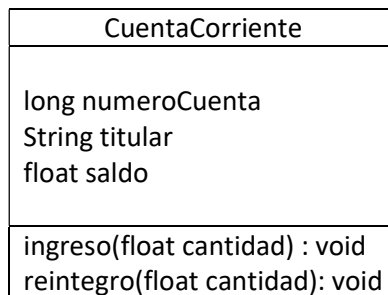


Ejercicios de Excepciones

1. Vamos a crear una clase CuentaCorriente que simule de una manera simple el funcionamiento de una cuenta corriente bancaria.

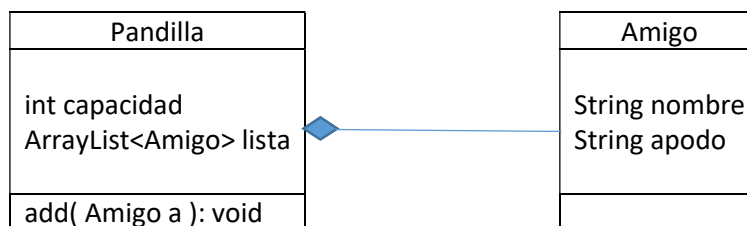
La clase tendrá el siguiente aspecto:



El método ingreso(), añade la cantidad al saldo, y el método reintegro() resta la cantidad al saldo, salvo en el caso en el que el saldo vaya a quedar en negativo, en ese caso el método reintegro() arrojará una excepción llamada **SaldoInsuficienteException**

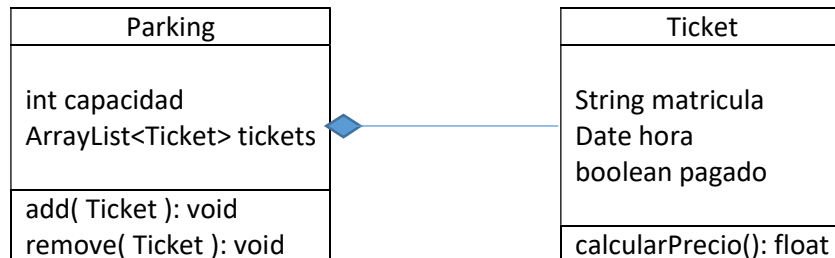
2. Usando la clase Contador realizada en las pasadas semanas, vamos a darle una nueva funcionalidad a los métodos incrementar() y decrementar(), haciendo que estos arrojen excepciones en caso de que la operación de incremento o decremento no se puedan realizar por encontrarse el Contador en sus límites inferior o superior.
3. Deseamos crear una clase que simule nuestro grupo de amigos. Para ellos vamos a programar una clase Pandilla que almacene todos los amigos que la formamos.

El diagrama UML sería:



El método add() debe retornar una excepción llamada **PandillaCompletaException** en caso de que la capacidad de la pandilla esté completa. Y además como somos una pandilla un poco especial, cuando alguien con un nombre terminado en 's' quiere entrar, lo rechazamos; por lo que el método add() debe arrojar la excepción **NoMeGustaTuNombreException** cuando se le pide añadir un amigo con nombre acabado en 's'.

4. Por ultimo vamos a informatizar la gestión de un Parking, de manera que podamos manejar los tickets que se asignan a cada Vehículo a la entrada o salida.
Las clases que vamos a crear son:



Los métodos de las dos clases arrojarán la excepción **TicketException** cuando se produzca alguna de las siguientes situaciones:

- Metodo add() arroja una excepción si el Parking está lleno, o si ya existe en el Parking otro ticket con la misma matricula.
- Metodo remove() arroja excepciones cuando queremos eliminar un ticket que no ha sido pagado.
- Metodo calcularPrecio(), calcula el coste que se debe abonar en este instante averiguando la diferencia entre la hora de entrada y la hora actual. Si por algún motivo la diferencia saliese negativa el método arrojaría una excepción.

Para calcular el precio del Ticket, usaremos los siguientes precios:

- Precio por minuto: 0,03€
- Precio por hora: 1,70€
- Precio por día: 17€ (si alguien está más de 10 horas, se le aplica el precio de día completo).
-