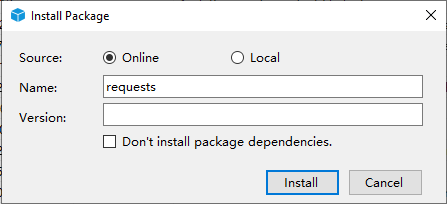
# AutoTest Studio Getting started tutorials 3: Package Manager

The python environment operated by Package Manager is the python environment of the current project. If a virtual environment is used, it refers to the python package of the virtual environment. For users, there is no need to care about whether the python environment is a virtual environment. Installation, uninstallation, and viewing operations are all identical.

As an example, the requests package used in the previous section of this article describes how to install, uninstall, view, upgrade, and configure the installation source.

**Installation Online**

Online installation, that is, install the python package directly through the network, this method is recommended.



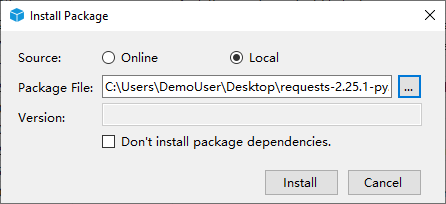
**Name**: python package name;

**Version**: The version of the package. When it is empty, the most detailed version is installed by default;

**Don’t install package dependencies**: Don’t install related dependencies, only install the specified packages. It is recommended not to check.

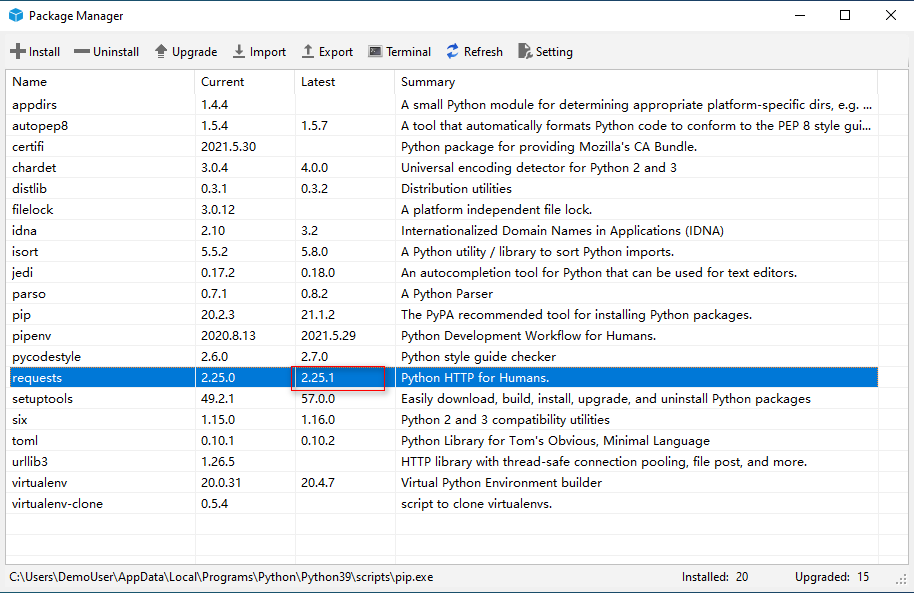
**Installation Offline**

If the package already exists locally, you can use the local installation, select "**Local**" and click “**...**"to select the path of the Python package.

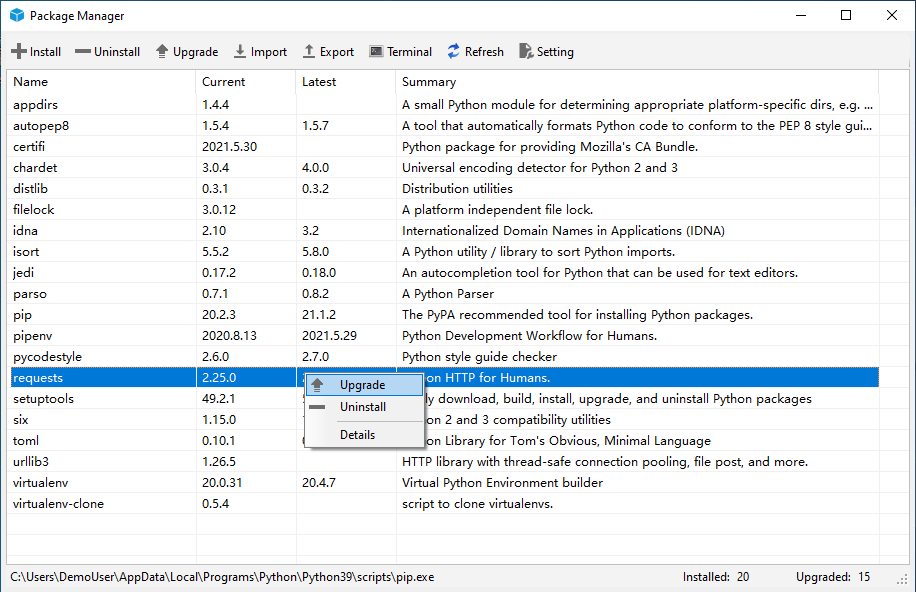


**Upgrade**

Package Manager automatically checks for the latest version and lists the thinnest versions in Latest, and to demonstrate the upgrade, try installing version 2.25.0 of requests first, as follows.



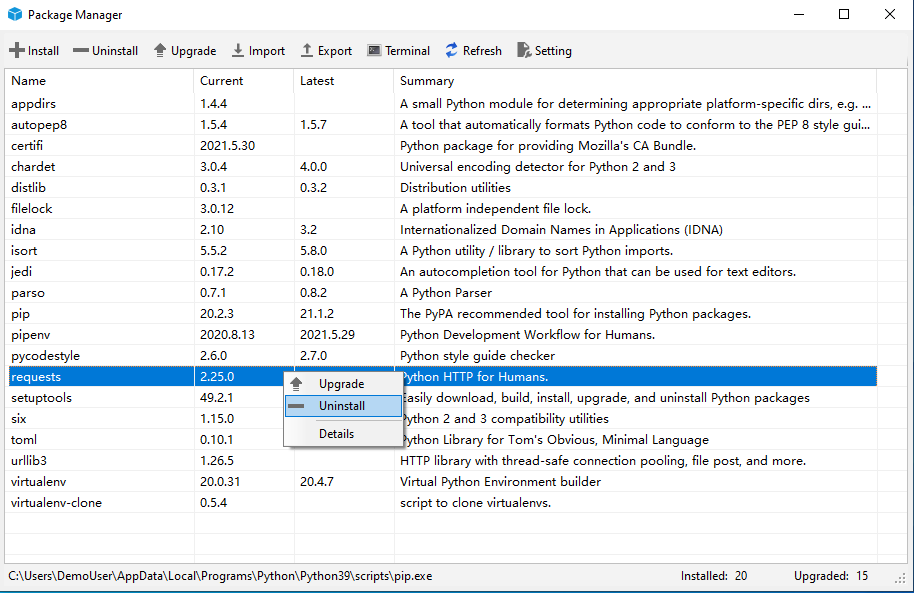
Select request in the list, click "Upgrade" and Package Manager will automatically upgrade to the latest version.



**Uninstall**

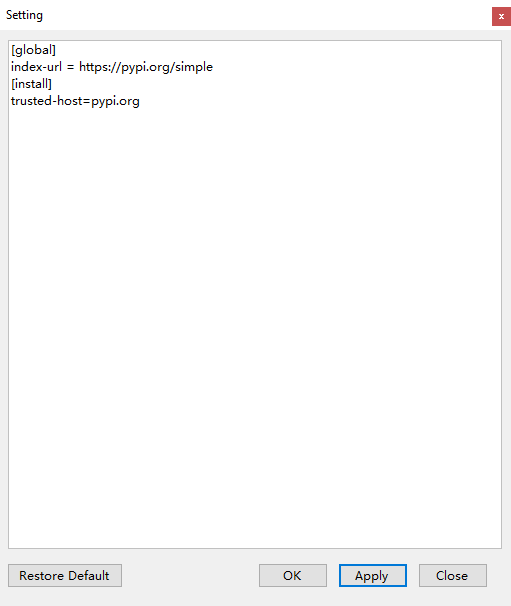
Select the package to be uninstalled and click "**Uninstall**" to uninstall the package.

**Note**: When uninstalling a package, dependent packages will not be uninstalled.



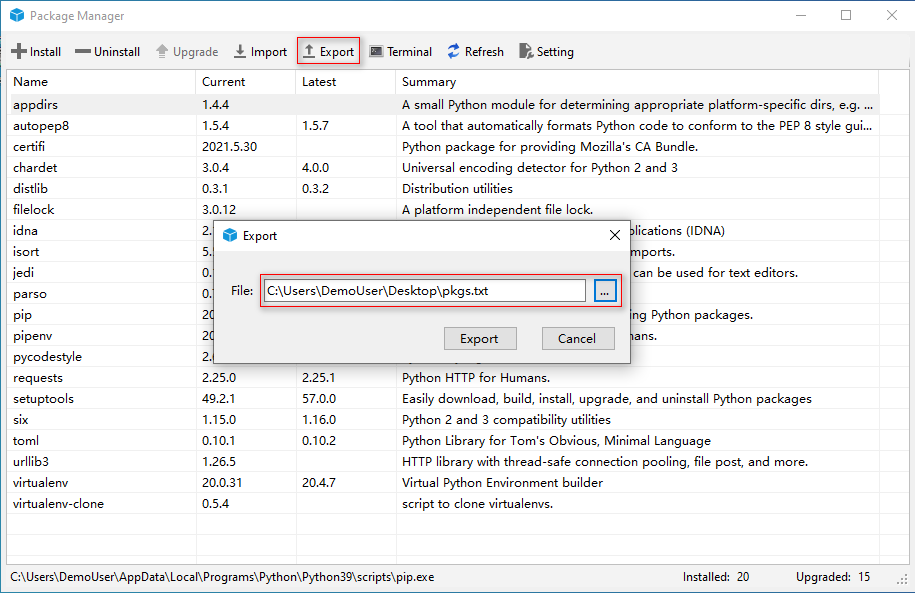
**Configure the installation source**

Package Manager allows users to configure the installation source, which is actually to configure the pip.ini file. Click "**Setting**" on the toolbar. After editing the installation source information in the pop-up dialog box, click "**OK**" to save. If you need to restore to the default installation source, you can click the "**Restore Defualt**" button at the bottom. The default installation source is the official python installation source.



**Export package list**

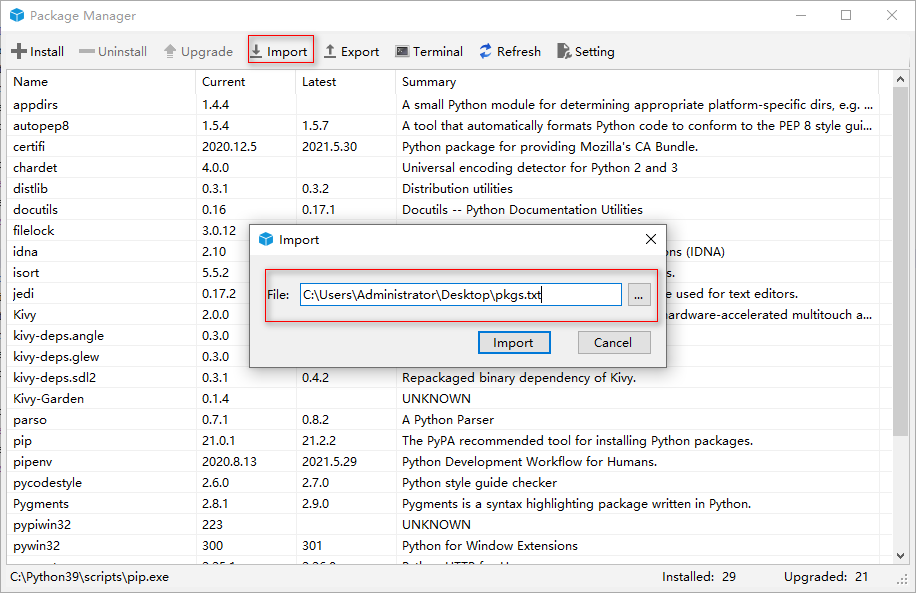
Package Manager can export the list of installed packages in the current environment.



The exported list is actually a text file containing the python package name and version.

**Import package list**

As shown in the figure below, the package in the text file can be imported into the current environment, and the import and export functions are mainly used to migrate the current environment package information to another python environment. It should be noted that when the package is imported, it is actually installed online, so the environment needs to be connected to the Internet.



Package Manager avoids the cumbersome management of packages through the command line. If you are familiar with pip and pipenv commands, you can also manage python packages directly through the command line.