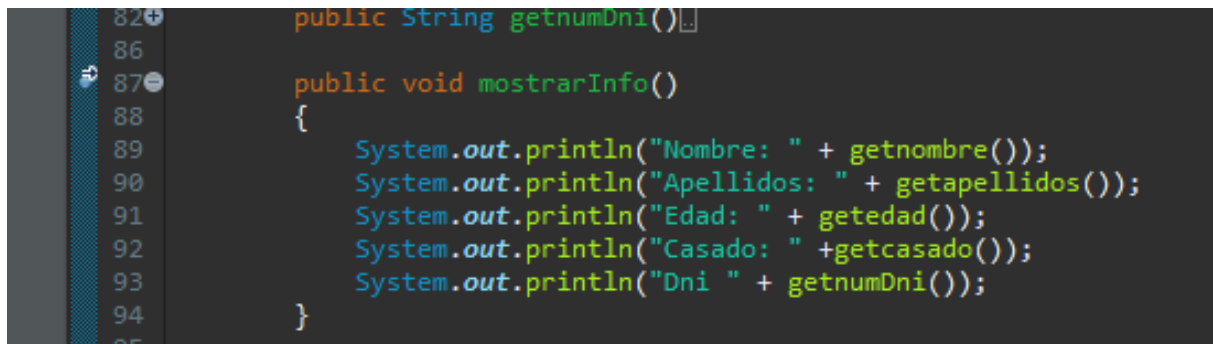


En Eclipse , realizaremos un proyecto con una clase y varios métodos. Por ejemplo, la clase persona con atributos tipo: nombre,apellidos, edad,casado y numDNI. Se tendrán que crear todos los getters and setters necesarios, junto a los correspondientes constructores. Compilaremos y ejecutaremos la prácticas desde el IDE. Además depuraremos el proyecto ejecutando las líneas paso a paso, añadiremos las variables que consideremos importantes a los watches, y definiremos los siguientes breakpoints:

🔍 Tipo línea, por ejemplo al pasar por una salida de pantalla.



```
82+ public String getnumDni()
86
87- public void mostrarInfo()
88 {
89     System.out.println("Nombre: " + getnombre());
90     System.out.println("Apellidos: " + getapellidos());
91     System.out.println("Edad: " + getedad());
92     System.out.println("Casado: " + getcasado());
93     System.out.println("Dni " + getnumDni());
94 }
95
```

En este breakpoint se marca la petición de impresión.

🔍 Tipo clase, al cargar alguna clase.



```
4 {
5- public static void main(String[] args)
6 {
7     Persona p1 = new Persona();
8     p1.mostrarInfo();
9     p1.setcasado(false);
10    p1.mostrarInfo();
11 }

//Constructor
public Persona()
{
    nombre = "Alvaro";
    apellidos = "Canyizares";
    edad = 1;
    casado = true;
    numDni = "00000000a";
}
```

Desde el main se cargan los valores por defecto de Persona. En la primera captura se puede observar como se crea una clase nueva (p1), a continuación, el programa lee los valores iniciales, definidos en el constructor, para que sean sus valores por defecto.

🔗 Tipo campo, al acceder a algún campo de alguna clase.

```
7      Persona p1 = new Persona();  
8      p1.mostrarInfo();  
9      p1.setcasado(false);  
10     p1.mostrarInfo();  
11 }
```

Se observa como, tras definir el método set para el campo “casado”, se accede a él para modificar su valor en la entidad p1 a false.

🔗 Tipo método, al acceder a algún método de alguna clase.

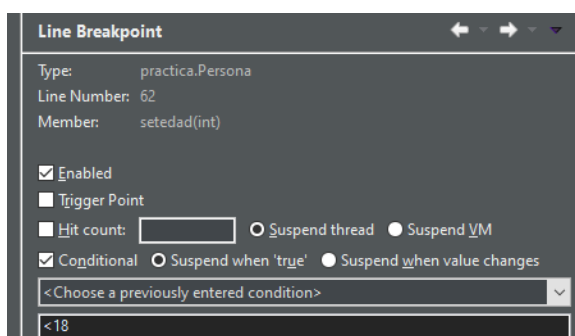
```
Persona p1 = new Persona();  
p1.mostrarInfo();  
p1.setcasado(false);  
p1.mostrarInfo();  
  
public void mostrarInfo()  
{  
    System.out.println("Nombre: " + getnombre());  
    System.out.println("Apellidos: " + getapellidos());  
    System.out.println("Edad: " + getedad());  
    System.out.println("Casado: " + getcasado());  
    System.out.println("Dni " + getnumDni());  
}
```

En la primera foto se observa como se accede al método mostrarInfo de la entidad p1. En la segunda se describe las funciones de este, concretamente la impresión de los valores de los atributos de la entidad p1 de la clase Persona.

En al menos alguno, en las propiedades de los breakpoints definiremos:

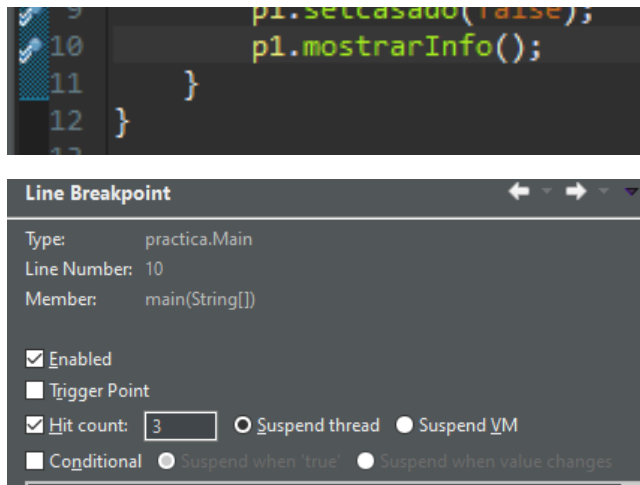
🔗 Condiciones para que el programa se detenga al alcanzar las variables ciertos valores

```
60  
61  
62     edad = _edad;  
63 }
```



En la primera imagen se observa como el breakpoint de la línea 62, respectiva al setter de edad, se encuentra condicionado a que el programa pare si el valor es menor a 18.

☑ Se alcance un número de veces (hints), determinado.



Se ha colocado en el breakpoint la limitación de que solo se pueda ejecutar mostrarInfo para p1, 3 veces.

Código:

Main:

```
package practica;  
  
public class Main  
{  
    public static void main(String[] args)  
    {  
        Persona p1 = new Persona();  
        p1.mostrarInfo();  
        p1.setcasado(false);  
        p1.mostrarInfo();  
    }  
}
```

Persona:

```
public class Persona
{
    String nombre;
    String apellidos;
    int edad;
    boolean casado;
    String numDni;

    public Persona()
    {
        nombre = "Alvaro";
        apellidos = "Canyizares";
        edad = 18;
        casado = true;
        numDni = "00000000a";
    }

    public void setnom(String _nombre)
    {
        nombre = _nombre;
    }

    public String getnombre()
    {
        return nombre;
    }

    public void setapellidos(String _apellidos)
    {
        apellidos = _apellidos;
    }
}
```

```
public String getapellidos()
{
    return apellidos;
}

public void setedad(int _edad)
{
    edad = _edad;
}

public int getedad()
{
    return edad;
}

public void setcasado(boolean _casado)
{
    casado = _casado;
}

public boolean getcasado()
{
    return casado;
}

public void setnumDni(String _numDni)
{
    numDni = _numDni;
}

public String getnumDni()
{
    return numDni;
}
```

```
        public void mostrarInfo()
        {
            System.out.println("Nombre: " + getnombre());
            System.out.println("Apellidos: " + getapellidos());
            System.out.println("Edad: " + getedad());
            System.out.println("Casado: " + getcasado());
            System.out.println("Dni " + getnumDni());
        }
    }
```