

Lecture 13: Build Your Own Python Library

1. Python包的作用



为何要使用Python包？Python包与普通的文件夹有何不同？

1. 相同之处：

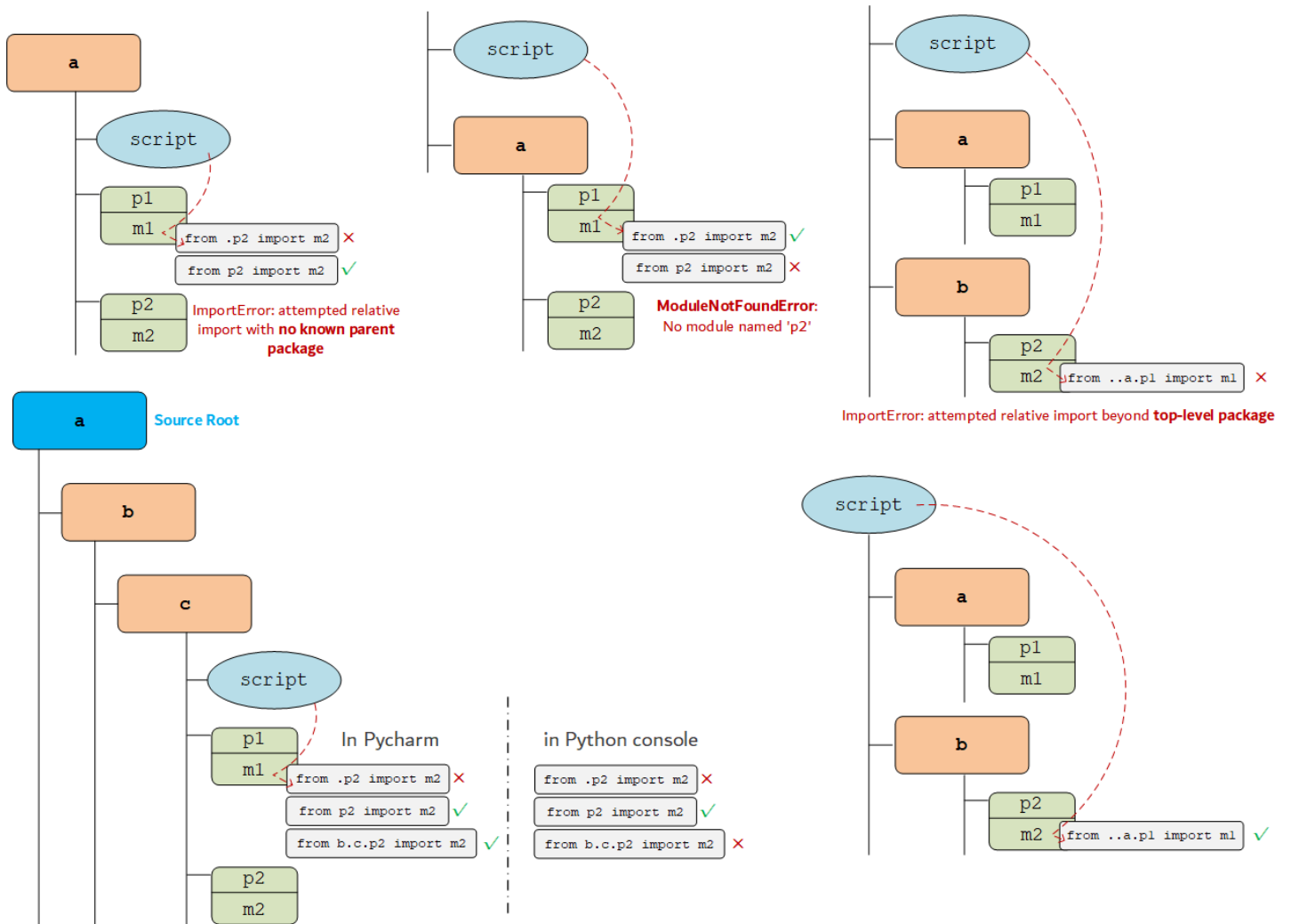
- a. Python包也是一个文件夹；
- b. 普通的文件夹与Python包一样，其中所定义的接口/类/函数，都可以被客户端程序 `import`。

2. 不同之处。它们在形式上最大的区别是：Python包中包含了一个 `__init__.py` 文件，或者说，只有包含了 `__init__.py` 的文件夹才是一个Python包。

3. Python包中的 `__init__.py` 的作用：

- a. 对Python包中所定义的对外公开的接口/类/函数等进行明确的“声明”。通常情况下，Python包是由很多个接口/类/函数构成的，但真正需要对外提供的只有一小部分，这一小部分可以认为是Python包的接口，写在 `__init__.py` 中统一“声明”，会使得Python包的作用和对外接口的定义更为明确；
- b. 简化对Python包中的接口/类/函数导入。Python包所定义的接口/类/函数有其自身的文件结构或包结构，并且通常是较深的树形结构，为了在客户端程序中导入这些接口/类/函数，如果不做处理，就需要比较长的包的前缀，这对于使用者来说，就容易记忆，也不友好。
- c. Python包中的 `__init__.py` 文件，很像class的 `__init__()` 方法：当我们初始化一个类的时候， `__init__()` 方法会被默认调用；当我们导入一个包的时候， `__init__.py` 文件中的代码也首先会被执行；比较特别的是，在一次运行一个Python项目时，Python包的 `__init__.py` 文件中的代码只会在第一次被导入时执行一次，随后的多次导入并不会再执行。因此，包所需要用的数据等可以放在 `__init__.py` 中加载。

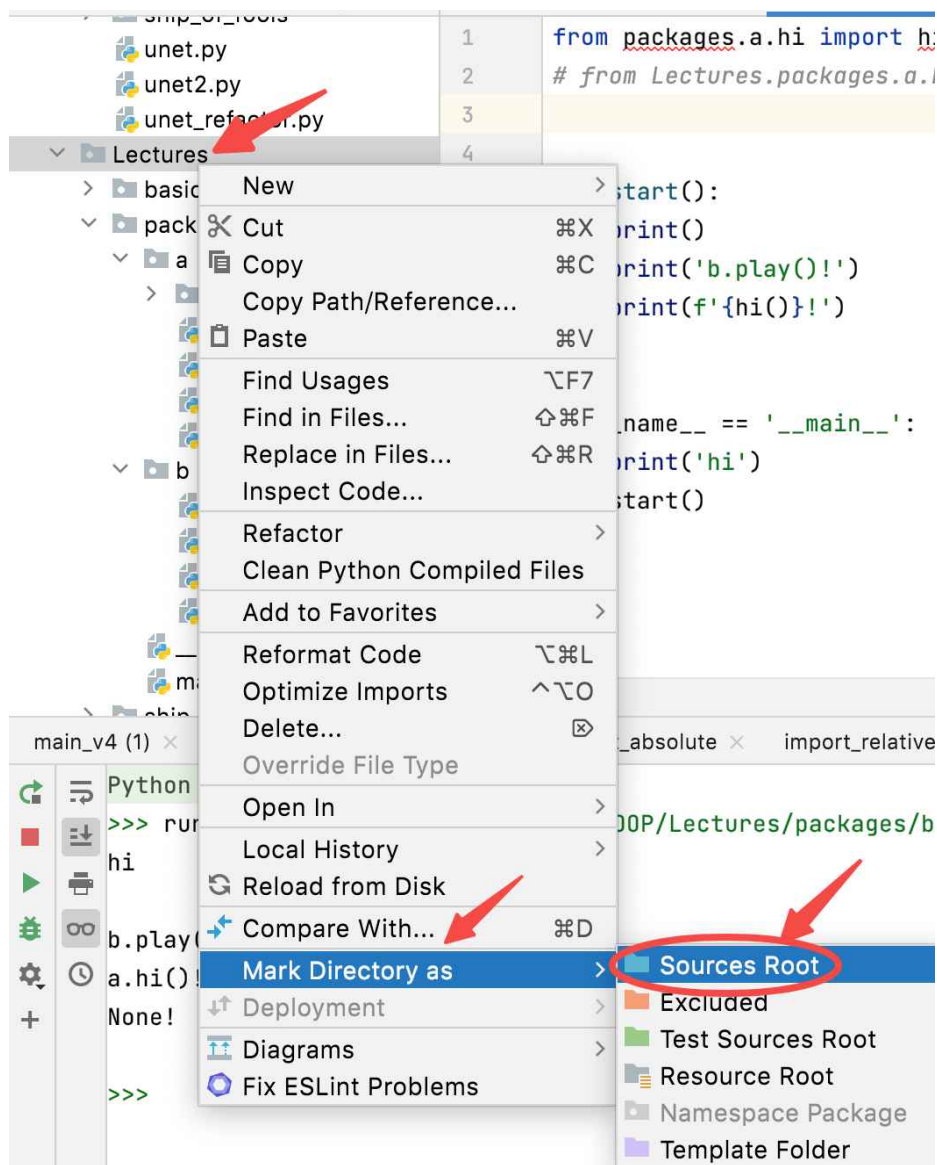
2. Absolute vs. Relative Imports



规则只有一个：当使用相对导入时，不能平级调用，只能降级调用。

2.1 在PyTest下测试上面的问题

2.2 在PyCharm中设置Source Root



2.3 在控制台下执行Python程序

```
1 $ python /Users/xxli/work/Python00P/Lectures/packages/b/import_absolute.py
2 Traceback (most recent call last):
3   File "/Users/xxli/work/Python00P/Lectures/packages/b/import_absolute.py", line
4     from Lectures.packages.a.hi import hi
5 ModuleNotFoundError: No module named 'Lectures'
```

3. Build Your Own Python Library

1. 为何要创建自己的Python库?
2. 如何创建?

```
1 from setuptools import setup, find_packages
2
3 # python setup.py install
```

```

4 # packages = find_packages('.') # , 'favourlab.img_recog*'
5 # packages = find_packages('.', exclude=['Lectures', 'Lectures.basics', 'Lectur
6 packages = find_packages('.', include=['Lectures.user_args']) # , 'favourlab.i
7
8 setup(
9     author='xxli',
10    author_email='mordecai@163.com',
11    name='NewFavourLib', # 将是我们所构建的 Python 库的名字
12    description='Python library written by xxli.',
13    url='https://github.com/mordeky',
14    version='1.0',
15    packages=packages,
16    # package_dir的说明, 详见: https://www.ucloud.cn/yun/45030.html
17    # package_dir={'Lectures.user_args': '../Lectures'},
18    package_dir={'Lectures.user_args': '.'},
19    # install_requires=[],
20    # exclude_package_data={
21    #     '': ['.gitignore', '.git'],
22    #     '': ['dist'],
23    #     '': ['.idea', '.pytest_cache', '_gsdata_'],
24    #     '': ['Trash', 'build', 'favourlab.egg-info'],
25    # },
26    # exclude_package_data={
27    #     '': ['adv_defense', 'dataset/fastMRI']
28    # }
29 )

```

3. 如何安装?

```
1 $ python setup.py install
```

查看安装好的包: 使用 `unzip` 解压即可查看。