

Copy Special Python Exercise

The Copy Special exercise goes with the file-system and external commands material in the [Python Utilities](#) section. This exercise is in the "copyspecial" directory within google-python-exercises (download [google-python-exercises.zip](#) if you have not already, see [Set Up](#) for details). Add your code in copyspecial.py.

The copyspecial.py program takes one or more directories as its arguments. We'll say that a "special" file is one where the name contains the pattern `__w__` somewhere, where the w is one or more word chars. The provided main() includes code to parse the command line arguments, but the rest is up to you. Write functions to implement the features below and modify main() to call your functions.

Suggested functions for your solution(details below):

- `get_special_paths(dir)` -- returns a list of the absolute paths of the special files in the given directory
- `copy_to(paths, dir)` given a list of paths, copies those files into the given directory
- `zip_to(paths, zippath)` given a list of paths, zip those files up into the given zipfile

Part A (manipulating file paths)

Gather a list of the absolute paths of the special files in all the directories. In the simplest case, just print that list (here the "." after the command is a single argument indicating the current directory). Print one absolute path per line.

```
$ ./copyspecial.py .  
  
/Users/nparlante/pycourse/day2/xyz__hello__.txt  
  
/Users/nparlante/pycourse/day2/zz__something__.jpg
```

We'll assume that names are not repeated across the directories (optional: check that assumption and error out if it's violated).

Part B (file copying)

If the "--todir dir" option is present at the start of the command line, do not print anything and instead copy the files to the given directory, creating it if necessary. Use the python module "shutil" for file copying.

```
$ ./copyspecial.py --todir /tmp/fooby .  
  
$ ls /tmp/fooby
```

```
xyz__hello__.txt      zz__something__.jpg
```

Part C (calling an external program)

If the "--tozip zipfile" option is present at the start of the command line, run this command: "zip -j zipfile <list all the files>". This will create a zipfile containing the files. Just for fun/reassurance, also print the command line you are going to do first (as shown in lecture). (Windows note: windows does not come with a program to produce standard .zip archives by default, but you can get download the free and open zip program from www.info-zip.org.)

```
$ ./copyspecial.py --tozip tmp.zip .

Command I'm going to do:zip -j tmp.zip
/Users/nparlante/pycourse/day2/xyz__hello__.txt

/Users/nparlante/pycourse/day2/zz__something__.jpg
```

If the child process exits with an error code, exit with an error code and print the command's output. Test this by trying to write a zip file to a directory that does not exist.

```
$ ./copyspecial.py --tozip /no/way.zip .

Command I'm going to do:zip -j /no/way.zip
/Users/nparlante/pycourse/day2/xyz__hello__.txt

/Users/nparlante/pycourse/day2/zz__something__.jpg


zip I/O error: No such file or directory

zip error: Could not create output file (/no/way.zip)
```

Except as otherwise noted, the content of this page is licensed under the [Creative Commons Attribution 3.0 License](https://creativecommons.org/licenses/by/3.0/), and code samples are licensed under the [Apache 2.0 License](https://www.apache.org/licenses/LICENSE-2.0). For details, see our [Site Policies](#). Java is a registered trademark of Oracle and/or its affiliates.

Last updated December 13, 2012.