

# ¿Qué es encapsular?

**DigitalHouse** >  
Coding School



**Certified Tech  
Developer**  
The Ultimate Degree



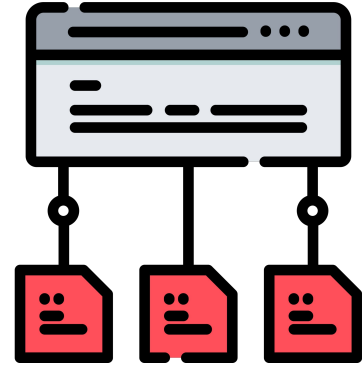
El **encapsulamiento** es una de las propiedades más importante de la programación orientada a objetos.



# Encapsulamiento

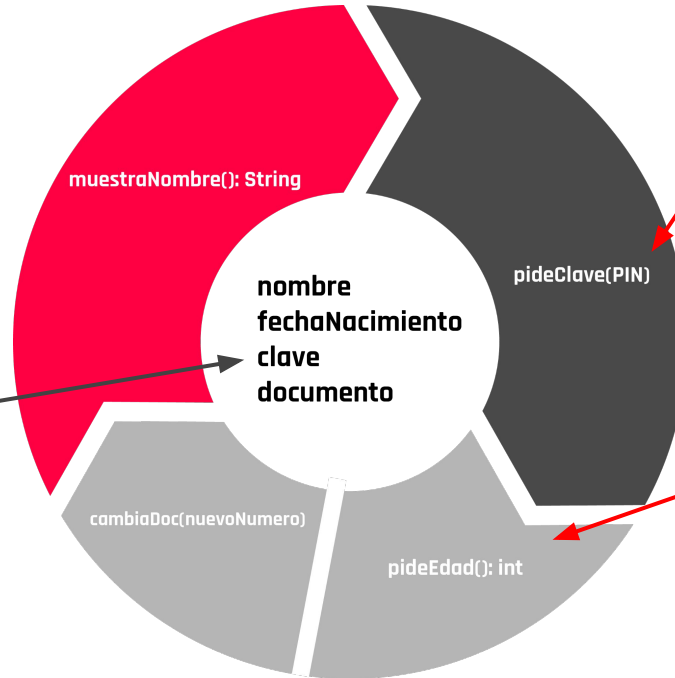
Cuando hablamos de encapsulamiento, no debemos olvidarnos del origen de la palabra “colocar en cápsulas”. ¿Qué serían las cápsulas? Es un envoltorio que protege el contenido en su interior. En la POO, buscamos impedir que cualquier otro objeto pueda tener acceso a la estructura interna de un objeto. Solamente **yo** puedo cambiar o mostrar mi estado y con los métodos específicos que van a indicar **cómo** pedir cambios en dichos atributos desde el exterior del objeto.

De ahora en más, cuando diseñamos nuestros objetos, tenemos que tener en cuenta el encapsulamiento. Por ejemplo, si tuviéramos un objeto Persona, que tiene como atributo su clave de acceso bancario, no sería conveniente que todos los objetos puedan acceder libremente a dicho atributo. El objeto Persona debería establecer un método controlado y seguro para devolver la clave de acceso bancario, por ejemplo, si es que se cumplen medidas de seguridad.



# Persona

Nadie puede ver o modificar directamente la información que forma la estructura interna del objeto, solamente el propio objeto.



Para cambiar la clave le pido que lo haga (dando el PIN).

Para saber la edad le pido que me la devuelva (no se cómo la calculará).

# Importante

- Cuando definamos un objeto, dejar sus atributos privados.
- Los métodos que sean públicos serán vistos por los otros objetos.
- Usar siempre métodos públicos para ver o modificar las características de tus objetos.
- Para cambiar el valor de un atributo se usa un método **set**, por ejemplo, para cambiar el nombre será **setNombre(String)**
- Para obtener el valor de un atributo se usa un método **get**, por ejemplo, para saber el nombre será **getNombre(): String**



Los métodos para ver o cambiar atributos se los denomina **getters** y **setters** respectivamente.

DigitalHouse>  
Coding School