**Metodos get y set**

Los métodos **get** y **set**, son simples métodos que usamos en las clases para mostrar (**get**) o modificar (**set**) el valor de un atributo. El nombre del método como sugerencia para una rapida identificacion del mismo dentro del codigo sera **get** o **set** y a continuación el nombre del atributo, su modificador siempre es**public** ya que queremos mostrar o modificar desde fuera la clase. Por ejemplo, **getNombre** o **setNombre**.

**Codigo Ejemplo metodo get**

public tipo\_dato\_atributo getAtributo (){

return atributo;

}

**Codigo Ejemplo metodo set**

public void setAtributo (tipo\_dato\_atributo variable){

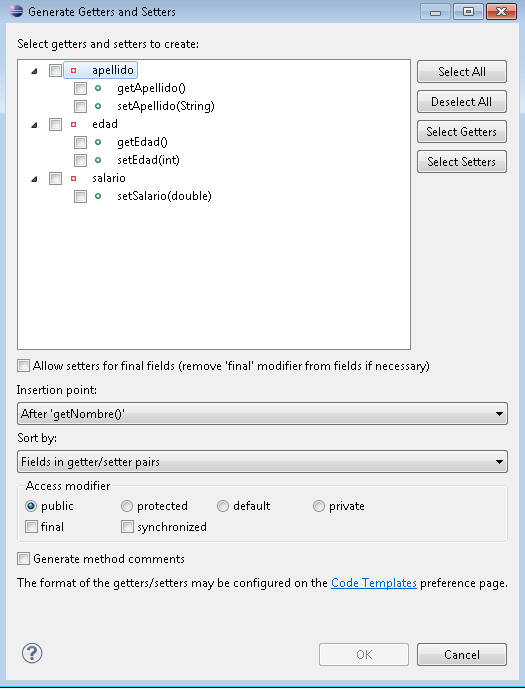
this.atributo = variable;

}

Si usamos Eclipse tenemos una manera aún mas sencilla de generarlos automáticamente.

La ruta es: **Source -> Generate Getters and Setters.**

Nos aparecerá una ventana donde elegiremos que atributos generar su get o set, al marcar en **OK** nos lo generara automáticamente estos métodos.



Vamos a otro ejemplo:

Ya vimos como crear una clase persona y además, creamos objetos de tipo persona desde el método main tanto con el [constructor por defecto](https://www.clasesdeinformaticaweb.com/java-desde-cero/constructores-en-java/), como con constructores personalizados.

package objetos;

public class Principal {

public static void main(String[] args) {

/\* Creamos un objeto persona llamado nuevaPersona con el constructor pasandole 3 parámetros El DNI, el nombre y la edad\*/

Persona nuevaPersona =new Persona("12345678","Fran",35);

}

}

Como se puede ver, se usa el constructor para Persona donde se envia los parametros:

**("12345678","Fran",35);**

De haber usado un constructor por defecto, no se le hubiera pasado ningun parametro al momento de crear al objeto.

Ahora, les mostramos un ejemplo **INCORRECTO** de como cargar los valores de los atributos de un objeto.

package objetos;

public class Principal {

public static void main(String[] args) {

/\* Creamos un objeto persona llamado nuevaPersona con el constructor pasandole 3 parámetros El DNI, el nombre y la edad\*/

Persona nuevaPersona =new Persona();

nuevaPersona.dni="12345678";

nuevaPersona.nombre="Fran";

nuevaPersona.edad=35;

}

}

El error se debe a que al tener las variables dni, nombre y edad declaradas como privadas, el propio entorno de desarrollo, nos daría un error. Por este motivo, para ello debemos utilizar los métodos get y set de java.

**Que es y como crear setters. Set en java.**

El [método](https://www.clasesdeinformaticaweb.com/java-desde-cero/metodos-o-funciones-en-java/) set es un **método público**, el cual se encarga de darle un valor a una propiedad o atributo de un objeto, el cual está encapsulado en la clase correspondiente, es decir, esta declarado con la palabra reservada **private o protected.**

Para declarar un método setter, lo haremos dentro de la clase persona y a continuación de los constructores de la siguiente manera. En este ejemplo declararemos los métodos set de las tres características del objeto persona.

**/**\* Construimos el método set para establecer un valor al DNI de la persona. Este método recibe un parámetro pDni, que será lo que el usuario introduzca por teclado.\*/

public void setDni(String pDni) {

this.dni = pDni;

}

/\* Construimos el método set para establecer un valor al nombre de la persona. Este método recibe un parámetro pNombre, que será lo que el usuario introduzca por teclado.\*/

public void setNombre(String pNombre) {

this.nombre = pNombre;

}

/\* Construimos el método set para establecer un valor a la edad de la persona. Este método recibe un parámetro pEdad, que será lo que el usuario introduzca por teclado.\*/

public void setEdad(int pEdad) {

this.edad = pEdad;

}

}

### Que es y como crear getters. Get en java.

El [método](https://www.clasesdeinformaticaweb.com/java-desde-cero/metodos-o-funciones-en-java/) get al igual que el set, es un **método público**, pero el get se encarga de mostrar un valor a una propiedad o atributo de un objeto, el cual está encapsulado en la clase correspondiente, es decir, esta declarado con la palabra reservada **private o protected.**

Para declarar un método getter, lo haremos también dentro de la clase persona y a continuación de los métodos setter declarados anteriormente y de la siguiente manera. En este ejemplo declararemos los métodos get de las tres características del objeto persona.

/\* Este método get, devuelve un valor del tipo String, ya que el dni se ha declarado como String y está devolviendo el valor que posee el atributo de la clase llamado dni que es de tipo String\*/

public String getDni() {

return dni;

}

/\* Este método get, devuelve un valor del tipo String, ya que el nombre se ha declarado como String y está devolviendo el valor que posee el atributo de la clase llamado nombre que es de tipo String\*/

public String getNombre() {

return nombre;

}

/\* Este método get, devuelve un valor del tipo entero (int), ya que el dni se ha declarado como entero (int) y está devolviendo el valor que posee el atributo de la clase llamado edad que es de tipo String\*/

public int getEdad() {

return edad;

}

}

Ahora una vez que ya tenemos declarados los métodos get y set, podemos establecer valores a los nuevos objetos, y mostrarlos por pantalla.

### Establecer valores a un objeto (set en java) y mostrarlo desde el método main (get en java).

package objetos;

//Importamos el paquete donde tenemos la clase Scanner

import java.util.\*;

public class Principal {

public static void main(String[] args) {

/\* Ahora declaramos tres variablesy las iniciamos\*/

String dniPersona="";

String nombrePersona="";

int edadPersona=0;

/\* Declaramos un objeto de tipo Scanner\*/

Scanner lectura=new Scanner(System.in);

/\* Pedimos el DNI. Lo introducimos dentro de la variable dniPersona y dentro de un bucle while para que se repita la pregunta si la longitud no es correcta.\*/

do {

System.out.println("Introduce un DNI:");

dniPersona=lectura.nextLine();

}while(dniPersona.equalsIgnoreCase("") || dniPersona.length()<8);

/\* Pedimos el nombre. Lo introducimos dentro de la variabla nombrePersona y dentro de un bucle while para que se repita la pregunta si el nombre queda vacio\*/

do {

System.out.println("Introduce un nombre:");

nombrePersona=lectura.nextLine();

}while(nombrePersona.equalsIgnoreCase(""));

/\* Pedimos la edad. La introducimos dentro de la variabla edadPersona y dentro de un bucle while para que se repita la pregunta si el edad es cero\*/

do {

System.out.println("Introduce una edad:");

edadPersona=lectura.nextInt();

}while(edadPersona==0);

/\* Creamos una variable objeto de tipo persona y le pasamos las tres variables. \*/

Persona nuevaPersona=new Persona(dniPersona,nombrePersona,edadPersona);

/\* El objeto nuevaPersona tiene ahora métodos set para asignar un valor a cada característica y get para mostrar el valor de cada característica. Los valores que queremos introducir están en las variables dniPersona, nombrePersona y edadPersona así que vamos a asignar valores al dni, al nombre y a la edad, a través de los métodos correspondientes\*/

nuevaPersona.setDni(dniPersona);

nuevaPersona.setNombre(nombrePersona);

nuevaPersona.setEdad(edadPersona);

// Mostramos los datos de la persona por pantalla

System.out.println("El DNI es: " + nuevaPersona.getDni());

System.out.println("El nombre es: " + nuevaPersona.getNombre());

System.out.println("La edad es: " + nuevaPersona.getEdad());

}

}