[**Installing cx\_Oracle on Windows**](https://cx-oracle.readthedocs.io/en/latest/installation.html#id16)

[**Install cx\_Oracle**](https://cx-oracle.readthedocs.io/en/latest/installation.html#id17)

Use Python’s [Pip](http://pip.readthedocs.io/en/latest/installing/) package to install cx\_Oracle from [PyPI](https://pypi.python.org/pypi/cx_Oracle):

python -m pip install cx\_Oracle --upgrade

This will download and install a pre-compiled binary [if one is available](https://pypi.python.org/pypi/cx_Oracle) for your architecture. If a pre-compiled binary is not available, the source will be downloaded, compiled, and the resulting binary installed.

[**Install Oracle Client**](https://cx-oracle.readthedocs.io/en/latest/installation.html#id18)

Using cx\_Oracle requires Oracle Client libraries to be installed. These provide the necessary network connectivity allowing cx\_Oracle to access an Oracle Database instance. Oracle Client versions 18, 12 and 11.2 are supported.

* If your database is on a remote computer, then download the free [Oracle Instant Client](http://www.oracle.com/technetwork/database/database-technologies/instant-client/overview/index.html)“Basic” or “Basic Light” package for your operating system architecture.
* Alternatively use the client libraries already available in a locally installed database such as the free [Oracle XE](https://www.oracle.com/database/technologies/appdev/xe.html) release.

[**Oracle Instant Client Zip Files**](https://cx-oracle.readthedocs.io/en/latest/installation.html#id19)

To use cx\_Oracle with Oracle Instant Client zip files:

1. Download an Oracle 18, 12, or 11.2 “Basic” or “Basic Light” zip file: [64-bit](http://www.oracle.com/technetwork/topics/winx64soft-089540.html) or [32-bit](http://www.oracle.com/technetwork/topics/winsoft-085727.html), matching your Python architecture.
2. Unzip the package into a directory that is accessible to your application. For example unzipinstantclient-basic-windows.x64-18.3.0.0.0dbru.zip to C:\oracle\instantclient\_18\_3.
3. Add this directory to the PATH environment variable. For example, on Windows 7, update PATHin Control Panel -> System -> Advanced System Settings -> Advanced -> Environment Variables -> System Variables -> PATH. The Instant Client directory must occur in PATH before any other Oracle directories.

Restart any open command prompt windows.

To avoid interfering with existing tools that require other Oracle Client versions, instead of updating the system-wide PATH variable, you may prefer to write a batch file that sets PATH, for example:

REM mypy.bat

SET PATH=C:\oracle\instantclient\_18\_3;%PATH%

python %\*

Invoke this batch file everytime you want to run python.

Alternatively use SET to change your PATH in each command prompt window before you run python.

1. Oracle Instant Client libraries require a Visual Studio redistributable with a 64-bit or 32-bit architecture to match Instant Client’s architecture. Each Instant Client version requires a different redistributable version:
   * For Instant Client 18 or 12.2 install [VS 2013](https://support.microsoft.com/en-us/kb/2977003#bookmark-vs2013)
   * For Instant Client 12.1 install [VS 2010](https://support.microsoft.com/en-us/kb/2977003#bookmark-vs2010)
   * For Instant Client 11.2 install [VS 2005 64-bit](https://www.microsoft.com/en-us/download/details.aspx?id=18471) or [VS 2005 32-bit](https://www.microsoft.com/en-ca/download/details.aspx?id=3387)
2. If you intend to co-locate optional Oracle configuration files such as tnsnames.ora, sqlnet.oraor oraaccess.xml with Instant Client, then create a network\admin subdirectory, for exampleC:\oracle\instantclient\_18\_3\network\admin.

This is the default Oracle configuration directory for executables linked with this Instant Client.

Alternatively, Oracle configuration files can be put in another, accessible directory. Then set the environment variable TNS\_ADMIN to that directory name.

[**Local Database or Full Oracle Client**](https://cx-oracle.readthedocs.io/en/latest/installation.html#id20)

cx\_Oracle applications can use Oracle Client 18, 12, or 11.2 libraries libraries from a local Oracle Database or full Oracle Client.

The Oracle libraries must be either 32-bit or 64-bit, matching your Python architecture.

1. Set the environment variable PATH to include the path that contains OCI.dll, if it is not already set. For example, on Windows 7, update PATH in Control Panel -> System -> Advanced System Settings -> Advanced -> Environment Variables -> System Variables -> PATH.

Restart any open command prompt windows.

1. Optional Oracle configuration files such as tnsnames.ora, sqlnet.ora or oraaccess.xml can be placed in the network/admin subdirectory of the Oracle Database software installation.

Alternatively, Oracle configuration files can be put in another, accessible directory. Then set the environment variable TNS\_ADMIN to that directory name.

[**Troubleshooting**](https://cx-oracle.readthedocs.io/en/latest/installation.html#id30)

If installation fails:

* Use option -v with pip. Review your output and logs. Try to install using a different method. **Google anything that looks like an error.** Try some potential solutions.
* Was there a network connection error? Do you need to see the environment variables http\_proxy and/or https\_proxy?
* Do you get the error “No module named pip”? The pip module is builtin to Python from version 2.7.9 but is sometimes removed by the OS. Use the venv module (builtin to Python 3.x) or virtualenv module (Python 2.x) instead.
* Do you get the error “fatal error: dpi.h: No such file or directory” when building from source code? Ensure that your source installation has a subdirectory called “odpi” containing files. If missing, review the section on [Install Using GitHub](https://cx-oracle.readthedocs.io/en/latest/installation.html#install-using-github).

If using cx\_Oracle fails:

* Do you get the error “DPI-1047: Oracle Client library cannot be loaded”?
  + Check that Python, cx\_Oracle and your Oracle Client libraries are all 64-bit or all 32-bit. The DPI-1047 message will tell you whether the 64-bit or 32-bit Oracle Client is needed for your Python.
  + On Windows, restart your command prompt and use set PATH to check the environment variable has the correct Oracle Client listed before any other Oracle directories.
  + On Windows, use the DIR command on the directory set in PATH. Verify that OCI.DLLexists there.
  + On Windows, check that the correct [Windows Redistributables](https://oracle.github.io/odpi/doc/installation.html#windows) have been installed.
  + On Linux, check the LD\_LIBRARY\_PATH environment variable contains the Oracle Client library directory.
  + On macOS, make sure Oracle Instant Client is in ~/lib or /usr/local/lib and that you are not using the bundled Python (use [Homebrew](https://brew.sh/) or [Python.org](https://www.python.org/downloads) instead).
* If you have both Python 2 and 3 installed, make sure you are using the correct python and pip (or python3 and pip3) executables.

**URL**

<https://cx-oracle.readthedocs.io/en/latest/installation.html#installing-cx-oracle-on-windows>

<http://ba6.us/?q=cx_Oracle_easy_windows_install>

<https://www.heatonresearch.com/content/oracle.html>

<https://stackoverflow.com/questions/11245985/easy-install-cx-oracle-python-package-on-windows>

<https://www.cs.utexas.edu/~scohen/cs327e_spr15/cx_Oracle/windows.html>