qué ocurre al ejecutar este código?

```
LIST<INTEGER> LISTA = NEW ARRAYLIST<>();  
LISTA.ADD(2);  
SYSTEM.OUT.PRINTLN(LISTA.GET(2));
```

- A) Lanza `IndexOutOfBoundsException`
- B) Imprime 2
- C) No muestra nada
- D) Da error al compilar



Nombre y Apellidos

DNI

FIRMA

6 PRRA6 ¿Qué ocurre con el siguiente código?

```
LIST<STRING> LISTA = NEW ARRAYLIST<>();  
LISTA.ADD("UNO");  
LISTA.ADD("DOS");  
ITERATOR<STRING> IT = LISTA.ITERATOR();  
WHILE (IT.HASNEXT()) {    SYSTEM.OUT.PRINT(IT.NEXT() + " ");}
```

- A) Imprime "uno "
- B) Imprime "uno dos "
- C) Lanza una excepción
- D) No muestra nada

7 PRRA6 ¿Qué patrón detecta esta expresión regular?

```
STRING REGEX = "\\D{3}-\\D{2}-\\D{4}";
```

- A) Fechas con barras
- B) Una dirección de correo electrónico
- C) Un número de seguridad social (formato 123-45-6789)
- D) Una URL

8 PRRA6 ¿Qué resultado devuelve este código?

```
PATTERN P = PATTERN.COMPILE("[A-Z]{2}\\D{3}");  
MATCHER M = P.MATCHER("AB123");  
SYSTEM.OUT.PRINTLN(M.MATCHES());
```

- A) false
- B) true
- C) Lanza una excepción PatternSyntaxException
- D) No muestra nada

9 PRRA6 ¿Qué ventaja tiene un HashSet sobre un ArrayList?

- A) Mantiene el orden alfabético
- B) Permite claves y valores
- C) No permite elementos duplicados
- D) Mejora el rendimiento de acceso por índice.



Nombre y Apellidos

DNI

FIRMA

PROG-RA7: Desarrolla programas aplicando características avanzadas de los lenguajes orientados a objetos y del entorno de programación.

1 PRRA7 ¿Cuál es el resultado de este código?

```
PUBLIC CLASS MAIN { PUBLIC STATIC VOID MAIN(STRING[] ARGS) {  
    CLASS A { INT VALOR = 1;}  
    CLASS B EXTENDS A { INT VALOR = 2;}  
    SYSTEM.OUT.PRINTLN(((A) NEW B()).VALOR); }}
```

- A) 1
- B) 2
- C) Error de compilación
- D) null

2 PRRA7 ¿Qué afirmación de las siguientes es correcta?

- A) Una clase puede heredar de múltiples superclases
- B) Una clase no puede implementar más de una interface
- C) Una interface puede heredar de otra interface.
- D) Una interfaz puede heredar de una clase

3 PRRA7 ¿Qué hace este constructor?

```
CLASS HIJO EXTENDS PADRE { HIJO() { SUPER(); }}
```

- A) Llama al constructor de la subclase
- B) Llama al constructor de la superclase
- C) Es innecesario y produce error
- D) Ignora la herencia

4 PRRA7 ¿Qué ocurre con este código?

```
CLASS A { FINAL VOID METODO() {} } CLASS B EXTENDS A { VOID METODO() {} }
```

- A) Sobrescribe metodo() correctamente
- B) Error de compilación
- C) Hace que metodo() sea abstracto
- D) Hace que metodo() sea estático

5 PRRA7 ¿Qué ocurre con este código?

```
@FUNCTIONALINTERFACE PUBLIC INTERFACE FUNCIONAL { VOID M1(); VOID M2(); }
```

- A) Es correcto
- B) Da un error de compilación



Nombre y Apellidos

DNI

FIRMA

C) Habría que declarar la interface como abstract para que compilara.

D) Se podría sustituir alguna de sus implementaciones por una expresión lambda.

6 PRRA7 ¿Qué ocurre si una clase implementa una interfaz y no sobrescribe todos sus métodos?

A) Se puede compilar sin problema

B) Lanza excepción en tiempo de ejecución

C) Debe ser declarada como abstract

D) Debe ser final

7 PRRA7 ¿Qué palabra clave se usa en Java para impedir que una clase sea heredada?

A) private

B) abstract

C) final

D) static

8 PRRA7 ¿Qué sale al ejecutar el siguiente código?

```
CLASS ANIMAL { VOID SONIDO()  
{ SYSTEM.OUT.PRINTLN("GROAR"); }}  
CLASS PERRO EXTENDS ANIMAL { VOID SONIDO()  
{ SYSTEM.OUT.PRINTLN("GUAU"); }}  
PUBLIC CLASS MAIN {  
PUBLIC STATIC VOID MAIN(STRING[] ARGS)  
{ ANIMAL A = NEW PERRO();  
A.SONIDO();  
}}
```

A) GROAR

B) No ejecuta. Da un error de compilación

C) GUAU

D) Lanza excepción NullPointerException



Nombre y Apellidos

DNI

FIRMA

PROG-RA8: Utiliza bases de datos orientadas a objetos, analizando sus características y aplicando técnicas para mantener la persistencia de la información.

1 PRRA8 ¿Qué clase de Java suele utilizarse para mapear una entidad persistente en una base de datos orientada a objetos con Hibernate?

- A) Serializable
- B) Object
- C) Una clase Java con anotaciones como @Entity
- D) HashMap

2 PRRA8 ¿Qué sucede si un objeto contiene referencias a otros objetos y se guarda en una base de datos orientada a objetos?

- A) Solo se guarda el objeto principal
- B) Las referencias deben eliminarse manualmente
- C) Pueden almacenarse en cascada automáticamente
- D) Hibernate no lo permite

3 PRRA8 ¿Qué tipo de consulta es típica en una base de datos orientada a objetos con Hibernate?

- A) from Clase where campo = :valor
- B) SELECT * FROM Clase WHERE campo = ?
- C) db.collection.find({})
- D) match (n) where n.valor = ?



Nombre y Apellidos

DNI

FIRMA

PROG-RA9: Gestiona información almacenada en bases de datos manteniendo la integridad y consistencia de los datos.

1 PRRA9 ¿Cuál es el método correcto (código dentro del paréntesis del while) para recorrer todos los resultados de una consulta JDBC?

ResultSet rs = ...;while (_____) { // PROCESAR FILA}

- A) rs.hasNext()
- B) rs.next()
- C) rs.more()
- D) rs.advance()

2 PRRA9 Dado este fragmento de código JDBC, ¿qué hace PS.SETSTRING(1, "X"); ?

PREPAREDSTATEMENT PS = CON.PREPARESTATEMENT ("SELECT * FROM EMPLEADOS WHERE NOMBRE = ? AND APELLIDO=?"); PS.SETSTRING(1, "X");

- A) Asigna el valor "X" a ps
- B) Inserta una nueva fila con nombre y apellido "X"
- C) Reemplaza el primer signo de interrogación con "X"
- D) Ejecuta la consulta y devuelve resultados

3 PRRA9 ¿Qué ventajas tiene usar sentencias preparadas PreparedStatement al consultar una base de datos?

- A) Evitan ataques de inyección de SQL
- B) Permiten mantener abierta la conexión con varias Bases de datos a la vez
- C) Permiten guardar a disco las consultas para poder reutilizarse en otra ocasión
- D) Son imprescindibles para trabajar con ORMs como Hibernate