



Clase Agenda y Contacto		
1º DAW - PRO		
Departamento de Informática	13/02/2025	Página 1 de 2

## CLASES AGENDA Y CONTACTOS

Vamos a crear una serie de programas que nos permitan trabajar sobre un objeto **Agenda** en el que podremos ir añadiendo, eliminando, buscando objetos de la clase **Contacto**. Ambas clases serán desarrolladas a partir de cero con los siguientes requisitos expuestos más adelante. Además, se desarrollará una clase principal (**Main\_Agenda**) que permitirá probar los métodos e instancias de dichas clases. Exponemos a continuación los requisitos de cada clase.

### Clase Main\_Agenda

- Crear instancias de **Contacto** con diferentes valores.
- Inicializar un array de **Contacto** y utilizarlo para crear una instancia de **Agenda**.
- Añadir contactos adicionales a la agenda utilizando el método `append()`.
- Imprimir la agenda en la consola utilizando el método `toString()`.

### Clase Contacto

Debes implementar una clase llamada **Contacto** que represente la información básica de un contacto. La clase debe contener los siguientes atributos privados:

- `String nombre`
- `String apellidos`
- `String dni`
- `String telefono`
- `int edad`

La clase **Contacto** debe incluir los siguientes elementos:

1. **Constructores:**
  - Un constructor que inicialice todos los atributos.
  - Un constructor que inicialice `nombre`, `apellidos`, `dni` y `edad`, y asigne el valor "No definido" al atributo `telefono`.
2. **Métodos de acceso (getters) y modificación (setters):**
  - `getNombre` y `setNombre`
  - `getApellidos` y `setApellidos`
  - `getTelefono` y `setTelefono`
  - `getEdad`
3. **Método `toString()`:**
  - Sobrescribe el método `toString()` para devolver una cadena de texto con la información del contacto en el siguiente formato para el contacto Juan:  
`nombre apellidos - DNI: dni - Teléfono: telefono - Edad: edad.`



Clase Agenda y Contacto		
1º DAW - PRO		
Departamento de Informática	13/02/2025	Página 2 de 2

## Clase Agenda

### tributos:

- `Contacto[] contacto`
- `int[] socio`

### Constructores:

- Un constructor sin parámetros que inicialice una agenda vacía.
- Un constructor que reciba un array de `Contacto` y lo utilice para inicializar la agenda. También deberá inicializar el array `socio` con números consecutivos empezando por 1.

### Métodos:

- `void append(Contacto nuevoContacto)`: Método que añada un nuevo contacto a la agenda.
- Sobrescribir el método `toString()` para devolver una lista de contactos alineada por columnas con la información de cada contacto en el siguiente formato: `ID: nombre apellidos - DNI: dni - Teléfono: telefono - Edad: edad`.