

# Tarea para DAW06.

1. Indica cada uno de los pasos que deberías de dar para proceder a la instalación de **phpDocumentor**, suponiendo que vas a partir de una máquina en la que tienes instalado la distribución **Ubuntu** y en la que ya están instalados y correctamente configurados **apache** y **php**.

Primero hemos de instalar **pear**, para lo que tecleamos:

```
apt-get install php-pear
```

Ahora configuramos su directorio de trabajo en **/var/www**:

```
pear config-set data_dir /var/www
```

Pasamos a instalar **phpDocumentor** con todas sus dependencias:

```
pear install --alldeps PhpDocumentor
```

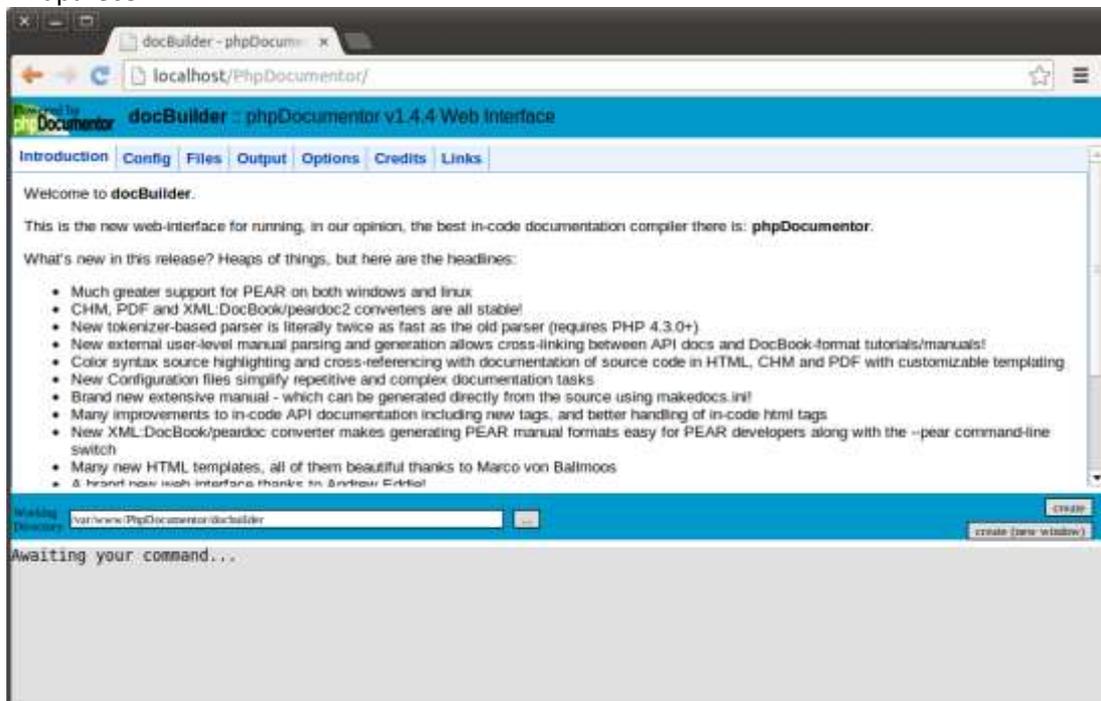
Ya sólo nos queda crear un directorio de salida para **phpDocumentor** y cambiar su propietario a **www-data**:

```
mkdir /var/www/docs  
chown www-data /var/www/docs
```

Para comprobar que todo ha salido de forma correcta, dentro de un navegador tecleamos:

```
http://localhost/PhpDocumentor
```

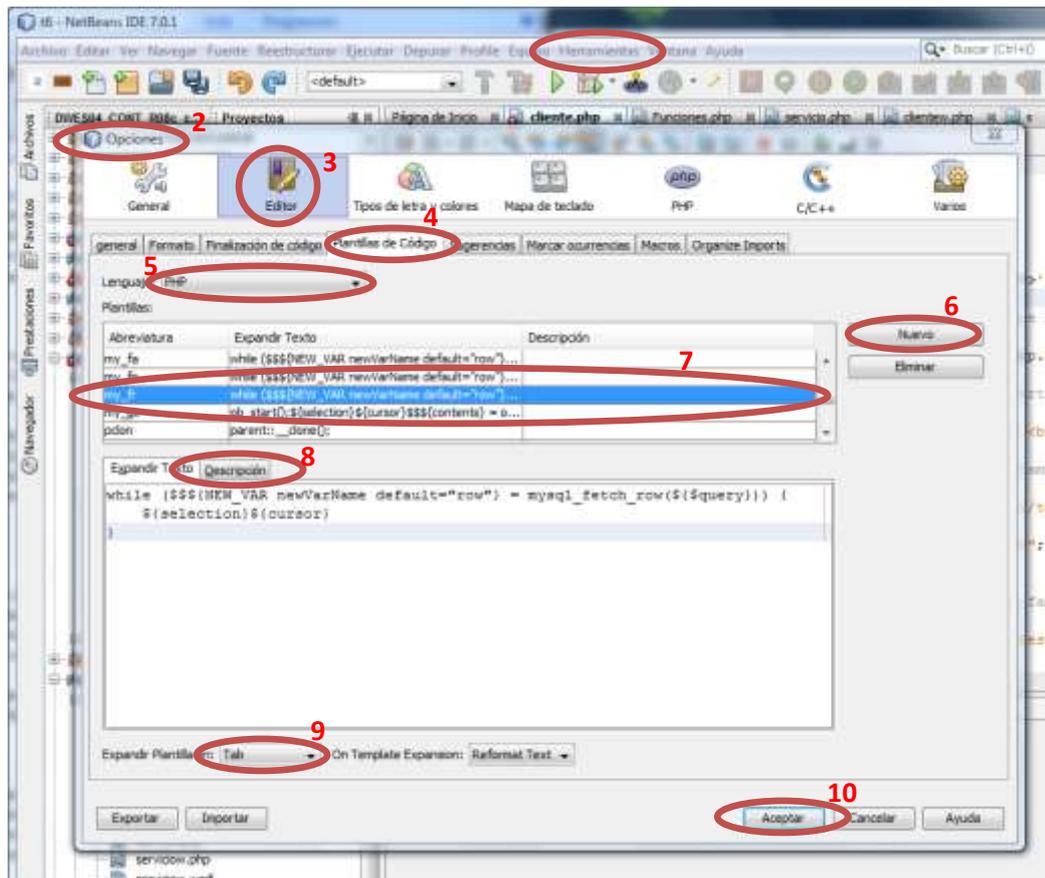
Y aparece:



2. Explica en qué consisten las plantillas de código en el caso de **Javadoc** y cada uno de sus componentes.

Con las plantillas **Javadoc** conseguimos documentar una aplicación, sus clases y métodos, siendo de mucha utilidad para una posterior actualización o refinamiento de la misma. También conseguimos que cuando estemos llamando a un método o clase determinado, se nos muestre una sugerencia para el código, la cual podemos coger con tan solo pulsar **Ctrl+Intro** o cualquier otra combinación de teclas que definamos.

Para poder crear nuestras propias plantillas en **NetBeans** tendremos que seleccionar la opción **Herramientas - Opciones - Editor - Plantillas de Código** y en el apartado de lenguaje seleccionamos aquel al que deseamos aplicarle la nueva plantilla que creamos (según el orden que nuestro en la imagen de la página siguiente).



Una plantilla se compone de nombre, descripción, contexto en función del lenguaje y un patrón o código de la plantilla. Este último puede estar compuesto de texto fijo o una serie de variables, como pueden ser:

- ✓ `{cursor}`: posición en la que se establecerá el cursor de texto tras desplegar el código de la plantilla.
- ✓ `{enclosing type}`: tipo de la clase en la que nos encontramos
- ✓ `{enclosing method}`: nombre del método en el que nos encontramos
- ✓ `{year}`: año en curso
- ✓ `{time}`: hora en curso

3. Dispones de una máquina que cuenta con el sistema operativo Ubuntu recientemente actualizado, en la que está el entorno de red configurado y, además, dispones de conexión a Internet y estás trabajando con la cuenta del usuario root. Indica cada uno de los pasos y comandos implicados en ellos para conseguir hacer lo siguiente:

- ✓ Suponiendo que el sistema ya tiene instalado las siguientes librerías de las que Git depende: `curl`, `zlib`, `openssl`, `expat`, y `libiconv`, pasos para realizar la compilación e instalación de Git considerando que ya disponemos del paquete `git-1.7.9.5.tar.bz2`

```
tar -xvfz git-1.7.9.5.tar.bz2
cd git-1.7.9.5
apt-get build-dep git-core
apt-get install libssl-dev
make prefix=/usr/local all doc
make prefix=/usr/local install install-doc
```

- ✓ Cómo obtener Git a través del propio Git para futuras actualizaciones, de manera que descargaría automáticamente el código fuente desde su repositorio.

Una vez que tengamos instalado git, teclearíamos en el terminal:

```
git clone git://git.kernel.org/pub/scm/git/git.git
```

- ✓ Comprobar la versión que se ha instalado de Git.

```
git --version
```

- ✓ Establecer el nombre de usuario y dirección de correo electrónico en la configuración de Git.

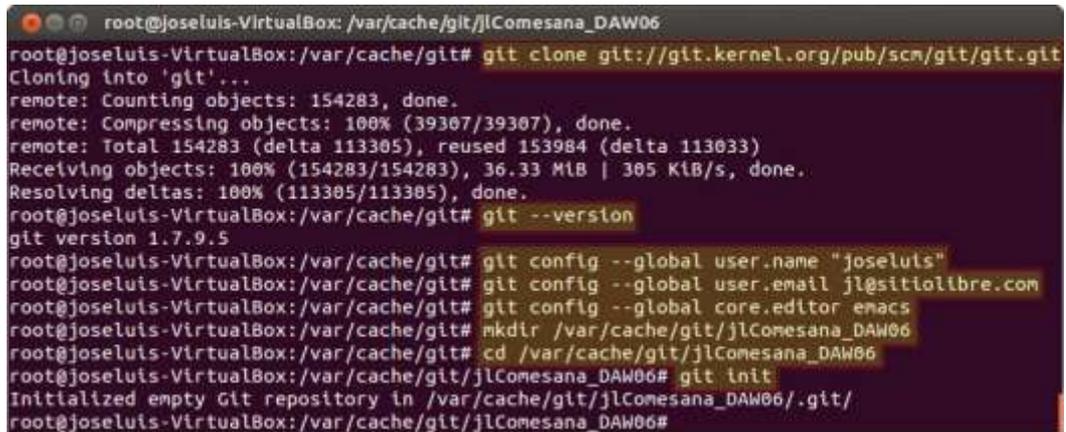
```
git config --global user.name "joseluis"  
git config --global user.email jl@sitiolibre.com
```

- ✓ Cambiar el editor de texto que trae por defecto Git al editor emacs.

```
git config --global core.editor emacs
```

- ✓ Dentro de la carpeta `/var/cache/git/` crear una carpeta para un nuevo proyecto denominado `tunombre DAW06` e iniciar un repositorio del nuevo proyecto, donde `tunombre` debes sustituirlo por la inicial de tu nombre, seguido de tu primer apellido.

```
mkdir /var/cache/git/jlComesana_DAW06  
cd /var/cache/git/jlComesana_DAW06  
git init
```



```
root@joseluis-VirtualBox: /var/cache/git/jlComesana_DAW06  
root@joseluis-VirtualBox:/var/cache/git# git clone git://git.kernel.org/pub/scm/git/git.git  
Cloning into 'git'...  
remote: Counting objects: 154283, done.  
remote: Compressing objects: 100% (39307/39307), done.  
remote: Total 154283 (delta 113305), reused 153984 (delta 113033)  
Receiving objects: 100% (154283/154283), 36.33 MiB | 305 KiB/s, done.  
Resolving deltas: 100% (113305/113305), done.  
root@joseluis-VirtualBox:/var/cache/git# git --version  
git version 1.7.9.5  
root@joseluis-VirtualBox:/var/cache/git# git config --global user.name "joseluis"  
root@joseluis-VirtualBox:/var/cache/git# git config --global user.email jl@sitiolibre.com  
root@joseluis-VirtualBox:/var/cache/git# git config --global core.editor emacs  
root@joseluis-VirtualBox:/var/cache/git# mkdir /var/cache/git/jlComesana_DAW06  
root@joseluis-VirtualBox:/var/cache/git# cd /var/cache/git/jlComesana_DAW06  
root@joseluis-VirtualBox:/var/cache/git/jlComesana_DAW06# git init  
Initialized empty Git repository in /var/cache/git/jlComesana_DAW06/.git/  
root@joseluis-VirtualBox:/var/cache/git/jlComesana_DAW06#
```