

# Snippet en LiClipse

30 de Noviembre de 2016 a las 12:57

Para programar en *Django* utilizo [LiClipse](#), que es un *IDE* basado en *Eclipse* que incluye el plugin *PyDev* para programar en *Python* o *Django*, también soporta otros lenguajes.

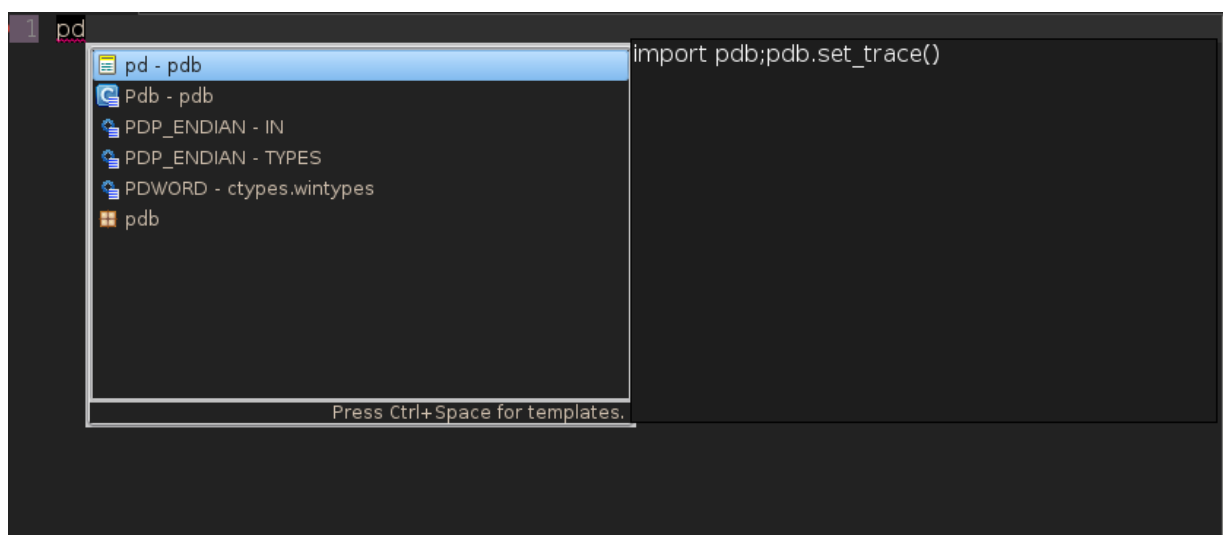
Una funcionalidad que estoy empezando a utilizar y me ha resultado muy útil, es el uso de *snippet*, que son pequeños fragmentos de código.

Cualquier programador utiliza de forma repetida diversos fragmentos de código mientras desarrolla, en vez de escribirlo una y otra vez, se pueden guardar el fragmento de código, y cuando haga falta insertarlo en el lugar correspondiente. Esto implica un ahorro de tiempo a tener en cuenta y libra al desarrollador de memorizar o buscar ese código.

En esta entrada veremos como utilizar, configurar y crear snippet de código en *LiClipse*.

Se utilizara la versión 2.3.0 de *LiClipse* corriendo en *Fedora 22*.

*LiClipse* proporciona una serie de snippet de código que tienen asociada una cadena de caracteres, cuando un desarrollador escribe en el editor una cadena de texto que tiene asociada un snippet, el editor muestra el código asociado y se puede insertar en el código. Si no muestra el snippet de forma automática, sera necesario pulsar *Ctrl* + *Space* para mostrarlo.



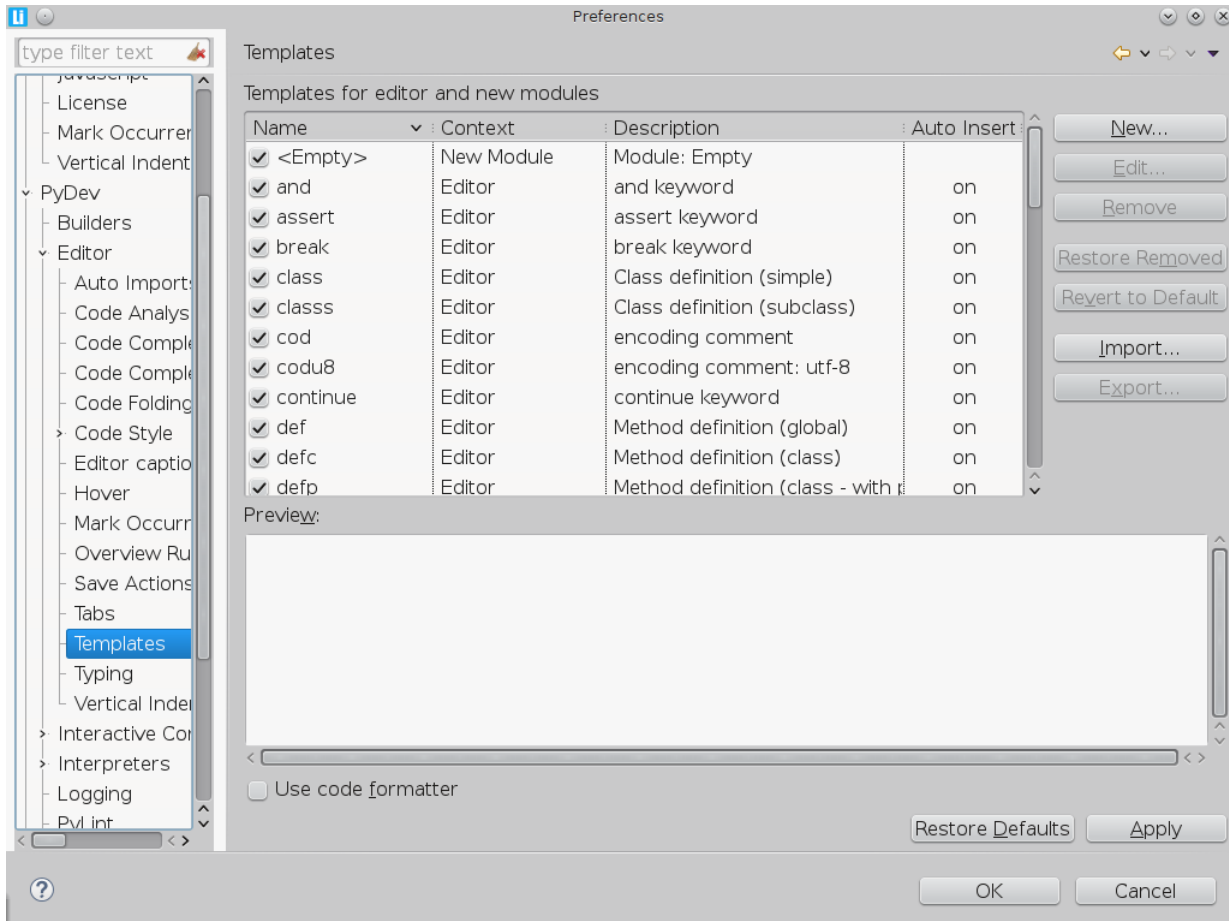
Por ejemplo, si escribimos *pd*, esta cadena de texto tiene asociado, *import pdb; pdb.set\_trace()*, que se utiliza para depurar código.

Blog de J.A. Jimenez Toro, [rooteando.com](http://rooteando.com)

Contenido esta bajo licencia [Creative Commons Reconocimiento No Comercial Sin Obra Derivada 3.0 Unported License](#)

Todos los snippet de LiClipse pueden ser configurados mediante una serie de parámetros e incluso el desarrollador podrá añadir sus propios snippet.

Para realizar esta tarea, debemos situarnos en el apartado de *Templates*, que se encuentra en *Menu Window->Preferences->PyDev->Editor->Templates*.



Dentro del apartado *Templates*, tenemos:

**Name:** Especifica la cadena de texto que identificara el snippet.

**Context:** Donde se aplica, normalmente en el editor.

**Description:** Breve descripción del código que insertar.

**Auto Insert:** Si esta activado (on), automáticamente aparecerá el snippet que corresponde, si esta desactivado, deberemos pulsar Ctrl + Space para que lo muestre.

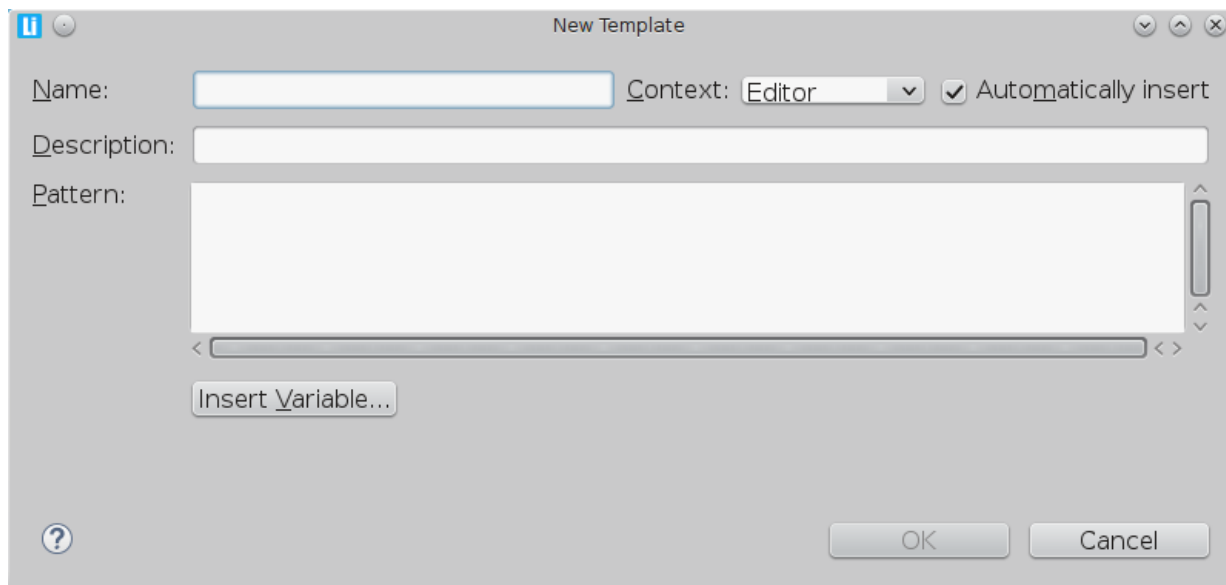
Todos los snippet que se muestran pueden ser editados, modificando diversos parámetros. Pulsando en cualquiera de ellos se activa el botón de *Edit*, que nos permitirá personalizar el snippet.

Existe un botón *Remove*, para borrar el snippet, botón *Export* para exportar y *Import* para importar el snippet seleccionado, utilizan archivos xml. Por último, tenemos el

Blog de J.A. Jimenez Toro, [rooteando.com](http://rooteando.com)

Contenido esta bajo licencia [Creative Commons Reconocimiento No Comercial Sin Obra Derivada 3.0 Unported License](https://creativecommons.org/licenses/by-nc-sa/3.0/)

botón *New* para crear un snippet.



*Name:* Especifica el nombre para identificarlo.

*Context:* Especifica donde sera utilizado, contexto, el snippet.

*Description:* Un breve texto donde se explica para que sirve.

*Pattern:* Donde escribimos el trozo de código que queremos utilizar.

En la parte inferior tenemos un botón *Insert Variable*, que permite insertar una serie de variables ha escoger de una lista, con el uso de variables podemos definir parámetros que serán rellenados cuando utilizamos en snippet.

Ya solo queda escribir tu propios snippet o utilizar los que proporciona el programa, evitando escribir el mismo código una y otra vez.