



**UNIVERSIDAD  
INTERAMERICANA PARA EL  
DESARROLLO**

**CAMPUS CANCÚN**

**“Modificadores de acceso”**

**“JSP, JSF JPA”**

**DESARROLLO ORIENTADO A OBJETOS**

**ELABORADO POR:  
MARIO DAVID MORALES CASTRO**

**LIC. INGENIERÍA EN TECNOLOGÍAS DE  
INFORMACIÓN Y COMUNICACIÓN**

**ENERO 2019**

## Modificadores de acceso: Public, Private, Protected

---

En java, el término de modificadores de acceso hace referencia a las palabras claves que se encargan de controlar la accesibilidad a clases, variables, campos, constructores y métodos. Eso significa que nosotros podemos determinar que puede acceder a nuestro código. Por ejemplo si queremos que nuestra clase "perro" pueda ser por accedida por cualquiera, usamos el modificador "public".

Las Variables definidas dentro de una clase son llamadas variables de instancias porque cada instancia de una clase (eso es, cada objeto de la clase) contiene su propia copia de aquella variable. Las variables de instancias. Pueden ser declaradas públicas, privadas, protegidas o default (sin modificador).

**Public:** Los miembros (variables, métodos y constructores) declarados public o públicos dentro de una clase publica son visibles para cualquier otra clase dentro del programa en java, ya sea que estas clases estén en otro o el mismo paquete.

**Protected:** Los campos o métodos protegidos no pueden ser usados por otras clases. Interfaces, campos, métodos y constructores declarados protegidos en una superclase pueden ser accedidos solo por sus subclases. Clases en el mismo paquete también pueden acceder a los campos, métodos y constructores protegidos, incluso si no son una subclase del miembro de la clase protegida.

**Default:** Este nivel de acceso es declarado cuando no se escribe ningún modificador de acceso. Cualquier campo, método o constructor que no tenga declarado un modificador de acceso es accesible solo por las clases dentro del mismo paquete en el que se encuentra.

**Private:** los modificadores privados (más restrictivos) pueden ser usados por miembros pero no pueden ser usados por clases o interfaces. Campos, métodos o constructores declarados privados están estrictamente controlados, lo que significa que no pueden ser accedidos desde cualquier otra parte fuera de la propia clase.

## Modificadores de no acceso: Public, Private, Protected

---

Los modificadores de acceso son utilizados para controlar la visibilidad de las clases, variables, métodos o constructores. Y los modificadores de no acceso son empleados para proveer otras funcionalidades como sincronizar un método bloque o restringir la serialización de una variable, etc.

**Static:** Los miembros declarados como estáticos son comunes para todas las instancias de una clase. Pueden ser accedidos sin haber creado el objeto de una clase. Se pueden acceder directamente a ellos con el nombre de la clase.

**Final:** Este modificador le dice al compilador que no debe cambiar el valor de la variable una vez asignada. Si se aplica a una clase, esta no puede ser una sub-clase. Si se aplica a un método, el método no puede sobre escribir en una sub-clase.

**Abstract:** Hay situaciones en las cuales se querrá definir una súper clase que declare la estructura de una abstracción dada sin haber provisto una completa implementación de cada método. Este modificador solo puede ser aplicable a clases y métodos.

**Synchronized:** Este modificador es usado para controlar el acceso de un método o block en particular por múltiples hilos. Solo un hilo puede entrar dentro del método bloque declarado como sincronizado.

**Stricfp:** Este modificador es cuando para operaciones de punto flotante. Esta palabra clave asegura que se obtendrá la misma presentación del punto flotante en cada plataforma, haciendo las variables de float más consistentes entre múltiples plataformas.

## JSP, JSF JPA

---

Java EE es una plataforma de programación con características orientadas a la computación distribuida y servicios web. Este incluye multiples tecnologías que sirven diversos propósitos tales como generar paginas web, leer y escribir una base de datos, etc.

**JSP** (Java Server Pages) es una tecnología del lado del servidor para construir páginas web dinámicas con java. JSP tiene acceso a todas las otras APIs de java tales como JDBC para trabajar con base de datos.

**JSF** (Java Server Faces) es una tecnología / framework especializado para la creación de interfaces web con widgets UI (User Interface) reutilizables.

**JPA** (Java Persistence API) es una especificación de java para el acceso, persistencia y gestión/mapeo de datos entre clases OOP de java y una base de datos relacional. Hibernate es el ORM (Object Relacional Mapper) dentro de JPA.

### Persistencia

La persistencia, en ciencias de la computación, es un sustantivo que describe datos que sobreviven al proceso que lo creó. La persistencia de Java podría definirse como el almacenamiento de cualquier nivel de persistencia utilizando el lenguaje de programación Java.

Existen muchas maneras de crear datos persistentes en java, la mayoría de ellos se encuentran en bases de datos. La gran mayoría de cosas que se realizan en una computadora involucran el almacenamiento de datos, involucrando consigo las bases de datos relacionales.

## Conclusion

---

Java como lenguaje de programación ofrece una gran versatilidad y herramientas no solo para el desarrollo de programas, si no también para el desarrollo de servicios web y paginas web, ejemplificamos con JSP, JSF Y JPA.

Su estructura de datos al ser un lenguaje codificado da al programador viabilidad para la creación de métodos, clases, variables, etc. Lo que lo hace un lenguaje con gran potencia y optimo para el desarrollo orientado a objetos y a su vez plenamente compatible con frameworks.

Por ello Java es considerado un lenguaje ampliamente utilizado dentro del ámbito empresarial en gran parte del mundo.

## Bibliografía

- (2018). Java Persistence. 18/01/2019, de WikiBooks Sitio web: [https://en.wikibooks.org/wiki/Java\\_Persistence](https://en.wikibooks.org/wiki/Java_Persistence)
  - Carol McDonald. (2009). JSF 2.0, JPA, GlassFish and MySQL. 18/01/2019, de DZone Sitio web: <https://dzone.com/articles/jsf-20-jpa-glassfish-and-mysql>
  - (2018). Jakarta EE Tutorial. 18/01/2019, de Techiediaries Sitio web: <https://www.techiediaries.com/jakarta-ee-tutorial/>
  - Fernando Pech-May , Mario A. Gomez-Rodriguez , Luis A. de la Cruz-Diaz , Salvador U. Lara-Jeronimo. Desarrollo de Aplicaciones web con JPA, EJB, JSF y PrimeFaces. Tabasco, México: Instituto Tecnológico Superior de los Ríos.
  - JAVA CLASS, METHODS, INSTANCE VARIABLES. 18/01/2019, de IDC Technologies Sitio web: [http://www.idc-online.com/technical\\_references/pdfs/information\\_technology/Java\\_Class\\_methods\\_instance\\_variables.pdf](http://www.idc-online.com/technical_references/pdfs/information_technology/Java_Class_methods_instance_variables.pdf)
  - (2015). What Are Access And Non-Access Modifiers In Java?. 18/01/2019, de Java Tutorial Site For Beginners Sitio web: <https://javaconceptoftheday.com/access-and-non-access-modifiers-in-java/>
- 
-