# GNU Emacs como editor de código Python

Edison Ibáñez

October 22, 2018

# Introducción

**GNU Emacs** 

# ¿Que es GNU Emacs?



Es un editor de texto libre, de código abierto, extensible y personalizable, es multiplataforma y lo encontraremos disponible para Gnu/Linux, Windows y Mac. Además está desarrollado por el proyecto GNU y publicado bajo licencia GNU GPL.

## ¿Como instalar GNU Emacs?



GNU Emacs viene en todas de Distribuciones GNU/Linux, por lo que se lo puede instalar desde los repositorios oficiales de cada una. Para encontrar instrucciones mas detallas de como instalar GNU Emacs en cualquier Sistema Operativo pueden visitar el siguiente sitio: https://www.gnu.org/software/emacs/download.html

# ¿Que es Python?



Python es un lenguaje de programación interpretado cuya filosofía hace hincapié en una sintaxis que favorezca un código legible. Creado a finales de los ochenta por Guido van Rossum en el Centro para las Matemáticas y la Informática....

#### ¿Que es un IDE?

Un entorno de desarrollo integrado o entorno de desarrollo interactivo, en inglés Integrated Development Environment  $(\mathsf{IDE})^1$ .

- Características:
  - Multiplataforma
  - Soporte para diversos lenguajes de programación
  - Integración con Sistemas de Control de Versiones
  - Reconocimiento de Sintaxis
  - Extensiones y Componentes para el IDE
  - Integración con Framework populares
  - Depurador
  - Importar y Exportar proyectos
  - Múltiples idiomas
  - Manual de Usuarios y Ayuda



Introducción

**GNU Emacs** 

#### **GNU Emacs**

# **GNU Emacs**

#### Uso Básico



GNU Emacs trae con sigo un excelente tutorial, para acceder a este bata con presionar la combinación de teclas: C-h>t.

# Archivo de Configuración



La configuración de GNU Emacs se la hace usando el lenguaje elisp o Emacs Lisp, para esto es necesario crear un archivo llamado init.el en el \$HOME de nuestro usuario<sup>2</sup>:

#### Shell

\$ touch ~/.emacs.d/init.el

#### Garbage Collection o Recolección de Basura

Para darle un poco de velocidad al momento de cargar nuestra configuración, vamos a establecer el umbral de Garbage Collection alto durante el inicio.

#### Configuración de Repositorios

Antes de empezar a editar la configuración de GNU Emacs, debemos indicarle al ELPA de donde debe descargar los paquetes, para esto configuramos los repositorios de la siguiente manera.

```
1  (setq package-archives
2   '(("gnu" . "http://elpa.gnu.org/packages/")
3    ("melpa" . "http://melpa.org/packages/")
4    ("org" . "https://orgmode.org/elpa/"))
5    package-archive-priorities
6   '(("melpa" . 10)
7    ("gnu" . 5)
8    ("org" . 0)))
```

#### use-package, bind-key, diminish

use-package es una macro declaración que permite aislar la configuración de un paquete en nuestro archivo \$HOME/.emacs.d/init.el

```
(setq load-prefer-newer t)
      ;; initialize the packages and create the packages list if not exists
     (package-initialize)
      (when (not package-archive-contents)
       (package-refresh-contents))
      ;; install use-package if not exists
      (unless (package-installed-p 'use-package)
        (package-install 'use-package))
10
     (eval-when-compile
11
       (require 'use-package))
12
13
      (require 'diminish)
      (require 'bind-key)
14
```

#### Cache

Para mantener lo mas limpio posible nuestro directorio \$HOME/.emacs.d vamos a utilizar el paquete no-littering, y vamos a empezar a enviar todo el contenido que no forme parte de nuestra configuración.

```
(use-package no-littering
 2
          :ensure t
3
          :init
          (setq no-littering-var-directory
                (expand-file-name "cache/var/" user-emacs-directory)
5
                no-littering-etc-directory
                (expand-file-name "cache/etc/" user-emacs-directory)))
8
9
10
      (setq-default semanticdb-default-save-directory
                    (concat no-littering-var-directory "semanticdb")
11
                    url-configuration-directory
12
                    (concat no-littering-var-directory "url")
13
                    eshell-directory-name
14
                    (concat no-littering-var-directory "eshell" ))
15
```

#### **Bookmarks**