Steps to Create a Python Package

Creating a Python Package

Creating a Python package involves organizing your code into a directory structure and adding special files. Here's a step-by-step guide to help you create a simple Python package:

Step 1: Create the Package Directory

Start by creating a new directory for your package. This directory will contain your package modules and other necessary files.

```
"Yame of directory" #example : mkdir demommath cd "Name of directory" # cd demommath
```

```
x:~/Desktop/mypackage$ mkdir demommath
x:~/Desktop/mypackage/demommath$ mkdir addsub
x:~/Desktop/mypackage/demommath$ cd addsub
```

```
##Step 2: Create Sub Directory mkdir "Name of Sub directory" # mkdir addsub
```

Step 3: Create Modules

Inside your package directory, create Python modules (`.py` files) that will be part of your package. For example:

```
""python
# demommath/addsub/addas.py
def func1():
    print("Function 1")

# demommath/multidiv/md.py
def func2():
    print("Function 2")
```





```
Function to add two numbers a,b

def add(a, b):
    return a + b # Returns: The sum of a and b

# Function to subtract two numbers

def subtract(a, b):
    return a - b # Returns: The result of subtracting b from a
```

Step 4: Create `__init__.py`

In the main package directory, create an `__init__.py` file. This file can be empty, or you can use it to initialize your package.

```
```python
#demommath/__init__.py
```

In each package sub directory, create an `\_\_init\_\_.py` file. This file can be empty, or you can use it to initialize your package.

```
demommath/addsub/__init__.py```
```

To initialize package in the sub directory

```
```python
#demommath/addsub/__init__.py
from .addas import add
from .addas import subtract```
```

Step 5: Create `setup.py`

Create a 'setup.py' file to define metadata about your package, such as its name, version, and dependencies.

```
'``python
# demommath/setup.py
from setuptools import setup, find_packages
setup(
    name='demommath',
    version='0.1',
    packages=find_packages(),
    install_requires=[
          # List your dependencies here
    ],
)
```

```
/Desktop/mypackage
                                                                Q
from setuptools import setup, find_packages
setup(
    name='demommath',
    version='0.1.6',
    packages=find_packages(),
    .
long_description_content_type="text/markdown",
    long_description=open("README.md", "r", encoding="utf-8").read(),
    install_requires=[
    'numpy'
        # List your dependencies here
    entry_points={
         console_scripts': [
             'your-command-name=your_package.module1:main_function',
    author='Gangesh',
    author_email='gvelip@nio.org',
description='This package is for testing And How to create a package',
 setup.py" 23 lines, 596 characters
```

Step 6: Install setuptools

If you don't have `setuptools` installed, install it using:

```
```bash
pip install setuptools
```

# ## Step 7: Create a README.md file

This text file contains all the information about how to use the package.

# ## Step 8: Build and Install the Package

Navigate to the directory containing your 'setup.py' file and run the following commands:

```
```bash
python setup.py sdist
```

This Step creates build and dist directories.

To install locally,

pip install dist/demommath-0.1.4.tar.gz

...



Step 9: Verify

Now, you should be able to import and use your package in other Python scripts or environments:

```
"python >>>from demonmath import addsub >>>addsub.add(2,8)
```

This is a basic example. Depending on your package's complexity, you may need to add more details to your `setup.py` file. Refer to the [setuptools documentation] (https://setuptools.pypa.io/) for more options and customization.

Feel free to customize the package structure and `setup.py` according to your project's requirements.