***Призначення та коротка характеристика програми***

Програма, яку я розробила для виконання завдання дослідження, призначена для того, щоб проаналізувати деякі характеристики 50-ти найкращих серіалів.

Дослідження, отримані моєю програмою, можуть стати особливо цікавими для людей, що захоплюються кінематографом.

По-суті, програма виконує три дослідження, кожне з яких аналізує певну інформацію, використовуючи масив із назвами серіалів та дані, які надає TheTVDB API:

* Перше дослідження базується на тому, що TheTVDB API дає можливість за назвою серіалу отримати список всіх акторів/акторок, що в ньому знімалися.

Використовуючи ці дані, я вирішила зробити дослідження того, які актори найчастіше знімалися у серіалах, що стали дуже популярними та чи загалом є багато акторів, що знімалися більше, ніж в одному популярному серіалі.

В результаті першого дослідження ми отримаємо своєрідний рейтинг, в якому буде вказане ім’я та прізвище актора/акторки й число, яке позначатиме, у скількох серіалах знявся цей актор чи ця акторка.

* Друге дослідження пов’язане із кількістю сезонів у серіалах.

Оскільки TheTVDB API дає точну інформацію про те, скільки сезонів має кожен серіал, моя програма робить дослідження того, які із найпопулярніших серіалів мають найбільше сезонів. В результаті цього дослідження ми отримаємо рейтинг, у якому буде вказана назва серіалу та кількість сезонів в ньому.

* Третє дослідження спрямоване на те, щоб дослідити чи у назвах популярних серіалів повторюються певні слова.

В результаті цього дослідження буде отримано рейтинг, який покаже які слова і скільки разів траплялися у назвах найкращих 50 серіалів (артиклі, сполучники та прийменники також туди входитимуть, тому це прийдеться врахувати під час аналізу виконаних досліджень).

Всі дані, отримані із досліджень, будуть збережені у текстові файли. При чому, всі рейтинги будуть збережені в однаковому форматі (кожна позиція в рейтингу з нового рядка, спочатку буде записане число, а після нього через дефіс дані, дослідження яких робилося у конкретному випадку).

***Вхідні та вихідні дані програми***

**Вхідні дані програми** – перелік найкращих 50 серіалів всіх часів. Дані взяті із сайту *http://www.empireonline.com* .

Ці дані проходять кілька етапів змін. Спочатку вони зчитуються у список, в якому кожен елемент – інформація про інший серіал. Потім цей список обробляється і перетворюється на список, який містить тільки назви серіалів. Для проведення дослідження створюється одновимірний масив, у який і записуються ці дані.

Детальніше про дані для дослідження описано у файлі *Information about the data.txt*.

**Вихідні дані програми** – три текстові файли (popular\_actors.txt , series\_seasons.txt та series\_titles.txt), в яких збережені результати досліджень:

* popular\_actors.txt : представляє кількість серіалів, у яких знімався певний актор (знімалась акторка) та ім’я і прізвище актора/акторки (рейтинг від більшої кількості до меншої, представлені тільки ті актори, які знімались більше, ніж у одному серіалі).
* series\_seasons.txt : представляє кількість сезонів у серіалі та назву цього серіалу (рейтинг від більшої кількості до меншої).
* series\_titles.txt : представляє слово та кількість його повторів у назвах даних серіалів.

***Структура програми з коротким описом модулів, функцій, класів та методів***

Головний модуль програми – my\_research.py . Він, в свою чергу, імпортує класи WorkWithAFile та SeriesResearch із модулів work\_with\_a\_file.py та series\_research.py відповідно.

Модуль **work\_with\_a\_file.py** реалізовує клас WorkWithAFile, що містить наступні методи:

* open\_file(self): забезпечує зчитування інформації із файлу ratings.txt. Повертає список, кожен елемент якого – це всі інформація про конкретний серіал, яка міститься у файлі.
* make\_list(self, lst): атрибут lst – це інформація, яку повертає метод open\_file(self). На основі отриманих даних створює список, в якому елементами є назви серіалів. Повертає створений список.

Модуль **series\_research.py** – це реалізація SeriesResearch ADT. Він імпортує методи для обробки інформації про серіали, які надає TheTVDB API. Також він імпортує клас AnArray із модуля an\_array.py, що є реалізацією такої структури даних як масив. Клас SeriesResearch містить такі методи:

* В методі \_\_init\_\_(self, length) створюється масив із довжиною length.
* get\_item(self, index): дозволяє доступитися до елемента масиву за заданим його індексом, повертає значення цього елемента;
* set\_item(self, index, value): дозволяє присвоїти елементу масиву з індексом index значення value;
* get\_actors(self, index): бере з масиву елемент з індексом index (це назва серіалу) та створює список, у який додає всіх акторів, що знімалися в даному фільмі; повертає цей список.
* seasons\_number(self, index): бере з масиву елемент з індексом index (це назва серіалу) та знаходить кількість сезонів у даному серіалі; повертає знайдене число.

У модулі **my\_research.py** міститься реалізація класу MainResearch:

* В методі \_\_init\_\_(self) створюється екземпляр класу SeriesResearch, тобто масив, елементами якого є назви серіалів.
* В методі \_\_str\_\_(self) створюється стрічка, в яку через кому записуються назви серіалів.
* popular\_actors(self): створює словник, ключами в якому є імена акторів, що знімалися у серіалах, назви яких ми одержали із зчитаного файлу ratings.txt, а значеннями – кількість серіалів, у яких вони знімалися. Повертає цей словник.
* series\_seasons(self): створює словник, у якому ключі – це назви серіалів, а значення – кількість сезонів у них. Повертає цей словник.
* series\_titles(self): створює словник, ключі в якому – це всі слова, які були у назвах заданих серіалів, а значення – кількість повторень цих слів. Повертає словник.

Також у модулі **my\_research.py** міститься реалізація функцій save\_info(info, filename) та main():

* save\_info(info, filename): info – це список кортежів, з якого буде зчитуватися інформація, filename – назва файлу, в який буде зберігатися інформація. Зберігає дані із info у файл filename.
* main(): функція, яка виконує основні дослідження. Викликає методи класу MainResearch. Сортує інформацію, отриману з цих методів, а потім викликає функцію save\_info(info, filename) зберігає інформацію про кожне дослідження в окремі файли.

Вкінці модуля **my\_research.py** міститься виклик функції main(), що забезпечує виконання дослідження при запуску даного модуля.

***Коротка інструкція по користуванню програмою***

Оскільки суть моєї програми в тому, щоб зробити дослідження, вона не вимагає введених даних від користувача.

Щоб користуватися моєю програмою на даному етапі, ви можете склонувати мій проект coursework2017 ( <https://github.com/victoria-yuzkiv/coursework2017> ).

Якщо ви хочете детальніше прочитати опис дослідження, API, яке я використовувала, приклад його використання, функціональності пакету XML, інформацію про дані, які я використовувала, то можете знайти все це в директорії **doc**.

У директорії **examples** містяться приклади використання API, яке я використовувала та використання абстрактного типу даних (SeriesResearch ADT в моєму випадку).

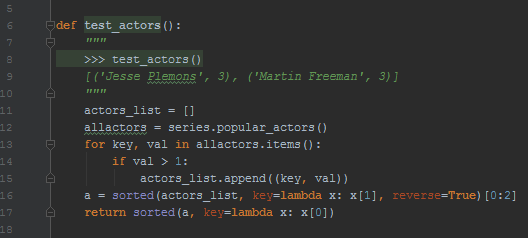
У директорії **modules** містяться всі основні модулі, які відповідають за реалізацію досліджень. Всі класи, методи та функції мають документацію, тому за бажанням можете з ними ознайомитись.

Якщо хочете дізнатися результати дослідження – відкрийте модуль ***my\_research.py*** та запустіть його виконання. Після цього в директорії modules з’являться файли popular\_actors.txt , series\_seasons.txt та series\_titles.txt. Ці файли представляють дані, які і є результатами досліджень.

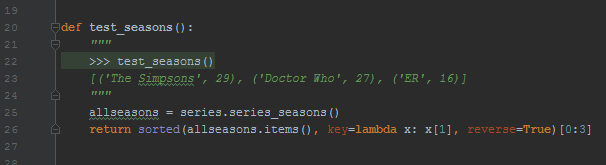
***Опис тестових прикладів для перевірки працездатності програми***

Я розробила 3 тести для перевірки кожного дослідження:

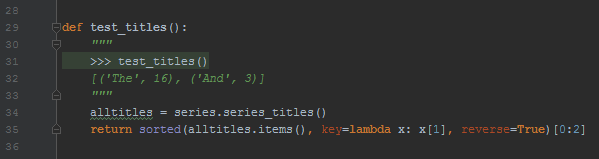
* test\_actors() – для перевірки працездатності методу popular\_actors(self). Я вибрала перші два значення (відсортовані за алфавітом) й перевірила, чи отримаю бажаний результат, викликавши цей метод:



* test\_seasons() – для перевірки працездатності методу series\_seasons(self). Я вибрала перші три значення (відсортовані за кількістю сезонів) й перевірила, чи отримаю бажаний результат, викликавши цей метод:



* test\_titles() – для перевірки працездатності методу series\_titles(self). Я вибрала перші два значення (відсортовані за частотою з’являння слів у назвах серіалів) й перевірила, чи отримаю бажаний результат, викликавши цей метод:



Виконавши дані доктести, я отримала наступний результат:



Отже, програма працює правильно.