There are two types of site-packages directories, *global* and *per user*.

1. **Global** site-packages ("dist-packages") directories are listed in `sys.path` when you run:

   ```
   python -m site
   ```

   For a more concise list run `getsitepackages` from the site module in Python code:

   ```
   python -c 'import site; print(site.getsitepackages())'
   ```

   Note: With virtualenvs getsitepackages is not available, `sys.path` from above will list the virtualenv's site-packages directory correctly, though. In Python 3, you may use the sysconfig module instead:

   ```
   python3 -c 'import sysconfig; print(sysconfig.get_paths()["purelib"])'
   ```

2. The **per user** site-packages directory (PEP 370) is where Python installs your local packages:

   ```
   python -m site --user-site
   ```

   If this points to a non-existing directory check the exit status of Python and see `python -m site --help` for explanations.

   Hint: Running `pip list --user` or `pip freeze --user` gives you a list of all installed *per user*site-packages.

Practical Tips

- `<package>.__path__` lets you identify the location(s) of a specific package: (details)

  ```
  $ python -c "import setuptools as _; print(_.__path__)"
  ['/usr/lib/python2.7/dist-packages/setuptools']
  ```

- `<module>.__file__` lets you identify the location of a specific module: (difference)

  ```
  $ python3 -c "import os as _; print(_.__file__)"
  /usr/lib/python3.6/os.py
  ```

- Run `pip show <package>` to show Debian-style package information:

  ```
  $ pip show pytest
  Name: pytest
  Version: 3.8.2
  Summary: pytest: simple powerful testing with Python
  Home-page: https://docs.pytest.org/en/latest/
  Author: Holger Krekel, Bruno Oliveira, Ronny Pfannschmidt, Floris Bruynooghe, Brianna
  Laugher, Florian Bruhin and others
  Author-email: None
  License: MIT license
  Location: /home/peter/.local/lib/python3.4/site-packages
  Requires: more-itertools, atomicwrites, setuptools, attrs, pathlib2, six, py, pluggy
  ```