

36019475. CSIFC03. MP0485. Programación

**XUNTA DE GALICIA**CONSELLERÍA DE CULTURA, EDUCACIÓN
E ORDENACIÓN UNIVERSITARIA

Páxina principal ► Os meus cursos ► Formación Profesional a Distancia ► Curso 2016-2017 ►
36019475 IES de Rodeira ► CSIFC03 Desenvolvemento de aplicacións web ►
125_36019475_CSIFC03_MP0485_A ► Unidade didáctica 9 ► Tarea para PROG09.

Tarea para PROG09.

Tarea para PROG09.

Detalles de la tarea de esta unidad.

Enunciado.

A lo largo de esta unidad has terminado de familiarizarte con el resto de conceptos relacionados con la **Programación Orientada a Objetos** que faltaban por ver de una manera más formal y con ejemplos explícitos: **composición; herencia; clases y métodos abstractos; sobrescritura de métodos; interfaces; polimorfismo; ligadura dinámica**, etc.

Has experimentando con todos estos conceptos y los has utilizado en pequeñas aplicaciones para comprobar su funcionamiento y su utilidad.

Una vez finalizada la unidad se puede decir que tienes ya un dominio adecuado del lenguaje Java como un lenguaje que permite aplicar todas las posibilidades de la **Programación Orientada a Objetos**. Dado ese supuesto, esta tarea tendrá como objetivo escribir una pequeña aplicación en Java empleando algunas de las construcciones que has aprendido a utilizar.

Se trata de desarrollar una aplicación Java que permita gestionar varios tipos de **cuentas bancarias**. Mediante un menú se podrán elegir determinadas operaciones:

1. **Abrir una nueva cuenta.**
2. **Ver un listado de las cuentas disponibles** (código de cuenta, titular y saldo actual).
3. **Obtener los datos de una cuenta concreta.**
4. **Realizar un ingreso en una cuenta.**
5. **Retirar efectivo de una cuenta.**
6. **Consultar el saldo actual de una cuenta.**
7. **Salir de la aplicación.**

Las cuentas se irán almacenando en alguna estructura en memoria según vayan siendo creadas. Cada cuenta será un objeto de una clase que contendrá la siguiente información:

- **Titular** de la cuenta (un objeto de la clase **Persona**, la cual contendrá información sobre el titular: **nombre, apellidos, fecha de nacimiento**).
- **Saldo** actual de la cuenta (número real).
- **Número de cuenta** (**CCC** - Código Cuenta Cliente).
- **Tipo de interés** anual (si se trata de una **cuenta de ahorro**).
- **Lista de entidades** autorizadas para cobrar recibos de la cuenta (si se trata de una **cuenta corriente**).
- **Comisión de mantenimiento** (para el caso de una **cuenta corriente personal**).
- **Tipo de interés por descubierto** (si es una **cuenta corriente de empresa**).
- **Máximo descubierto permitido** (si se trata de una **cuenta corriente de empresa**).

Las **cuentas bancarias** pueden ser de dos tipos: **cuentas de ahorro** o bien **cuentas corrientes**. Las **cuentas de ahorro** son **remuneradas** y tienen un determinado **tipo de interés**. Las **cuentas corrientes** no son remuneradas, pero tienen asociada una **lista de entidades autorizadas** para cobrar **recibos domiciliados** en la cuenta.

Dentro de las **cuentas corrientes** podemos encontrar a su vez otros dos tipos: las **cuentas corrientes personales**, que tienen una **comisión de mantenimiento** (una cantidad fija anual) y las **cuentas corrientes de empresa**, que no la tienen. Además, las **cuentas de empresa** permiten tener una cierta cantidad de **descubierto** (**máximo descubierto permitido**) y por tanto un **tipo de interés por descubierto** y una **comisión fija** por **cada descubierto** que se tenga. Es el único tipo de cuenta que permite tener **descubiertos**.

- **Datos personales:** nombre, apellidos, fecha de nacimiento.
- **Tipo de cuenta** que desea abrir: **cuenta de ahorro**, **cuenta corriente personal** o **cuenta corriente de empresa**.
- **Saldo inicial**.

Además de esa información, el **usuario de la aplicación** deberá también incluir:

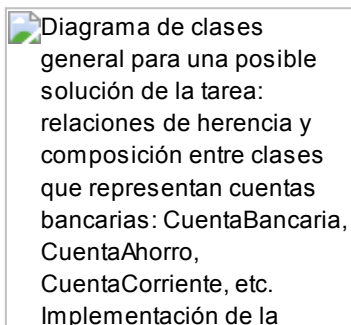
- **Número de cuenta (CCC)** de la nueva cuenta. Debe ser válido (habrá que comprobarlo).
- **Tipo de interés** de remuneración, si se trata de una **cuenta de ahorro**.
- **de mantenimiento**, si es una **cuenta corriente personal**.
- **Máximo descubierto permitido**, si se trata de una **cuenta corriente de empresa**.
- **Tipo de interés por descubierto**, en el caso de una **cuenta corriente de empresa**.
- **Comisión fija por cada descubierto**, también para el caso de una **cuenta corriente de empresa**.

La aplicación deberá asegurarse que la información introducida sea **válida y coherente** (CCC válido; saldos, comisiones y tipos de interés positivos, etc.).

El programa que escribas debe cumplir al menos los siguientes requisitos:

- Para almacenar los objetos de tipo cuenta podrás utilizar cualquier estructura de almacenamiento que consideres oportuna (`ArrayList`, `Hashtable`, etc.).
- Para trabajar con los datos personales, debes utilizar una clase **Persona** que contenga la información sobre los datos personales básicos del cliente (**nombre**, **apellidos**, **fecha de nacimiento**).
- Para trabajar con el número de cuenta debes utilizar el modelo de **Código Cuenta Cliente (CCC)**, que es posible que también la ya hayas usado en otras unidades.
- Para guardar las entidades autorizadas a cobrar recibos debes utilizar una `Hashtable` que contenga pares de tipo (código de entidad (`String`), máxima cantidad autorizada para un recibo).

Aquí tienes un ejemplo de una posible estructura de clases para llevar a cabo la aplicación:



El **código fuente** Java de **cada clase** debería incluir **comentarios** en cada **atributo** (o en cada conjunto de atributos) y **método** (o en cada conjunto de métodos del mismo tipo) indicando su **utilidad**. El **programa principal (clase principal)** también debería incluir algunos comentarios explicativos sobre su funcionamiento y la utilización de objetos de las distintas clases utilizadas.

Además del programa deberás escribir también un **informe** con todas las consideraciones oportunas que se necesiten para entender cómo has realizado la tarea.

El proyecto deberá contener al menos los siguientes archivos fuente Java:

- **Programa principal** (clase con método `main` : **AplicacionCuentaBancaria.java**).
- Un archivo por cada clase o interfaz que hayas implementado.

El documento que contendrá el informe lo elaborarás con un procesador de texto. Debe ser de tipo ".doc" (Microsoft Word) o de tipo ".odt" (OpenOffice.org). Debe tener tamaño de página A4, estilo de letra Times New Roman, tamaño 12 e interlineado normal.

Criterios de puntuación. Total 10 puntos.

Criterios de puntuación.

Existe una clase <code>CuentaBancaria</code> base que proporciona los atributos y métodos necesarios para cualquier tipo de cuenta bancaria genérica. La clase funciona correctamente.	2,00
Existe una clase <code>CuentaAhorro</code> que proporciona los atributos y métodos necesarios para trabajar con una cuenta de ahorro . La clase funciona correctamente.	2,00
Existe una clase <code>CuentaCorrientePersonal</code> que proporciona los atributos y métodos necesarios para trabajar con una cuenta de corriente personal . La clase funciona correctamente.	2,00
Existe una clase <code>CuentaCorrienteEmpresa</code> que proporciona los atributos y métodos necesarios para trabajar con una cuenta de corriente de empresa . La clase funciona correctamente.	2,00
Las clases de tipo cuenta corriente disponen de una colección que contiene la lista de entidades autorizadas para cobrar recibos en esa cuenta.	1,00
Se utiliza algún tipo de colección (<code>Hashtable</code> , <code>ArrayList</code> , etc.) para manipular las cuentas que se van creando durante la ejecución del programa.	1,00
La información de un titular no es almacenada en objetos de la clase Persona que a su vez son almacenados dentro de los objetos de cuenta bancaria.	-1,00
Las clases no son capaces de validar y gestionar correctamente un CCC .	-1,00
Los métodos de las clases no son capaces de lanzar excepciones si se produce alguna situación anómala.	-1,00
Se utiliza el polimorfismo y la ligadura dinámica para trabajar con las cuentas bancarias y funciona correctamente.	2,00
No se han incluido comentarios en las clases tal y como se ha pedido en el enunciado.	-1,00
No se han incluido comentarios apropiados en el programa principal describiendo el funcionamiento de éste.	-1,00
No se ha entregado el informe explicativo.	-2,00
El programa principal no es capaz de crear objetos de alguno de los tres tipos de cuentas bancarias solicitados (cuenta de ahorro, cuenta corriente personal o cuenta corriente de empresa).	-5,00
Alguna de las opciones de menú pedidas en el enunciado (menú del programa principal) no funciona correctamente.	-1,00 por cada opción
Total (máximo)	10,00
Dado que algunos criterios de puntuación son negativos, podría suceder que el balance final fuera negativo. En tal caso la puntuación final será simplemente de 0,00. Si el balance final es positivo y mayor que 10,00, la puntuación final quedará como 10,00.	

Recursos necesarios para realizar la Tarea.

- Ordenador personal.
- JDK y JRE de Java SE.
- Entorno de desarrollo NetBeans con las funcionalidades necesarias para desarrollar y emular aplicaciones Java.

Consejos y recomendaciones.

Para realizar la aplicación te sugerimos las siguientes recomendaciones:

- Básate en los diferentes ejemplos que has tenido que probar durante el estudio de esta unidad y de unidades anteriores. Algunos de ellos te podrán servir de mucha ayuda, así que aprovéchalos.
- Puedes obtener información sobre el funcionamiento del CCC y cómo calcular los dígitos de control del siguiente artículo de Wikipedia:
Wikipedia: Código Cuenta Cliente.
- Puedes generar cuentas bancarias válidas (o comprobarlas) para hacer pruebas en tu programa desde el siguiente enlace:
Generador/validador de cuentas bancarias.

Indicaciones de entrega.

Una vez que tengas terminados el programa (**carpeta de proyecto** Netbeans, incluyendo todos los archivos fuente de todos los recursos) y el **documento explicativo**, comprime ambos en un **único archivo comprimido**. El envío se realizará **a través de la plataforma** de la forma establecida para ello, y el archivo se nombrará siguiendo las siguientes pautas:

apellido1	apellido2	nombre	SI Gxx	Tarea
...

Asegúrate que el nombre no contenga la letra ñ, tildes ni caracteres especiales extraños. Así por ejemplo la alumna **Begoña Sánchez Mañas para la novena unidad del MP de PROG**, debería nombrar esta tarea como...

sanchez_manas_begona_PROG09_Tarea

Estado da entrega

Estado da entrega	Sen intentos
Estado das cualificacións	Sen cualificar
Data límite	Martes, 18 de Abril do 2017, 11:55
Tempo restante	A tarefa venceu fai: 69 días 7 horas

Engadir entrega

Facerlle cambios á súa entrega

NAVEGACIÓN

[Páxina principal](#)

- Amiña área persoal

Páxinas do sitio

O meu perfil

Curso actual

125 36019475 ZSIFC03 MP0485 A

Participantes

Distinciones

Xeral

Unidade didáctica 1

Unidade didáctica 2

Unidade didáctica 3

Unidade didáctica 4

Unidade didáctica 5

Unidade didáctica 6

Unidade didáctica 7

Unidade didáctica 8

Unidade didáctica 9

 Orientaciones para el alumnado. PROG09.

 Solución a la tarea para PROG09.

 PROG09 Guiada.- Utilización avanzada de clases.

🗨️ Actividades presenciales de la UD9 en la tutoría c...

 Foro para PROG09.

Mapa conceptual para PROG09.

PROG09 Completa.- Utilización avanzada de clases.

 Recursos complementarios UD09.

 Examen de la UD09. Tarefa 1 Solución Tarefa 1 Tarefa 2 Solución Tarefa 2 Tarefa3 Solución Tarefa 3 Tarefa4 Solución Tarefa 4 Tarefa5 Solucións Tarefa 5

Unidade didáctica 10

Unidade didáctica 11

Os meus cursos

ADMINISTRACIÓN

Administración do curso

Configuración do meu perfil

[Inicio](#)[Preguntas Frecuentes](#)[Accesibilidade](#)[Aviso](#)[Privacidade](#)[Contacto](#)

