

Navegación por el cuestionario

1	2	3	4	5	6	7	8	9	10
✓		✓	✓	✓	✓	✓			

Finalizar revisión

Comenzado el	jueves, 21 de febrero de 2019, 20:43
Estado	Finalizado
Finalizado en	viernes, 22 de febrero de 2019, 16:38
Tiempo empleado	19 horas 54 minutos
Calificación	3,13 de 10,00 (31%)

Pregunta 1 Correcta

Puntúa 0,50 sobre 1,00

▢ Marcar pregunta

Tenemos el siguiente código:

```
Casa miCasa = new Casa();  
ArrayList<Casa> listaCasas = new ArrayList<Casa>();  
listaCasas.add(miCasa);  
miCasa.superficie = 55.55;
```

Si la clase Casa se define:

```
public class Casa {  
    public String direccion;  
    public double superficie;  
    public int edad;  
}
```

¿Qué valor tendrán las siguientes expresiones?

Expresión	Valor
miCasa.direccion	null ✓
miCasa.superficie	55.55 ✓
listaCasas.get(0).edad	0 ✓
listaCasas.get(0).superficie	55.55 ✓

Correcta
Puntos para este envío: 1,00/1,00. Contando con los intentos anteriores, daría 0,50/1,00.

Pregunta 2 Incorrecta

Puntúa 0,00 sobre 1,00

▢ Marcar pregunta

Tenemos el siguiente código:

```
Circulo miCirculo = new Circulo();  
ArrayList<Circulo> listaCirculos = new ArrayList<Circulo>();  
for (int i = 1; i <= 10; i++) {  
    listaCirculos.add(new Circulo(i, "verde"));  
}
```

Si la clase Circulo tiene los siguientes atributos y métodos:

```
public class Circulo {  
    private double radio;  
    private String color;  
  
    public void setRadio(double radio)  
    public void setColor(String color)  
    public double getRadio()  
    public String getColor()  
    public String toString()  
    public double calculaArea()  
    public boolean esIgual(Circulo otro)  
}
```

¿Qué código necesitaríamos para realizar las siguientes acciones? (no se debe incluir un ; al final)

Acción	Código
Cambiar el color del tercer circulo de listaCirculos para que sea "rojo"	listaCirculos(3).setColor("rojo") ✗
Obtener el radio del último circulo de listaCirculos	listaCirculos.get(listaCirculos.size()).radio ✗
Calcular el area del segundo elemento de la lista	listaCirculos.get(2).calculaArea ✗
Comprobar si miCirculo y el quinto circulo de listaCirculos son iguales	miCirculo.esIgual(listaCirculos(5)) ✗

Incorrecta
Puntos para este envío: 0,00/1,00.

Pregunta 3 Correcta

Puntúa 1,00 sobre 1,00

▢ Marcar pregunta

Tenemos el siguiente código:

```
ArrayList<Circulo> listaCirculos = new ArrayList<Circulo>();  
Circulo unCirculo = new Circulo(2.2, "blanco");  
for (int i = 0; i < 5; i++) {  
    listaCirculos.add(unCirculo);  
}
```

Si la clase Circulo se define:

```
public class Circulo {  
    public double radio;  
    public String color;  
  
    public Circulo(double radio, String color) {  
        this.radio = radio;  
        this.color = color;  
    }  
  
    ...  
}
```

Indica las afirmaciones correctas.

Seleccione una o más de una:

a. Cada uno de los elementos del array contiene el valor null

b. Se crea una lista con 5 elementos ✓

c. Todos los elementos de la lista son iguales pero independientes. Si cambiamos el radio del primer elemento, el radio del resto de elementos no cambia.

d. Todos los elementos de la lista apuntan al mismo objeto. Si cambiamos el radio del primer elemento cambia el radio de todos los elementos ✓

Respuesta correcta

La respuesta correcta es: Se crea una lista con 5 elementos. Todos los elementos de la lista apuntan al mismo objeto. Si cambiamos el radio del primer elemento

cambia el radio de todos los elementos

Correcta

Puntos para este envío: 1,00/1,00.

Pregunta 4

Correcta

Puntúa 0,75 sobre
1,00

▼ Marcar
pregunta

Tenemos el siguiente código:

```
Circulo miCirculo = new Circulo();
ArrayList<Circulo> listaCirculo = new ArrayList<Circulo>();
Circulo tuCirculo = new Circulo(5.5, "rojo");
listaCirculos.add(new Circulo(miCirculo));
listaCirculos.add(0, tuCirculo);
tuCirculo.radio = 3.3;
miCirculo.color = "azul"
```

Si la clase Circulo se define:

```
public class Circulo {
    public double radio;
    public String color;

    public Circulo() {
        this(0.0, "");
    }

    public Circulo(double radio, String color) {
        this.radio = radio;
        this.color = color;
    }

    public Circulo(Circulo otro) {
        this(otro.radio, otro.color);
    }
    ...
}
```

¿Qué valor tendrán las siguientes expresiones? Recuerda escribir los valores de acuerdo al tipo que les corresponda: 7.0 en vez de 7 para los double, comillas dobles para las cadenas de caracteres...:

Expresión	Valor	Expresión	Valor
miCirculo.radio	0.0 ✓	listaCirculos.get(0).radio	3.3 ✓
miCirculo.color	"azul" ✓	listaCirculos.get(0).color	"rojo" ✓
tuCirculo.radio	3.3 ✓	listaCirculos.get(1).radio	0.0 ✓
tuCirculo.color	"rojo" ✓	listaCirculos.get(1).color	" " ✓

Correcta

Puntos para este envío: 1,00/1,00. Contando con los intentos anteriores, daría 0,75/1,00.

Pregunta 5

Correcta

Puntúa 0,88 sobre
1,00

▼ Marcar
pregunta

Tenemos el siguiente código:

```
Cuenta miCuenta = new Cuenta(1234, 1000.55, "Pedro Aristi");
ArrayList<Cuenta> listaCuentas = new ArrayList<Cuenta>();
Cuenta tuCuenta;
listaCuentas.add(new Cuenta());
listaCuentas.add(miCuenta);
listaCuentas.get(1).numero = 1111;
```

Si la clase Cuenta se define:

```
public class Cuenta {
    public int numero;
    public double saldo;
    public String titular;

    public Cuenta() {
        this(0, 0.0, "");
    }

    public Cuenta(int numero, double saldo, String titular) {
        this.numero = numero;
        this.saldo = saldo;
        this.titular = titular;
    }

    ...
}
```

¿Qué valor tendrán las siguientes expresiones? Recuerda escribir los valores de acuerdo al tipo que les corresponda: 7.0 en vez de 7 para los double, comillas dobles para las cadenas de caracteres...:

Expresión	Valor	Expresión	Valor
miCuenta.numero	1111 ✓	listaCuentas.get(1).numero	1111 ✓
miCuenta.saldo	1000.55 ✓	listaCuentas.get(1).titular	"Pedro Aristi" ✓
miCuenta.titular	"Pedro Aristi" ✓	listaCuentas.get(0).saldo	0.0 ✓
tuCuenta	null ✓	listaCuentas.get(0).titular	✓

Correcta

Puntos para este envío: 1,00/1,00. Contando con los intentos anteriores, daría 0,88/1,00.

Pregunta 6

Incorrecta

Puntúa 0,00 sobre
1,00

▼ Marcar
pregunta

Tenemos la clase Concesionario y queremos añadir un constructor que reciba el nombre del concesionario y el nombre de un fichero, lea los datos de diferentes coches y los almacene el el ArrayList de la clase. El formato del fichero será el siguiente:

```
Opel Astra 2015
Hyundai Ioniq 2018
Renault Megane 2014
Volkswagen Golf 2013
Ford Focus 2016
```

Si ejecutamos las siguientes líneas:

```
Concesionario unConcesionario = new Concesionario("Mas Motor", "concesionario.txt");
System.out.println(unConcesionario);
```

Por consola se verá:

```
MAS MOTOR
1: Opel Astra (2015)
2: Hyundai Ioniq (2018)
3: Renault Megane (2014)
4: Volkswagen Golf (2013)
5: Ford Focus (2016)
```

Respuesta: (penalty regime: 10, 20, ... %)

Reiniciar respuesta

```
1 import java.util.*;
2 import java.io.*;
3
4 public class Concesionario {
5     // Atributos
6     private String nombre;
7     private ArrayList<Coche> lista;
8
9     // Constructor
10    public Concesionario(String concesionario) {
11        nombre = concesionario;
12        lista = new ArrayList<Coche>();
13    }
14
15    // METODO QUE HAY QUE CORREGIR
16    public Concesionario(String nombre, String fichero) throws FileNotFoundException {
17        this.nombre = nombre;
18        Scanner leerFichero = new Scanner(new File(fichero));
19        while (leerFichero.hasNext()) {
20            String marca = leerFichero.next();
```

Test	Expected	Got
<pre>try { Concesionario unConcesionario = new Concesionario("Mas Motor", "concesionario.txt"); System.out.println(unConcesionario); } catch (FileNotFoundException e) { System.out.println("ERROR"); }</pre>	MAS MOTOR 1: Opel Astra (2015) 2: Hyundai Ioniq (2018) 3: Renault Megane (2014) 4: Volkswagen Golf (2013) 5: Ford Focus (2016)	***Error*** Exception in thread "main" at Concesionario.<init> at __Tester__.runTest at __Tester__.main

Prueba cancelada debido a un error.
Your code must pass all tests to earn any marks. Try again.

Mostrar diferencias

Question author's solution:

```
import java.util.*;
import java.io.*;

public class Concesionario {
    // Atributos
    private String nombre;
    private ArrayList<Coche> lista;

    // Constructor
    public Concesionario(String concesionario) {
        nombre = concesionario;
        lista = new ArrayList<Coche>();
    }

    public Concesionario(String nombre, String fichero) throws FileNotFoundException {
        this.nombre = nombre;
        lista = new ArrayList<Coche>();
        Scanner leerFichero = new Scanner(new File(fichero));
        while (leerFichero.hasNext()) {
            String marca = leerFichero.next();
            String modelo = leerFichero.next();
            int year = leerFichero.nextInt();
            Coche unCoche = new Coche(marca, modelo, year);
            lista.add(unCoche);
        }
    }

    // Metodos
    public String toString() {
        String formato = nombre.toUpperCase() + "\n";
        int i = 1;
        for (Coche unCoche : lista) {
            formato += i + ": " + unCoche + "\n";
            i++;
        }
        return formato;
    }
}
```

Incorrecta

Puntos para este envío: 0,00/1,00.

Pregunta 7

Incorrecta

Puntúa 0,00 sobre

1,00

▼ Marcar pregunta

Tenemos el siguiente código:

```
Concesionario miConcesionario = new Concesionario("Mi concesionario");
Coche miCoche = new Coche("Opel", "Corsa", 2015);
miConcesionario.addCoche(miCoche);
miConcesionario.setCoche(0, "Seat", "Ibiza", 2010);
System.out.println(miCoche);
System.out.println(miConcesionario.getCoche(0));
```

Corrige el método addCoche de la clase Concesionario para que el resultado que se muestre por consola sea:

```
Opel Corsa (2015)
Seat Ibiza (2010)
```

Ten en cuenta que la clase Coche tiene los siguientes métodos:

```
public class Coche {
    private String marca;
    private String modelo;
    private int year;

    public Coche(String marca, String modelo, int year)
    public void setMarca(String marca)
    public void setModelo(String modelo)
    public void setYear(int year)
    public String getMarca()
    public String getModelo()
    public int getYear()
    public String toString()
}
```

Respuesta: (penalty regime: 10, 20, ... %)

Reiniciar respuesta

```
1 import java.util.*;
2
3 public class Concesionario {
4     // Atributos
5     private String nombre;
6     private ArrayList<Coche> lista;
7
8     // Constructor
9     public Concesionario(String concesionario) {
10        nombre = concesionario;
11        lista = new ArrayList<Coche>();
12    }
13
14    // Metodos
15    public void setCoche(int indice, String marca, String modelo, int year) {
16        lista.get(indice).setMarca(marca);
17        lista.get(indice).setModelo(modelo);
18        lista.get(indice).setYear(year);
19    }
}
```

Test	Expected	Got
<pre> Concesionario miConcesionario = new Concesionario("Mi concesionario"); Coche miCoche = new Coche("Opel", "Corsa", 2015); miConcesionario.addCoche(miCoche); miConcesionario.setCoche(0, "Seat", "Ibiza", 2010); System.out.println(miCoche); System.out.println(miConcesionario.getCoche(0)); </pre>	Opel Corsa (2015) Seat Ibiza (2010)	Seat Ibiza (2010)
<pre> Concesionario miConcesionario = new Concesionario("Mi concesionario"); Coche miCoche = new Coche("Opel", "Corsa", 2015); miConcesionario.addCoche(miCoche); miCoche.setMarca("Seat"); miCoche.setModelo("Ibiza"); miCoche.setYear(2010); System.out.println(miCoche); System.out.println(miConcesionario.getCoche(0)); </pre>	Seat Ibiza (2010) Opel Corsa (2015)	Seat Ibiza (2010)

Your code must pass all tests to earn any marks. Try again.

[Mostrar diferencias](#)

Question author's solution:

```

import java.util.*;

public class Concesionario {
    // Atributos
    private String nombre;
    private ArrayList<Coche> lista;

    // Constructor
    public Concesionario(String concesionario) {
        nombre = concesionario;
        lista = new ArrayList<Coche>();
    }

    // Metodos
    public void setCoche(int indice, String marca, String modelo, int year) {
        lista.get(indice).setMarca(marca);
        lista.get(indice).setModelo(modelo);
        lista.get(indice).setYear(year);
    }

    public void addCoche(Coche unCoche) {
        Coche otroCoche = new Coche(unCoche.getMarca(), unCoche.getModelo(), unCoche.getYear());
        lista.add(otroCoche);
    }

    public Coche getCoche(int indice) {
        return lista.get(indice);
    }
}

```

incorrecta

Puntos para este envío: 0,00/1,00.

Pregunta 8

Sin contestar

Puntúa 0,00 sobre
1,00

[▼ Marcar
pregunta](#)

Ordena el método cochesYear de la clase Concesionario para que reciba como parámetro un año y devuelva un objeto de la clase Concesionario con todos los coches de ese año que hay en el concesionario. El nombre del concesionario será "Coches del añoElegido".

```

public class Concesionario {
    ...
    public Concesionario cochesYear(int year) {
        
        
        
        
        
        
        
        
        
        
        
        
        
    }
}

```

Si el concesionario tiene los siguientes coches y se elige el año 2015:

```

Opel Astra 2015
Hyundai Ioniq 2018
Opel Meriva 2015
Renault Megane 2015
Opel Vectra 2017
Volkswagen Golf 2013
Ford Focus 2016

```

El método devolverá un Concesionario con los siguientes coches:

```

Opel Astra 2015
Opel Meriva 2015
Renault Megane 2015

```

Ordena

mismoYear.addCoche(copia);	for (Coche esteCoche : lista) {
if (esteYear == year) {	int esteYear = esteCoche.getYear();
return mismoYear;	Coche copia = new Coche(esteCoche);

Concesionario mismoYear = new Concesionario("Coches del " + year);

Respuesta incorrecta.

La respuesta correcta es:

Ordena el método cochesYear de la clase Concesionario para que reciba como parámetro un año y devuelva un objeto de la clase Concesionario con todos los coches de ese año que hay en el concesionario. El nombre del concesionario será "Coches del añoElegido".

```

public class Concesionario {
    ...
    public Concesionario cochesYear(int year) {
        [Concesionario mismoYear = new Concesionario("Coches del " + year);]
        [for (Coche esteCoche : lista) {}]
        [int esteYear = esteCoche.getYear();]
        [if (esteYear == year) {}]
        [Coche copia = new Coche(esteCoche);]
        [mismoYear.addCoche(copia);]
        [return mismoYear;]
    }
}

```

Si el concesionario tiene los siguientes coches y se elige el año 2015:

```

Opel Astra 2015
Hyundai Ioniq 2018

```

Opel Meriva 2015
Renault Megane 2015
Opel Vectra 2017
Volkswagen Golf 2013
Ford Focus 2016

El método devolverá un Concesionario con los siguientes coches:

Opel Astra 2015
Opel Meriva 2015
Renault Megane 2015

Ordena

Pregunta 9

Incorrecta

Puntúa 0,00 sobre

1,00

▼ Marcar pregunta

Tenemos el siguiente código:

```
Banco unBanco = new Banco("banco");
unBanco.addCuenta(1234, 1000.55, "Pedro");
unBanco.addCuenta(4321, 222.22, "Miren");
unBanco.addCuenta(2222, 4444.44, "Ana");
System.out.print(unBanco);
System.out.println("La cuenta con el saldo menor es: " + unBanco.saldoMenor());
```

Crea el método saldoMenor de la clase Banco para que devuelva un String con la información de la cuenta con menor saldo. Recuerda que para formatear un objeto tenemos el método toString().

El resultado del código será:

```
BANCO
Pedro(1234): 1000.55 euros
Miren(4321): 222.22 euros
Ana(2222): 4444.44 euros
La cuenta con el saldo menor es: Miren(4321): 222.22 euros
```

Ten en cuenta que la clase Cuenta tiene los siguientes métodos:

```
public class Cuenta {
    public int numero;
    public double saldo;
    public String titular;

    public Cuenta(int numero, double saldo, String titular)
    public int getNumero()
    public double getSaldo()
    public String getTitular()
    public String toString()
}
```

NOTA: Para ejecutar el programa puedes usar los siguientes ficheros. [Cuenta.java](#)

Respuesta: (penalty regime: 10, 20, ... %)

Reiniciar respuesta

```
1 import java.util.*;
2
3 public class Banco {
4     // Atributos
5     private String nombre;
6     private ArrayList<Cuenta> lista;
7
8     // Constructor
9     public Banco(String nombre) {
10         this.nombre = nombre;
11         lista = new ArrayList<Cuenta>();
12     }
13
14     public void addCuenta(int numero, double saldo, String titular) {
15         Cuenta unaCuenta = new Cuenta(numero, saldo, titular);
16         lista.add(unaCuenta);
17     }
18
19     // toString
20     public String toString() {
```

Syntax Error(s)

```
_Tester_.java:46: error: cannot find symbol
System.out.println("La cuenta con el saldo menor es: " + unBanco.saldoMenor());
                                              ^
symbol:   method saldoMenor()
location: variable unBanco of type Banco
1 error
```

Question author's solution:

```
import java.util.*;

public class Banco {
    // Atributos
    private String nombre;
    private ArrayList<Cuenta> lista;

    // Constructor
    public Banco(String nombre) {
        this.nombre = nombre;
        lista = new ArrayList<Cuenta>();
    }

    public void addCuenta(int numero, double saldo, String titular) {
        Cuenta unaCuenta = new Cuenta(numero, saldo, titular);
        lista.add(unaCuenta);
    }

    // toString
    public String toString() {
        String formato = nombre.toUpperCase() + "\n";
        for (Cuenta unaCuenta : lista) {
            formato += unaCuenta + "\n";
        }
        return formato;
    }

    // Buscar cuenta con saldo menor
    public String saldoMenor() {
        Cuenta min = lista.get(0);
        for (int i = 1; i < lista.size(); i++) {
            Cuenta unaCuenta = lista.get(i);
            if (unaCuenta.getSaldo() < min.getSaldo()) {
                min = unaCuenta;
            }
        }
        return min.toString();
    }
}
```

Incorrecta

Puntos para este envío: 0,00/1,00.

Pregunta 10

Incorrecta

Puntúa 0,00 sobre

Tenemos el siguiente código:

```
Banco unBanco = new Banco("banco");
unBanco.addCuenta(4320, 555.55, "Miren");
```

```

unBanco.addCuenta(1234, 1000.55, "Pedro");
unBanco.addCuenta(4321, 222.22, "Miren");
unBanco.addCuenta(2222, 4444.44, "Ana");
unBanco.addCuenta(4322, 1111.11, "Miren");
Banco cuentasTitular = unBanco.cuentasTitular("Miren");
System.out.println(cuentasTitular);

```

Crea el método cuentasTitular de la clase Banco para que devuelva un objeto también de la clase Banco con la información de las cuentas es el titular indicado como parámetro.

El resultado del código será:

```

CUENTAS DEL TITULAR: MIREN
Miren(4328): 555.55 euros
Miren(4321): 222.22 euros
Miren(4322): 1111.11 euros

```

Ten en cuenta que la clase Cuenta tiene los siguientes métodos:

```

public class Cuenta {
    public int numero;
    public double saldo;
    public String titular;

    public Cuenta(int numero, double saldo, String titular)
    public Cuenta(Cuenta unaCuenta)
    public int getNumero()
    public double getSaldo()
    public String getTitular()
    public String toString()
}

```

NOTA: Para ejecutar el programa puedes usar los siguientes ficheros. [Cuenta.java](#)

Respuesta: (penalty regime: 10, 20, ... %)

Reiniciar respuesta

```

1 import java.util.*;
2
3 + public class Banco {
4     // Atributos
5     private String nombre;
6     private ArrayList<Cuenta> lista;
7
8     // Constructor
9     public Banco(String nombre) {
10         this.nombre = nombre;
11         lista = new ArrayList<Cuenta>();
12     }
13
14     public void addCuenta(int numero, double saldo, String titular) {
15         Cuenta unaCuenta = new Cuenta(numero, saldo, titular);
16         lista.add(unaCuenta);
17     }
18
19     // toString
20     public String toString() {

```

Syntax Error(s)

```

__Tester__.java:46: error: cannot find symbol
Banco cuentasTitular = unBanco.cuentasTitular("Miren");
^
symbol:   method cuentasTitular(String)
location: variable unBanco of type Banco
1 error

```

Question author's solution:

```

import java.util.*;

public class Banco {
    // Atributos
    private String nombre;
    private ArrayList<Cuenta> lista;

    // Constructor
    public Banco(String nombre) {
        this.nombre = nombre;
        lista = new ArrayList<Cuenta>();
    }

    public void addCuenta(int numero, double saldo, String titular) {
        Cuenta unaCuenta = new Cuenta(numero, saldo, titular);
        lista.add(unaCuenta);
    }

    // toString
    public String toString() {
        String formato = nombre.toUpperCase() + "\n";
        for (Cuenta unaCuenta : lista) {
            formato += unaCuenta + "\n";
        }
        return formato;
    }

    // Buscar todas las cuentas de un mismo titular
    // Las devuelve como un objeto banco con nombre "Cuentas del titular: nombreTitular"
    public Banco cuentasTitular(String titular) {
        Banco bancoTitular = new Banco("Cuentas del titular: " + titular);
        for (Cuenta unaCuenta : lista) {
            String elTitular = unaCuenta.getTitular();
            if (elTitular.equals(titular)) {
                Cuenta copia = new Cuenta(unaCuenta);
                bancoTitular.lista.add(copia);
            }
        }
        return bancoTitular;
    }
}

```

Incorrecta

Puntos para este envío: 0.00/1.00.

Finalizar revisión

