

Package `cuentas`

# Class `CCuenta`

`java.lang.Object`  
`cuentas.CCuenta`

```
public class CCuenta
extends Object
```

Clase cuenta con atributos privados de la clase CCuenta de tipo String: nombre y cuenta de tipo double: saldo y tipo de interes Esta clase contiene métodos para operar con cuentas bancarias

## Constructor Summary

### Constructors

Constructor	Description
<code>CCuenta()</code>	Constructor vacío No recibe ningún parámetro
<code>CCuenta(String nom, String cue, double sal, double tipo)</code>	Método constructor con parámetros

## Method Summary

All Methods	Instance Methods	Concrete Methods
Modifier and Type	Method	Description
double	<code>estado()</code>	Método getter para obtener el estado de la cuenta
String	<code>getCuenta()</code>	Método Getter
String	<code>getNombre()</code>	Método Getter
double	<code>getSaldo()</code>	Método getter para obtener el saldo
double	<code>getTipoInterés()</code>	Método getter del tipoInterés
void	<code>ingresar(double cantidad)</code>	Método que tiene como parámetro cantidad de tipo double comprueba que la cantidad es menor a 0 y si se cumple lanza una excepción con un mensaje, finalmente settea el saldo sumandole al saldo actual la cantidad que se le pasa por parámetro
void	<code>retirar(double cantidad)</code>	Método que tiene como parámetro cantidad de tipo double Compueba que el dato introducido como argumento es

menor o igual a 0 si se cumple lanza una excepción con un mensaje si no se cumple, comprueba que el método estado es menor a la cantidad si se cumple lanza otra excepción con otro mensaje finalmente calcula el saldo, restando la cantidad que se le pasa por parámetro al saldo actual

void	<b>setCuenta</b> (String cuenta)	Método setter del atributo de tipo String cuenta
void	<b>setNombre</b> (String nombre)	Método setter del atributo de tipo String nombre
void	<b>setSaldo</b> (double saldo)	Método setter del atributo de tipo double saldo
void	<b>setTipoInterés</b> (double tipoInterés)	Método setter del atributo de tipo double tipoInterés

### Methods inherited from class java.lang.Object

clone , equals , finalize , getClass , hashCode , notify , notifyAll , toString , wait , wait , wait

## Constructor Details

### CCuenta

```
public CCuenta()
```

Constructor vacío No recibe ningún parámetro

### CCuenta

```
public CCuenta(String nom,
               String cue,
               double sal,
               double tipo)
```

Método constructor con parámetros

#### Parameters:

nom - nombre del usuario de la cuenta

cue - numero de cuenta

sal - saldo de la cuenta

tipo - tipo de interés de la cuenta bancaria

## Method Details

### estado

```
public double estado()
```

Método getter para obtener el estado de la cuenta

**Returns:**

saldo de la cuenta bancaria

### getNombre

```
public String getNombre()
```

Método Getter

**Returns:**

nombre de la persona titular de la cuenta bancaria

### setNombre

```
public void setNombre(String nombre)
```

Método setter del atributo de tipo String nombre

**Parameters:**

nombre - asigna el nombre de la cuenta bancaria

### getCuenta

```
public String getCuenta()
```

Método Getter

**Returns:**

numero de cuenta

### setCuenta

```
public void setCuenta(String cuenta)
```

Método setter del atributo de tipo String cuenta

**Parameters:**

cuenta - asigna numero de cuenta bancaria

### getSaldo

```
public double getSaldo()
```

Método getter para obtener el saldo

**Returns:**

saldo de la cuenta bancaria

### setSaldo

```
public void setSaldo(double saldo)
```

Método setter del atributo de tipo double saldo

**Parameters:**

saldo - asigna el saldo a la cuenta bancaria

### getTipoInterés

```
public double getTipoInterés()
```

Método getter del tipoInterés

**Returns:**

tipo de interés de la cuenta bancaria

### setTipoInterés

```
public void setTipoInterés(double tipoInterés)
```

Método setter del atributo de tipo double tipoInterés

**Parameters:**

tipoInterés - asigna el tipo de interés de la cuenta bancaria

### ingresar

```
public void ingresar(double cantidad)  
    throws Exception
```

Método que tiene como parámetro cantidad de tipo double comprueba que la cantidad es menor a o y si se cumple lanza una excepción con un mensaje, finalmente settea el saldo sumandole al saldo actual la cantidad que se le pasa por parámetro

**Parameters:**

cantidad - a ingresar o retirar en la cuenta bancaria

**Throws:**

**Exception** - que genera un mensaje para indicar que no se puede ingresar una cantidad negativa

**retirar**

```
public void retirar(double cantidad)
    throws Exception
```

Método que tiene como parámetro cantidad de tipo double Comprueba que el dato introducido como argumento es menor o igual a 0 si se cumple lanza una excepción con un mensaje si no se cumple, comprueba que el método estado es menor a la cantidad si se cumple lanza otra excepción con otro mensaje finalmente calcula el saldo, restando la cantidad que se le pasa por parámetro al saldo actual

**Parameters:**

cantidad - a ingresar o retirar de la cuenta bancaria

**Throws:**

**Exception** - lanza una excepción