**MI GUÍA PERSONAL DE LENGUAJE JAVA**

**By:... Beanswalker**

Esta guía la he construido a partir de mis experiencias con el Lenguaje de Programación Java y el IDE Netbeans Corriendo sobre GNU/Linux Deepin.

La Instalación y conficuración del Sistema Operativo, del IDE Netbeans con sus plugins y del OpenJDK está descrita en el Manual respectivo(Configuración de mi GNULinux para desarrollo.doc) que, por supuesto, está en Github.

**Lenguaje JAVA**

Empezamos nuestro aprendizaje del Lenguaje de Programación Java, el cual trabajaremos sobre el OpenJDK versión 11 y Utilizando mayoritariamente el IDE Netbeans y en casos muy aislados, el IDE Eclipse.

Como apoyos de aprendizaje. Vamos a utilizar:...

* Los cursos:...
  + Universidad de Java
  + Universidad de Java 2

Los utilizaremos Consecutivamente, ya que son del mismo autor y, en efecto, el 2 es la continuación. Están contenidos en mi Espacio de nube de MEGA.

* Los Libros:...
  + Cómo Programas en Java edición N° 10 de los hermanos Deitel (año 2016. Es lo que hay :P)
  + Programaci+on Orientada a Objetos Usando Java del Colombiano Hector Arturo Flórez Fernández (año 2012 porque también es lo que hay :P :P )

Java es un Lenguaje de programación que, en escencia es Programación Orientada a Objetos.

**Generalidades:..**

1. Todo el código que picamos de Java, se contiene en archivos que tienen la extensión .java
2. Java utiliza un compilador tanto como Un interprete, ya que al ejecutarlo con el IDE por primera vez, el código es Compilado a un bytecode(código intermedio) por medio de un compilador llamado JavaCompiler, mas conocido como JavaC, convirtiendo nuestros archivos .java en archivos .class y posteriormente, cada vez que el mismo código de nuevo es ejecutado, el bytecode(todos los archivos .class) es interpretado en la JVM. Esta peculiaridad la vamos a comprender en la marcha.
3. Gracias a la JVM, Java es multiplataforma, y el mismo código escrito en X sistemaOperativo y compilado a bytecode, puede ejecutarse sin problemas y sin modificaciones en GNU/Linux u otro sistema Operativo que tenga instalada la JVM respectiva
4. El lenguaje de programación Java puede implementarse en muchos ámbitos de la tecnología informática (en componentes electrónicos, programas para electrodomesticos, en desarrollo de software a nivel empresarial para sistemas operativos o para la Web, así como para dispositivos móviles y en varios escenarios mas)
5. El Kit de desarrollo o JDK Tiene dos implementacioes principales, la de Oracle, llamada JDK y la de la comunidad Libre, el OpenJDK, que es el que vamos a utilizar
6. Pendiente

A grandes rasgos, podemos representar el proceso de creación y ejecución de un programa Java de la siguiente manera:...

**Archivos .java**

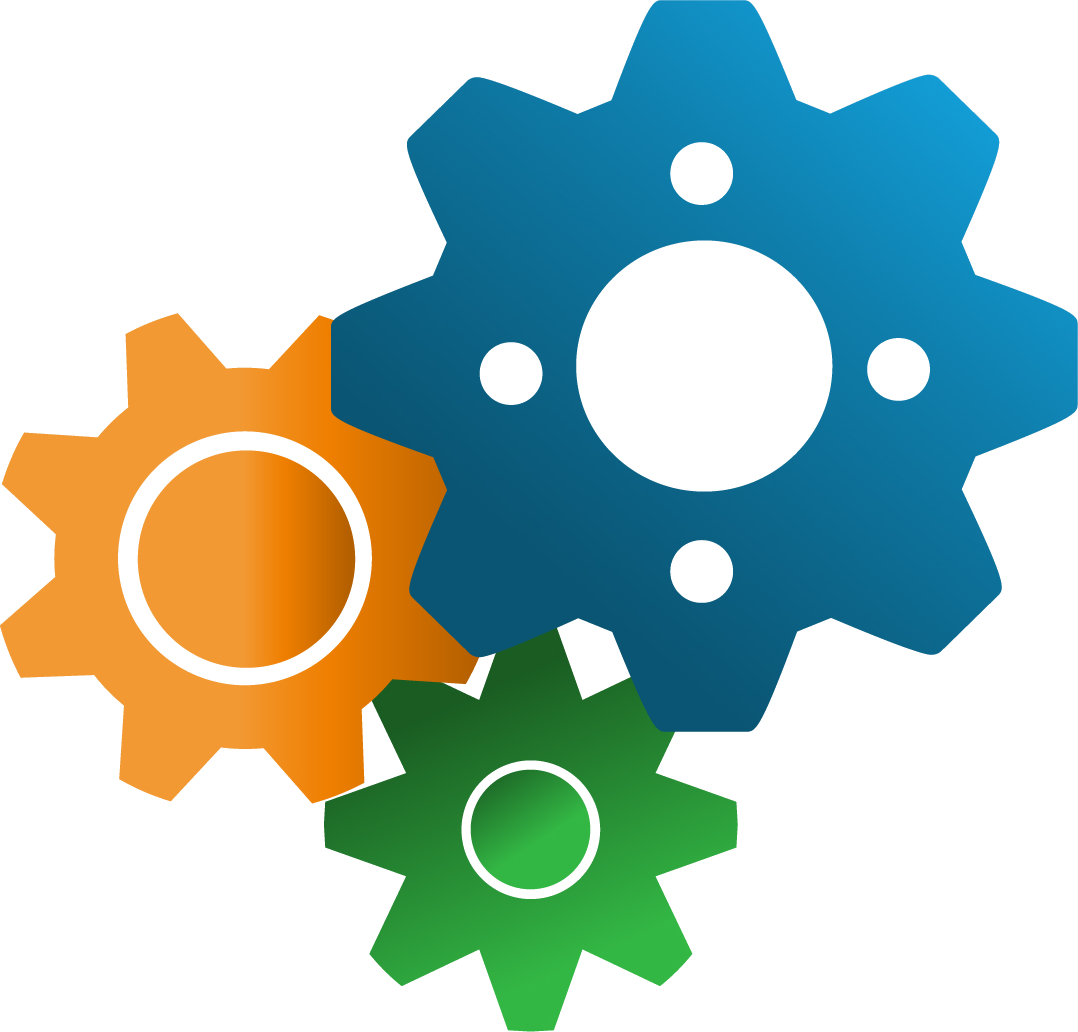
**Saved as**

**Class2**

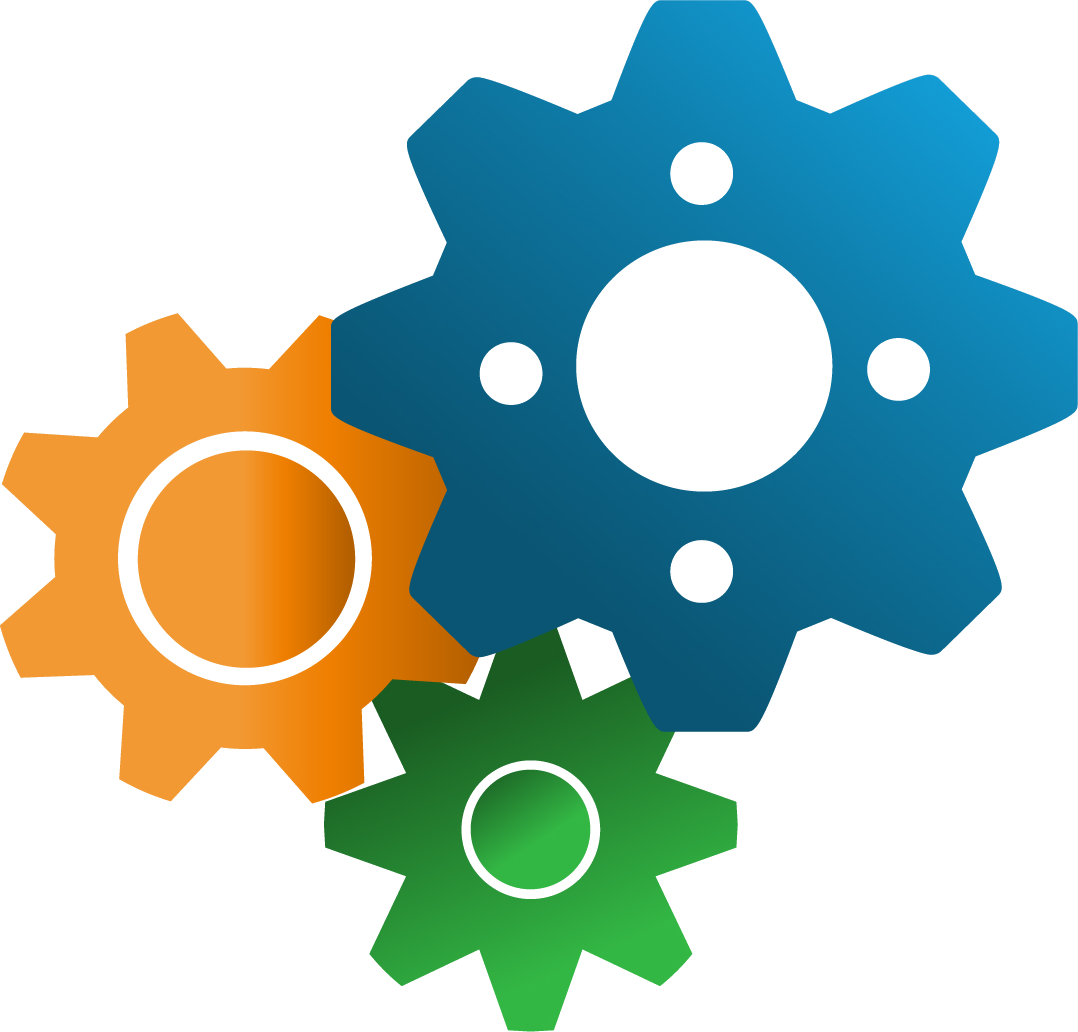
**Class1**

**Nuestro código implementado, organizado y conectado mediante clases**

**Class3**

****

**JavaCompiler...**

****

**Class1**

**JVM**

**Read with**

**Class2**

**Archivos .class**

**Class3**

**(ByteCode)**

**Ejecución**

**Output or Results**

1. Una de las cualidades de Java es una implementación de un Garbage Collector(recolector de basura) que se encarga de asignación de datos en la memoria RAM de la computadora, como también se encarga de eliminar lod datos que ya no están siendo utilizados durante la ejecución y/o que ya no es necesario mantenerlos(liberación automática de memoria utilizada), de manera que el programador puede concentrar sus esfuerzos unicamente en la creación de datos primitivos(variables comunes) y objetos, como de las operaciones entre estos. La gestión de memoria corre por cuenta de la JVM.
2. Java es Case-sensitive y esto debemos tenerlo en cuenta cuando vamos a utilizar variables u objetos que ya hemos implementado
3. Recordemos que las Funciones y procedimientos en la Programación Orientada a Objetos pasan a llamarse “métodos”

Aunque seguimos llamandolos Funciones cuando ejecutan el dódigo que contienen y retornan un valor, y procedimientos cuando ejecutan su código pero no retornan valor alguno.

**Picando Código:...**

Para empezar a escribir o “picar” código de Java. Es importante saber sobre que versión de la plataforma Java estamos trabajando. Para ello podemos ejecutar en nuestra terminal, dos comandos que nos muestran, por un lado la versión del JDK utilizado y por otro lado, la versión del compilador JavaC, implementados en nuestro sistema operativo:...

* Java --version

openjdk version "11" 2018-09-25

OpenJDK Runtime Environment (build 11+13-Debian-2)

OpenJDK 64-Bit Server VM (build 11+13-Debian-2, mixed mode, sharing)

* Javac --version

javac 11

Podemos ver que tenémos en nuestra computadora el OpenJDK de 64 Bit y el compilador JavaC, ambos en su versión 11

**Nuestro primer programa con Netbeans:...** (video 11)

* Creamos un nuevo proyecto File, New Project, Java, Java Application
* Botón Next
* Ingresamos el nombre de nuestro proyecto(java\_one) y su ubicación. Como es la creación de nuestro proyecto, habilitamos la opción “create Main Class” e ingresamos en esa casilla “Main”

Main Class es el archivo de arranque de los proyectos Java

* Botón Finish
* Vemos entonces que se genera un proyecto básico llamado “java\_one” con un subdirectorio “Source Packages” y dentro de éste, un paquete de código que, asu vez contiene un archivo “Main.java” en el que vamos a implementar nuestro primer código.
* Podemos ver que el archivo ya tiene generada la clase Pública Main y dentro de ésta, está el metodo estático main, aue es donde vamos a escribir nuestro primer código:...

public static void main(String[] args)

{

System.out.println(" Hello, I am demonscript!!!:... ");

}

* Presionamos el botón verde “Run Project” o la tecla F6, con lo que se inicia el proceso de verificación, compilación y posterior ejecucuón de nuestro programa. Resultado que puedo ver en el mismo Netbeans

**TRICK:...**

**Algo MUYYYYY!!! curioso que apenas descrubro es que como ya tenía instalado el sistema de control de versiones GIT. Puedo ver en el mismo Netbeans el Historial de cambios de mi archivo:...**

* + **Click derecho sobre el archivo**
  + **Opción “History”**
  + **“Show History”**

**Esto, desde luego que es MAGNIFICO!!! :)**