**Convenciones Clases e Interfaces.**

Las clases

* La primer letra debe ser mayúscula
* Utiliza nomenclatura camelCase
* Para las clases, los nombres deben de ser sustantivos (Sujeto) y van después de la palabra reservada **class**
* Para las interfaces, los nombres deben de ser adjetivos (Califica el sustantivo) y van después de la palabra reservada **interface**

**Ej**: class Persona  
class  ClasePrincipal  
class  VentanaRegistro  
interface ActionListener  
interface MouseInputListener

**Convenciones en Paquetes.**

Los paquetes.

* Deben ser escritos todo en minúscula.
* Van después de la palabra reservada **package**
* Si se van a usar paquetes dentro de otros paquetes, se unen mediante un punto (.)
* Finalizan con ;

**Ej**:  package ventanas;  
package vo;  
package dao;   
package imagenes.iconos;

**Convenciones en Métodos.**

Los métodos

* La primer letra debe ser minúscula
* Utiliza nomenclatura camelCase
* Los nombres deben conformarse por el par verbo + sustantivo
* el nombre va después del tipo de método (void, int, double, String)
* al finalizar el nombre del método debe indicarse mediante paréntesis con o sin argumentos ()

**Ej**:  void miMetodo()  
int sumaEnteros(int a, int b)    
Sting mensaje(String saludo)   
boolean retornaPermisos(int tipoUsuario)

**Convenciones en Variables.**

Las variables

* La primer letra debe ser minúscula
* Utiliza nomenclatura camelCase
* el nombre va después del tipo de dato (int, String, double, boolean)
* Es recomendable utilizar nombres con un significado explícito, y en lo posible, cortos

**Ej**:  int edad  
String nombre  
String direccionResidencia  
boolean resultadoPrueba

**Convenciones en Constantes.**

Las constantes

* Todas las letras de cada palabra deben estar en mayúsculas
* Se separa cada palabra con un \_
* se declaran similar a las variables, con la diferencia de que el tipo de dato va después de la palabra reservada **final**.

**Ej**:  final int EDAD

final String CODIGO\_CIUDAD  
final double PI