

Ejercicio 1. Clase Estudiante.

Implementa una clase *Estudiante* teniendo en cuenta las siguientes consideraciones.

Clase *Estudiante*. Clase que modela un estudiante considerando su **nombre**, **nota media numérica real** y **número de notas** que se tienen en cuenta para calcular esa nota media. Implementa dicha clase utilizando programación orientada a objetos, es decir, escribiendo el código de la clase que modele el comportamiento del estudiante con el siguiente diseño:

Nombre de la clase: Estudiante

Atributos de la clase:

- String nombre Nombre del estudiante
- double notaMedia Nota media del estudiante
- int numNotas Número de notas del estudiante

Tendrás que implementar un método get para cada atributo (`getNombre`, `getNotaMedia`, `getNumNotas`).

Método constructor: `Estudiante (String nombre, double notaMedia, int numNotas)`. Este método instancia un objeto de la clase *Estudiante* e inicializa los atributos con los valores que se pasan como parámetros.

En caso de que el nombre esté en blanco, la nota media o el número de notas recibido sean menores que 0 lanza una excepción indicando “**Datos iniciales erróneos**”.

Otros métodos de la clase:

`boolean actualizarNotaMedia(double nuevaNota)`. Si la nota que se recibe como parámetro no es negativa, se considera la actualización y se devolverá **true**. Así, **se incrementará el número de notas** del estudiante y **se actualizará la nota media** con esta nueva nota que se recibe como parámetro. En otro caso, se devolverá **false**.

`char calificacionMedia()`.

Según el valor de la nota media, se devolverá alguno de los siguientes caracteres:

'A'	A partir de 9.
'B'	A partir de 8.
'C'	A partir de 7.
'D'	A partir de 6.
'E'	A partir de 5.
'F'	Menor de 5.

`String toString()` . Devuelve una cadena con el texto correspondiente al valor de los atributos, de modo que devolvería un `String`. como por ejemplo, supuesto un objeto con esos valores en sus atributos (la nota media se mostrará con al menos un decimal y un máximo de dos decimales usando como separador de la parte decimal el punto, o sea '.', en lugar de la coma española):

Ejemplo:

“Estudiante: Francisco Cucharas Gómez, nota media: 2.0, número de notas: 1”

Método main de prueba de la clase

Crea también un método `main()` para probar la clase realizando las siguientes acciones:

- 1 Un primer intento de crear un objeto con datos erróneos para que salte la excepción y sea capturada como ya sabemos con una estructura *try-catch*.
- 2 Crear dos objetos *Estudiante* correctos y mostrar su estado usando el método *toString*. Usar los datos indicados en el ejemplo:
 - *Antonio Bernal Campillo*, nota media: 0.0, número de notas: 0.
 - *Francisco Cucharas Gómez*, nota media: 2.0, número de notas: 1.
- 3 Al estudiante que has creado con el nombre *Antonio Bernal Campillo*, **actualiza su nota media con una nueva calificación de 7.7**. Muestra de nuevo por consola su información usando el método *toString*.
- 4 Al estudiante que has creado con el nombre *Francisco Cucharas Gómez*, **actualiza su nota media con una nueva calificación de 7.5**. Muestra de nuevo por consola su información usando el método *toString*.
- 5 Finalmente, **muestra por pantalla la calificación media de Antonio Bernal Campillo**.

Aquí tienes como debería quedar la salida de ejecución del programa de prueba una vez implementada la clase solicitada:

```
run:
Intento crear estudiante de nombre Ricardo
Error al intentar crear: Datos iniciales erróneos

Vamos a crear dos objetos estudiante
Mostramos la información de los dos objetos estudiante
Estudiante: Antonio Bernal Campillo, nota media: 0.0, número de notas: 0
Estudiante: Francisco Cucharas Gómez, nota media: 2.0, número de notas: 1

Actualizamos la nota de Antonio con una nueva nota de 7.7
Estudiante: Antonio Bernal Campillo, nota media: 7.7, número de notas: 1

Actualizamos la nota de Francisco con una nueva nota de 7.5
Estudiante: Francisco Cucharas Gómez, nota media: 4.75, número de notas: 2

Mostramos la calificación media de Antonio
La calificación media de: Antonio Bernal Campillo es C
BUILD SUCCESSFUL (total time: 0 seconds)
```