

П'ятий етап виконання курсової роботи

5. Аналіз отриманих результатів. Підготовка дистрибутиву. Оформлення звіту та документації по розробленому програмному забезпеченню. Підготовка презентаційних матеріалів до виконаної роботи.

	Завдання	Результат виконання завдання
5.1	Аналіз отриманих результатів.	Текстовий документ 2 сторінки А4 10 шрифт
5.2	Створення дистрибутиву програми за допомогою вбудованої утиліти Python Distribution Utilities ("Distutils")	Дистрибутив програми
5.3	Оформлення документації по розробленому програмному забезпеченню. Автоматична генерація документації засобами середовища розробки.	Текстовий документ в форматі pdf або html.
5.4	Запис відео презентації виконаної роботи. Приклади таких презентаційних відео можна побачити наприклад за наступним посиланням http://www.cs.cmu.edu/~112/gallery.html	Відео файл в форматі *.avi тривалістю до 3х-хвилин
5.5	Висновки до курсової роботи	Текстовий документ 1 сторінка А4 10 шрифт

5.2 Повний опис використання Python Distribution Utilities (“Distutils”) можна знайти в документації <https://docs.python.org/3/distutils/index.html>. Українською мовою процес підготовки пакету докладно описано в https://uk.wikibooks.org/wiki/Пориньте_у_Python_3/Пакування_бібліотек. Коротко процес підготовки дистрибутиву можна побачити на наступних слайдах:

РОЗПОВСЮДЖЕННЯ PYTHON ПРОГРАМ ТА БІБЛІОТЕК

- ▶ **Модуль distutils (вбудований)**
 - ▶ Дозволяє підготувати дистрибутив програми
 - ▶ На комп’ютері користувача, який буде використовувати цей дистрибутив повинен бути встановлений інтерпретатор Python
- ▶ **Модуль py2exe, cx_freeze, pyinstaller (сторонні розробки)**
 - ▶ Дозволяють підготувати виконуваний модуль (exe файл), який не вимагає наявності інтерпретатора Python



МОДУЛЬ distutils

- ▶ Підготувати структуру папок
 - ▶ Створити файл README,
 - ▶ Скопіювати в папку документацію і файли з сирцевим кодом.
- ```
spam/
 README.txt
 Documentation.txt
 libspam.py # Окремий модуль
 spampkg/ # Пакет допоміжних модулів
 __init__.py
 foo.py
 bar.py
 runspam.py # Сценарій (код який можна
 запустити як: python runspam.py
```
- ▶ Створити файл setup.py в кореневій папці
- ```
# setup.py  
from distutils.core import setup  
setup(name = "spam",          # ім'я пакета  
      version = "1.0",        # версія пакета  
      py_modules = ['libspam'], # список всіх файлів окремих модулів  
      packages = ['spampkg'],  # список всіх папок з пакетами  
      scripts = ['runspam.py'], # список файлів сценаріїв  
      )
```



МОДУЛЬ distutils

- ▶ Створення дистрибутиву - в командному рядку виконати
 - ▶ `python setup.py sdist`
 - ▶ В папці `spam/dist` буде створено файл архіву (`spam-1.0.tar.gz` або `spam-1.0.zip`).
 - ▶ `python setup.py bdist` - створення двійкового дистрибутиву (замість `.py` будуть файли `.pyc`)
 - ▶ `python setup.py bdist_wininst` - буде створено `.exe` файл і при його запуску буде запущений майстер встановлення.
- ▶ Встановлення програми – в командному рядку виконати
 - ▶ `unzip spam-1.0.zip`
 - ▶ `cd spam-1.0`
 - ▶ `python setup.py install`



5.3 Оформлення документації по розробленому програмному забезпеченню.
Автоматична генерація документації засобами середовища розробки.

В розробці програмного забезпечення важливе місце займає створення документації програмного продукту. Для автоматизації і спрощення цього процесу існують спеціальні програми для автоматичної генерації документації. Для створення документації проектів, які розроблені на Python рекомендується використовувати Sphinx (<http://www.sphinx-doc.org>). Sphinx це потужний засіб для генерації Python документації але розроблені і інші засоби, які дозволяють отримати подібні результати.

За наступними посиланнями можна знайти перелік засобів для автоматичної генерації документації та рекомендації по її використанню.

<https://wiki.python.org/moin/DocumentationTools>

<http://docs.python-guide.org/en/latest/writing/documentation/>

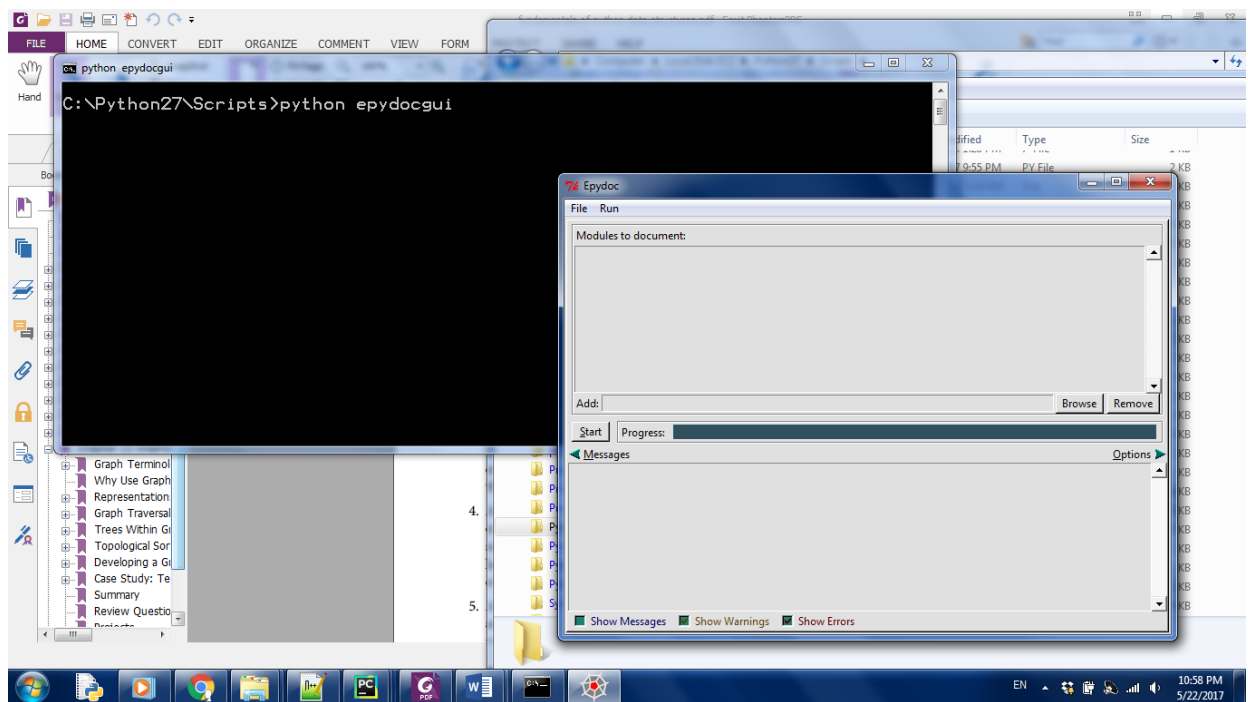
https://docs.readthedocs.io/en/latest/getting_started.html#in-rst

Наступний приклад демонструє створення документації до курсової роботи за допомогою засобу (<http://epydoc.sourceforge.net/>) за умови що всі модулі, класи, методи та функції містять рядки документації. **Увага! Epydoc не підтримує Python3 а вимагає Python 2.7 і тому рекомендується використовувати Sphinx.**

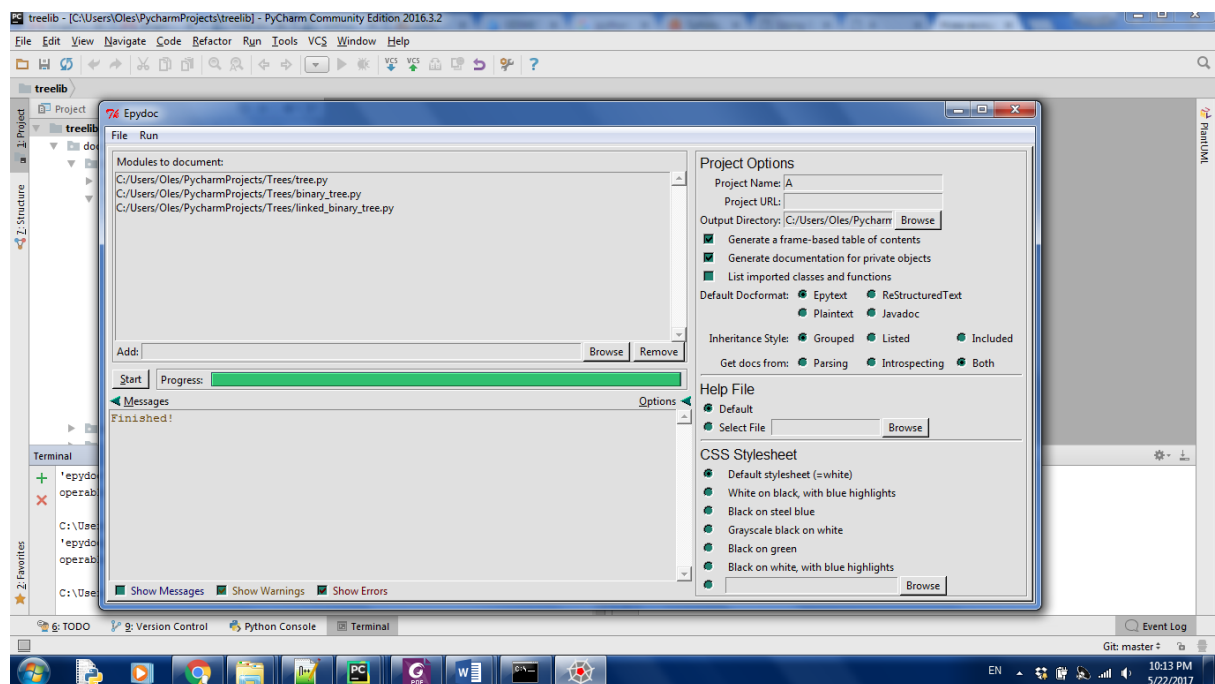
Для автоматичної генерації документації достатньо встановити Epydoc згідно інструкції користувача <http://epydoc.sourceforge.net/manual-install.html> і в командному рядку або за допомогою графічного інтерфейсу запустити процес генерації.

Встановлення: `pip install epydoc`.

Графічний інтерфейс: `python epydocgui`



У вікно Modules to document потрібно додати всі модулі, які входять в проект. У вікні Project Option вказати шлях до папки docs проекту та вказати назву проекту.



Після запуску (Start) процесу створення документації у папці docs проекту буде автоматично створено документацію до проекту в форматі html.

Browser tabs: M Відкриті, M Відкриті, First Str, GitHub, CMS, GISMET, python, Sphinx, Using E, A, Нова вкладка, Андрій.

Address bar: file:///C:/Users/Oles/PycharmProjects/Trees/index.html

Bookmarks: Додатки, ма, LSA, NewMarks, NewBook, Programing_coding, Social_Media, Sentiment, Каадроконтр, English_study, PM, ірця, IS_Design, Olya_edu, Інші закладки.

Table of Contents

- Everything
- Modules
 - [binary_tree](#)
 - [linked_binary_tree](#)
 - [tree](#)
- [hide private]
- Everything
- All Classes
 - [binary_tree.BinaryTree](#)
 - [linked_binary_tree.LinkBinaryTree](#)
 - [linked_binary_tree.LinkBinaryTreePo](#)
 - [linked_binary_tree.LinkBinaryTree_N](#)
 - [tree.Tree](#)
 - [tree.Tree Position](#)
- All Variables
 - [binary_tree.__package__](#)
 - [linked_binary_tree.__package__](#)
 - [tree.__package__](#)
- [hide private]

Trees Indices Help

[Module Hierarchy | Class Hierarchy]

Module Hierarchy

- [binary_tree](#)
- [linked_binary_tree](#)
- [tree](#)

Trees Indices Help

Generated by Epydoc 3.0.1 on Mon May 22 22:06:43 2017

<http://epydoc.sourceforge.net>

Taskbar: Основи програму....zip, Основи програму....zip, BST.rar, Trees_easy.zip, Trees.zip, Показати все

System tray: EN, 10:14 PM, 5/22/2017