

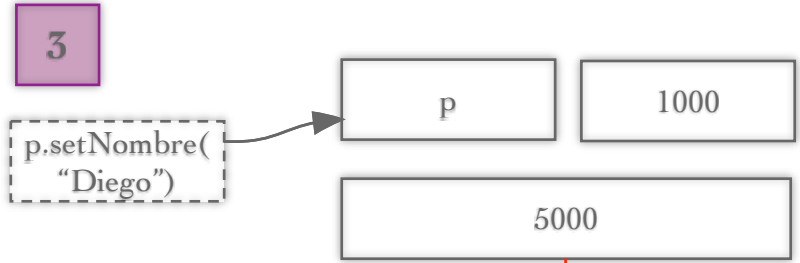
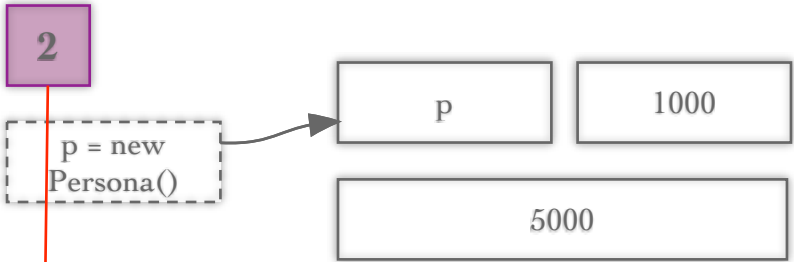
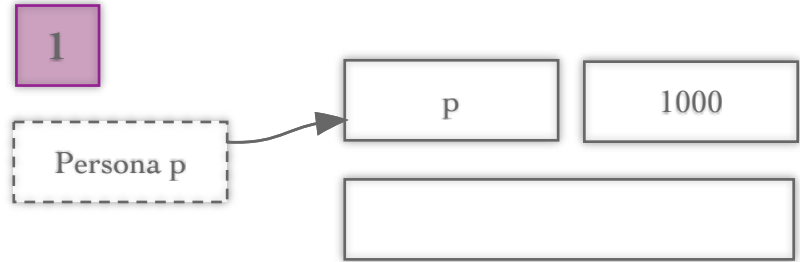
OBJETOS

Se va a analizar el paso por parámetros. Se produce por VALOR.

PARÁMETROS ( CLASE: ObjetosParámetros.java )

```
1 -- Persona p;  
2 -- p = new Persona();  
3 -- p.setNombre("Diego");
```

```
p.getNombre() = Diego
```



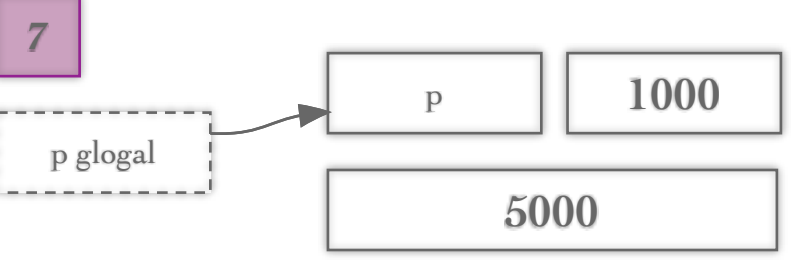
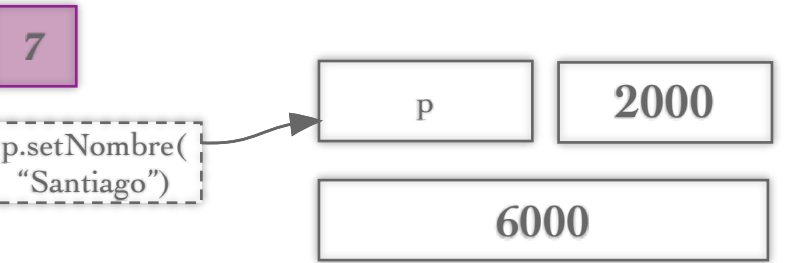
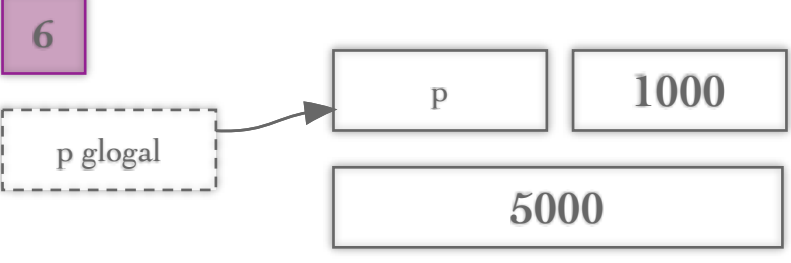
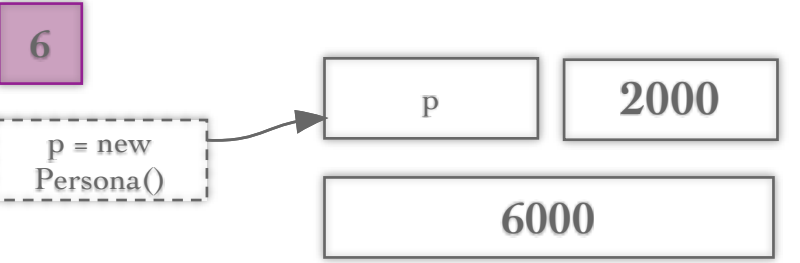
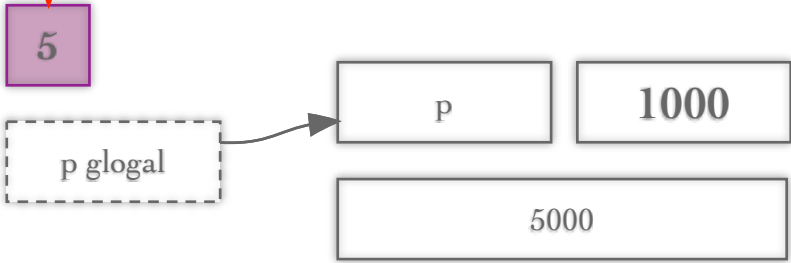
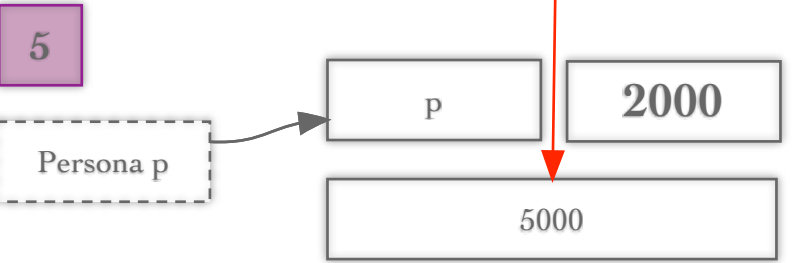
Definimos o declaramos una variable de tipo puntero que va a recibir una dirección de memoria que corresponde a un objeto de la clase persona, a la que apuntará, y posteriormente le asignamos un nombre al atributo nombre del objeto.

```
4 -- cambiarNombre1(p);
```

Se envía el objeto de la clase Persona al método cambiarNombre1. Realmente lo que hará será una copia del valor que guarda p, es decir, una copia de la dirección de memoria que es el valor que guarda p.

```
5 -- private static void cambiarNombre1(Persona p) {  
6 --     p = new Persona();  
7 --     p.setNombre("Santiago");  
8 -- }
```

```
p.getNombre() = Diego -p global-  
p.getNombre() = Santiago -p local-
```



Se crea una variable p local que recibe como valor una copia del valor de p global, es decir, recibe una copia de la dirección de memoria de p global con lo que se tiene dos variables que se llaman igual pero con direcciones de memoria diferente y que almacenan como valor la misma dirección de memoria que corresponde a un objeto de la clase Persona. En el método se crea un nuevo objeto de la clase Persona y se guarda en la variable p local, por lo tanto, ahora se tiene dos variables que se llaman igual con dos direcciones de memoria diferente y que almacenan dos direcciones de memoria diferente, es decir, dos objetos de la clase Persona diferente.