

# Python 打包分发案例

AUTHOR: 彭玲 TIME: 2021/9/29

## Python 打包分发案例

setuptools 工具

案例 `setup_whl`

`setup.py` 安装文件

`devops` 自定义包源码

`__init__.py`

`foo.py`

打包 `whl` 格式

前置条件

打包 `whl` 文件

## setuptools 工具

作为 Python 标准的打包及分发工具，`setuptools` 可以说相当简单易用。`setuptools` 会随着 Python 一起安装在你的机器上。你只需写一个简短的 `setup.py` 安装文件，就可以将你的 Python 应用打包。

## 案例 `setup_whl`

`setup_whl` 项目组织结构如下：

```
1 | setup_whl
2 | └─ devops # 源代码
3 |   └─ foo.py
4 |     └─ __init__.py
5 | └─ setup.py # 安装文件
6 | └─ venv
```

## `setup.py` 安装文件

```
1 | #!/usr/bin/env python
2 | # -*- coding:utf-8 -*-
3 | # @author: julin
4 |
5 | from setuptools import setup
6 |
7 | setup(
8 |     name='julin-devops',
9 |     version='0.0.1',
10 |     description='Python automatic operation and maintenance platform',
```

```
11     author='julin',
12     author_email='julinpeng3@163.com',
13     url='https://github.com/julin',
14     packages=['devops'],
15 )
```

## devops 自定义包源码

### \_\_init\_\_.py

```
1  from . import foo
2
3  print('__file__={0:<20} | __name__={1:<20} | __package__={2:
  <20}'.format(__file__, __name__, str(__package__)))
4
5
6  def bar():
7      print('I am bar function.')
8
9
10 if __name__ == '__main__':
11     print("__name__ is __main__")
12     foo.say()
13 else:
14     print("__name__ is {0}".format(__name__))
15     bar()
```

### foo.py

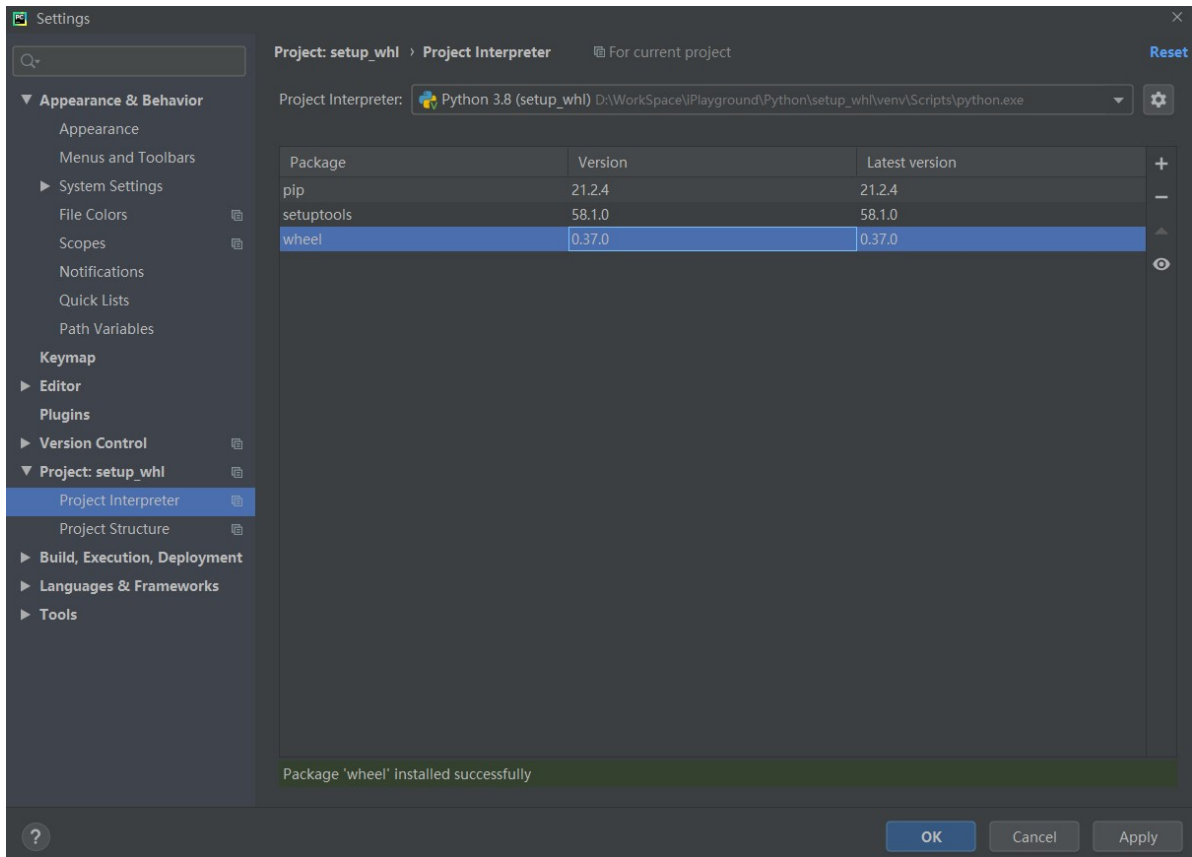
```
1  def say():
2      print("I am foo function.")
```

## 打包 whl 格式

---

### 前置条件

Python venv 虚拟环境下加入 wheel 包。



## 打包 whl 文件

```
1 | D:\Julin\iPlayground\whl_demo>python setup.py bdist_wheel
```

执行 `python setup.py bdist_wheel` 后, 本地 dist 文件夹下生成 whl 文件:

```
setup_whl/dist/julin_devops-0.0.1-py3-none-any.whl
```