

Перед выполнение последующих шагов вам необходимо сделать приготовления:

- Из кода необходимо убрать все обращения к `__file__`, иначе в собранный пакет будет выдавать ошибку, что данная переменная не определена. Например, вместо `dir_path = os.path.dirname(os.path.realpath(__file__))` используйте `dir_path = os.getcwd()`.
- Вместо обычного вызова `exit(arg)` нужно использовать `sys.exit(arg)`.
- В скрипты `database.py` сервера и клиента надо добавить импорт: `from sqlalchemy import default_comparator`.
- Проверьте, что у вас установлен пакет `wheel`. Если нет, то его нужно установить.
- Не забудьте установить утилиту `sx_Freeze`.

Для разработанного проекта «Мессенджер» сформировать whl-пакеты с дистрибутивами сервера и клиента.

- а) Необходимо разделить наш проект на собственно сервер и клиент. Каждую из этих двух логических частей нужно привести к виду пакета, т.е. директории с файлом `__init__.py`. В этой директории должны находиться все необходимые папки и файлы. В итоге у вас будут две директории – `server` и `client`.
- б) Далее необходимо создать файлы `setup.py` и прописать в нем нужные настройки.

Пример для серверного файла `setup.py`:

```
from setuptools import setup, find_packages

setup(name="mess_server",
      version="0.8.7",
      description="mess_server",
      author="Ivan Ivanov",
      author_email="iv.iv@yandex.ru",
      packages=find_packages(),
      install_requires=['PyQt5', 'sqlalchemy', 'pycryptodome', 'pycryptodomex']
)
```

Пример для клиентского файла `setup.py`:

```
from setuptools import setup, find_packages

setup(name="mess_client",
      version="0.8.7",
      description="mess_client",
      author="Ivan Ivanov",
      author_email="iv.iv@yandex.ru",
      packages=find_packages(),
      install_requires=['PyQt5', 'sqlalchemy', 'pycryptodome', 'pycryptodomex']
)
```

- в) Выполнить сборку whl-пакетов командой **`python setup.py sdist bdist_wheel`**

Выполнить загрузку сформированных whl-пакетов с дистрибутивами сервера и клиента в репозиторий сервиса PyPi.

- а) Зарегистрируйтесь на `pypi.org`
- б) После регистрации необходимо выждать сутки.

- c) Скопируйте файл .ruirc и разместите в вашей домашней директории, например Диск C: -> Пользователи -> Дмитрий.
- d) Измените настройки в файле .ruirc и пропишите свои логин и пароль, указанные на pypi.org.

```
[distutils] # this tells distutils what package indexes you can push to
index-servers =
    pypi
    pypitest

[pypi]
repository: https://pypi.python.org/pypi
username: cdi999
password: YXQPp5iUF!sF6Pz
```

- e) Перейдите в директории с пакетами и для каждого выполните команду

python setup.py sdist upload

Выполнить процедуру сборки созданного проекта «Мессенджер» с помощью утилиты cx_Freeze (создание исполняемых файлов)

- a) Настройте еще одну пару файлов-конфигов, которым можно дать названия, например, setup_server.py и setup_client.py.

Пример для серверного файла setup_server.py:

```
import sys
from cx_Freeze import setup, Executable

build_exe_options = {
    "packages": ["common", "logs", "server", "unit_tests"],
}

setup(
    name="mess_server",
    version="0.8.7",
    description="mess_server",
    options={
        "build_exe": build_exe_options
    },
    executables=[Executable('server.py',
                            # отключаем появление консоли
                            base='Win32GUI',
                            targetName='server.exe',
                            )]
)
```

Пример для серверного файла setup_client.py:

```
import sys
from cx_Freeze import setup, Executable

build_exe_options = {
    "packages": ["common", "logs", "client", "unit_tests"],
}

setup(
    name="mess_client",
    version="0.8.7",
    description="mess_client",
    options={
        "build_exe": build_exe_options
    },
    executables=[Executable('client.py',
                            # отключаем появление консоли
                            base='Win32GUI',
                            targetName='client.exe',
                            )]
)
```

b) Выполните сборку для сервера и клиента, воспользовавшись командами

python setup_server.py build_exe и **python setup_client.py build_exe**

c) cx_freeze некорректно обрабатывает SQLAlchemy, поэтому после подготовки дистрибутива нужно вручную заменить папку sqlalchemy в папке lib на аналогичную папку из директории интерпретатора.

d) Перейдите в директорию lib -> PyQt5 -> Qt -> plugins. Нужно вырезать папку platforms и перенести в директорию, где находится exe-файл.

Python 3.7 + PyQt 5