

Steps to follow:

Reference: <https://packaging.python.org/en/latest/tutorials/packaging-projects/>

How to Create a Python Library

This document provides a simple guide on how to create a basic Python library and distribute it via PyPI.

Step 1: Setup

- Ensure you have the latest versions of setuptools and wheel installed:

```
pip install --upgrade setuptools wheel
```

- Make sure you have a PyPI account: <https://pypi.org/account/register/>

Step 2: Project Structure

Create a folder structure for your project as follows:

```
your_library_name/  
  src/  
    your_library_name/  
      __init__.py  
      library_code.py  
  tests/  
  setup.py  
  README.md  
  LICENSE  
  .gitignore
```

- The `src` directory should contain your library code.
- The `tests` directory will contain test cases for your library.

Step 3: Writing the Library

- Write your library code inside the `src/your_library_name/` directory.
- The `__init__.py` file can be empty or contain code.

Step 4: Create README.md

- Write instructions and documentation for your library in the README.md file.
- This file will be shown as the library/project description on the pypi.

Step 5: Choose a License

- Choose a license for your library and include it in the LICENSE file.

Step 6: Writing setup.py

- Create a setup.py file with the necessary information about your library.

Example setup.py:

```
from setuptools import setup, find_packages

# Read the contents of your README file
with open("README.md", "r", encoding="utf-8") as fh:
    long_description = fh.read()

setup(
    name="alert_on_exception",
    version="0.1.5",
    package_dir={"": "src"}, # This line is added
    packages=find_packages(where="src"), # And this line is modified
    install_requires=[],
    python_requires=">=3.6",
    description="A Python package to alert users about exceptions.",
    # Short, concise description
    long_description=long_description, # Detailed description
    long_description_content_type="text/markdown", # Content type
    # for long_description, assuming it's in Markdown
)
```

Step 7: Building the Package

- Run the following command from the same directory where setup.py is located:

```
python setup.py sdist bdist_wheel
```

This command should output a lot of text and once completed should generate two files in the dist directory.

Step 8: Upload to PyPI

- First, install Twine:

```
pip install twine
```

- Then, run Twine to upload all of the archives under dist:

```
twine upload dist/*
```

Step 9: Installing Your Package from PyPI

- Once uploaded to PyPI, you can install your package with pip:

```
pip install your_library_name
```

Congratulations! You've packaged and distributed a Python library. 🎉