

## Hoja de ejercicios 60\_62:

- Número variable de parámetros a un método.
- Herencia.
- Polimorfismo. *instanceof*. *Downcasting* / *Upcasting*
- `String.split(String token);`

60. Crea una clase llamada *Calcula*:

- Implementa los siguientes métodos estáticos, que devolverán en cada caso el menor o mayor número de los pasados como parámetros. En el caso de las cadenas devolverá la de mayor o menor longitud.
  - `int mayor(int ... num )`
  - `double mayor(float ... num)`
  - `String mayor(String ... cadena)`
  - `int menor(int ... num )`
  - `double menor(double ... num)`
  - `String menor(String ... cadena)`
- Crea una clase aparte con el método *main* para probar todas las funcionalidades de la clase *Calcula*.

61. Crea una **subclase** llamada *Programador*, que herede de la clase *Empleado*, creada en el ejercicio 55. Añadirá el siguiente atributo privado:

- `String[] lenguajes`. Almacenará los lenguajes de programación que conoce.
- El constructor será el siguiente:
  - `Programador (String nombre, String telefono, String ... lenguajes)`
- Implementa un método público para recuperar los lenguajes de programación:
  - `String[] .getLenguajes()`
- Crea una clase aparte con el método *main* que rellene un array con objetos que podrán ser instancias de la clase *Empleado* o de la clase *Programador*.

62. Crea una **subclase** llamada *Conductor*, que herede de la clase *Empleado*, creada en el ejercicio 55.

- Tendrá los siguientes **atributos**:
  - `double totalKilometros`. Guardará el total de kilómetros recorridos.
  - `String ciudadesRecorridas`. Separadas por una coma, almacena los nombres de las ciudades por las que pasa.
- Y los siguientes **métodos**:
  - `void sumarKilometros (double kilometros)` Suma *kilometros* a *totalKilometros*.
  - `double getTotalKilometros()`
  - `void pasaPorCiudad (String ciudad)` Añade el nombre de la ciudad a las ciudades recorridas, permitiendo ciudades repetidas.
  - `String[] .getCiudadesRecorridas ( )` Devuelve un **array** con los nombres de las ciudades recorridas.
- Al **constructor** de la clase hay que pasarle el nombre y el teléfono del conductor. *totalKilometros* y *ciudadesRecorridas* son inicializados a 0 y "" respectivamente.
- Crea una clase aparte con el método *main* para probar todas las funcionalidades de la subclase *Conductor*.