**¿Cuáles son las ventajas de usar un ambiente virtual de Python?**

Python, como otros lenguajes de programación modernos tienen su manera única de descargar, almacenar y resolver dependencias de paquetes.

En el caso de Python, los paquetes externos instalados con *easy\_install* o *pip* son típicamente ubicados en uno de los directorios señalados por la variable *site.getsitepackages*.

Esto funciona bien cuando los proyectos requieren de un único ambiente, pero en el momento en que se requiera de diferentes versiones de un mismo paquete o del mismo framework de Python tendremos problemas porque los paquetes suelen almacenarse en base a nombre.

Es decir, si ocupáramos un *PaqueteA* Versión 1.0 y *PaqueteA* Versión 1.1, el sistema solo encontraría el ultimo *PaqueteA* que fue instalado.

La razón principal de la creación de ambientes virtuales de Python es con el fin de crear ambientes aislados para los proyectos. Esto permite que cada proyecto tenga sus propias dependencias independientes de lo que otros proyectos en otros ambientes estén utilizando.

**¿Qué es NumPy?**

NumPy es el paquete fundamental para computación de cálculos científicos en Python. Es una librería de Python que expone arreglos multidimensionales (así como derivados más simples como matrices), y variedad de herramientas y rutinas para realizar operaciones rápidas en arreglos, incluyendo matemáticas, lógicas, manipulación de dimensiones, ordenamiento, transformadas discretas de Fourier, algebra linear, operaciones estadísticas básicas y muchas más.

**Describa ejemplos de 2 aplicaciones de NumPy**

NumPy es una librería que optimiza las operaciones con arreglos en términos de consumo de memoria y velocidad de ejecución, y brilla por su utilidad en aplicaciones de algebra lineal y transformada discreta de Fourier en sistemas embebidos que buscan alto desempeño.

**¿Para qué sirve Cmake?**

Cmake es una herramienta multi plataforma de generación de código. Su nombre es una abreviación de "Cross Platorm Make". Es una suite de herramientas de más algo nivel que la herramienta "Make" de Unix, siendo similar a Autotools.

Se usa para controlar el proceso de compilación del software usando archivos de configuración sencillos e independientes de la plataforma, que sirven para que Cmake genere makefiles nativos del ambiente de trabajo en que se usan.

**Que Ventajas/Desventajas tiene Cmake antes otras herramientas de configuración de Makefiles (Autotools, qmake, Scons, etc) en sistemas embebidos?**

**Ventajas**

* Mas rápido
* Requiere menos código para realizar tareas comunes
* Mas estable
* Soporta creación de archivos de proyecto para Code::Blocks, Xcode, Visual Studio, entre otros

**Desventajas**

* Usa su propio lenguaje de scripting
* Al tener un lenguaje propio de scripting limita la extensibilidad de sus funciones por otros usuarios
* Requiere del uso de "make"