

作业参考

复制目录

选择一个已存在的目录作为当前工作目录，在其下创建a/b/c/d这样的子目录结构并在这些子目录的不同层级生成50个普通文件，要求文件名由随机4个小写字母构成。

将a目录下所有内容复制到当前工作目录dst目录下，要求复制的普通文件的文件名必须是x、y、z开头。

举例，假设工作目录是/tmp，构建的目录结构是/tmp/a/b/c/d。在a、b、c、d目录中放入随机生成的文件，这些文件的名称也是随机生成的。最终把a目录下所有的目录也就是b、c、d目录，和文件名开头是x、y、z开头的文件。

```
1  from pathlib import Path
2  import shutil
3  from string import ascii_lowercase
4  import random
5
6  # 当前工作目录
7  basedir = Path('o:/temp')
8  subdir = Path('a/b/c/d')
9  dstdir = Path('dst')
10 dirs = (subdir, *subdir.parents)[:1]
11 print(dirs)
12
13 # 创建所有目录
14 (basedir / subdir).mkdir(parents=True, exist_ok=True)
15
16 # 随机文件名
17 filenames = ("".join(random.choices(ascii_lowercase, k=4)) for i in
18 range(50))
19
20 # 拼接路径生成文件
21 for name in filenames:
22     (basedir / random.choice(dirs) / name).touch()
23
24 heads = set('xyz')
25 def ignore_files(src, names):
26     #@return {name for name in names if name[0] not in heads and not
27 Path(src, name).is_dir()}
28     return set(filter(lambda name: name[0] not in heads and not Path(src,
29 name).is_dir(), names))
30
31 shutil.rmtree(str(basedir / dstdir), True)
32 shutil.copytree(str(basedir / 'a'), str(basedir / dstdir),
33 ignore=ignore_files)
34
35 # 遍历所有文件
36 print('-' * 30)
37 for f in (basedir / dstdir).rglob('*'):
38     print(f)
```

