

## `__init__.py`

- The `__init__.py` file is required to make python treat a directory of python files as a package.
- You need to mark a directory as a package like this explicitly, otherwise directories with common names could unintentionally hide a package occurs later in python's search path.
- Many times `__init__.py` will be empty, but it can contain package initialization code (that will be executed upon import) or set the special `__all__` variable. More details are [here](#)
- `init.py` makes a folder a package.\
- For example, here is an example project:

Python module

```
|-----> setup.py
|-----> calendermod
           |-----> __init__.py
           |-----> day.py
           |-----> month.py
           |-----> year.py
           |-----> decade.py
```

If I were to try `import module calendermod` without the `__init__.py` file, it would fail. It would say that it is not a module.

- `__init__.py` is executed when you use an **import** statement.
- `__main__.py` is executed when you run the module.