

## Task: Crearea programului pentru gestionarea filmelor

Curs: Data Management and Storage with Python

Modul: Datele în Python

### Scopul sarcinii:

Creați un program Python care permite utilizatorilor să creeze o listă de filme de vizionat.

### Contextul sarcinii:

Vi s-a atribuit această sarcină pentru a crea un program care permite utilizatorilor să creeze o listă de filme care merită vizionate (*watchlist*, *engl.*). Aceasta este, de fapt, o listă în care utilizatorii vor putea adăuga filme interesante pe care doresc să le vizioneze în viitor. Programul este menit să fie un fel de memento pe care utilizatorii îl pot folosi pentru a nota filmele pe care doresc să le vizioneze într-o zi.

### Sarcină:

Programul vostru ar trebui să permită utilizatorilor:

- **introducerea** unui nou film;
- **vizualizarea** filmelor existente.

Fiecare film trebuie să conțină următoarele informații:

- titlul filmului (`title`);
- anul lansării filmului (`release_year`);
- genul (unul sau mai multe genuri, separate prin virgule) (`genre`);
- adresa URL de pe site-ul IMDB (`imdb_url`).

Când rulează programul, utilizatorul ar trebui să poată alege ce vrea să facă. Utilizatorului ar trebui să i se ofere posibilitatea:

- **să introducă** detaliile noului film;
- **să vizualizeze** toate filmele introduse anterior;
- **să părăsească** programul.

Selectând opțiunea de **introducere** a unui nou film, va trebui să obțineți datele filmului (titlu, gen, an etc.) de la utilizator și apoi să le salvați în program.

Prin selectarea opțiunii de **vizualizare**, utilizatorului ar trebui să i se afișeze toate filmele introduse anterior. Dacă nu există filme introduse anterior, utilizatorul ar trebui să fie informat despre acest lucru.

Selectând opțiunea de **părăsire** a programului, programul ar trebui să afișeze un mesaj de rămas bun utilizatorului și să termine execuția.

Dacă utilizatorul selectează opțiunile de introducere sau vizualizare, după finalizarea acestora, utilizatorului trebuie să i se afișeze din nou lista de opțiuni posibile. În acest fel, utilizatorul ar trebui să poată introduce câte filme dorește la rând.

Puteți vedea cum ar putea arăta toate acestea în practică în imaginea care ilustrează un caz de utilizare a programului, în care utilizatorul introduce mai întâi date pentru trei filme, apoi vizualizează toate filmele introduse și, în cele din urmă, părăsește programul.

```
PROBLEMS  OUTPUT  DEBUG CONSOLE  TERMINAL  PORTS

*****WELCOME TO MOVIE WATCHLIST APP*****
Add new movie(1)
Show All movies(2)
Exit(3)
1
Movie title: The Godfather
Movie release year: 1972
Movie genre: Crime, Drama
Movie IMDB URL: https://www.imdb.com/title/tt0068646/
Add new movie(1)
Show All movies(2)
Exit(3)
1
Movie title: Interstellar
Movie release year: 2014
Movie genre: Adventure, Drama, Sci-Fi
Movie IMDB URL: https://www.imdb.com/title/tt0816692/
Add new movie(1)
Show All movies(2)
Exit(3)
1
Movie title: Inception
Movie release year: 2010
Movie genre: Action, Adventure, Sci-Fi
Movie IMDB URL: https://www.imdb.com/title/tt1375666/
Add new movie(1)
Show All movies(2)
Exit(3)
2
The Godfather, 1972
Crime, Drama
https://www.imdb.com/title/tt0068646/

Interstellar, 2014
Adventure, Drama, Sci-Fi
https://www.imdb.com/title/tt0816692/

Inception, 2010
Action, Adventure, Sci-Fi
https://www.imdb.com/title/tt1375666/

Add new movie(1)
Show All movies(2)
Exit(3)
3
Have a nice day!
```

*Imaginea 1.1. Unul dintre posibilele cazuri de utilizare a programului*

Desigur, toate datele pe care utilizatorul le introduce în program ar trebui

să fie salvate permanent chiar și după ce programul este închis.

## Trimiterea sarcinii:

După finalizarea sarcinii, trebuie să-i trimiteți coordonatorului cursului codul sursă al proiectului complet sau folderul rădăcină al proiectului, pe care trebuie mai întâi să îl arhivați într-o arhivă (.rar, .zip, .7z).

## Cerințe de evaluare:

Coordonatorul va evalua sarcina pe baza următoarelor criterii:

1. Introducerea corectă a datelor utilizând funcțiile `input()` **(10%)**.
2. Modelarea corectă a conceptului „Movie” (clase și obiecte) **(20%)**.
3. Efectuarea corectă a serializării **(20%)**.
4. Efectuarea corectă a deserializării **(20%)**.
5. Controlul corect al fluxului prin care utilizatorii vor controla execuția programului **(20%)**.
6. Calitatea codului și lizibilitatea **(10%)**.

## Ghid pentru rezolvarea sarcinii:

Mai jos sunt specificați pașii care vă pot ajuta în realizarea sarcinii:

1. **Modelarea conceptului Movie** – în primul rând, trebuie creată o clasă `Movie` cu toate proprietățile bazate pe proprietățile filmelor care trebuie stocate în program. Clasa `Movie` ar trebui să aibă, de asemenea, o metodă de conversie a obiectelor de tip `Movie` în string (metoda `__str__()`).
2. **Crearea controlului fluxului** – deoarece trebuie permisă execuția unui număr arbitrar de operații în timpul rulării programului, cel mai bine este să folosiți o buclă `while` în cadrul căreia operațiile pe care

utilizatorul le poate efectua vor fi repetate ciclic. Pe lângă o astfel de buclă, utilizatorul trebuie să aibă posibilitatea de a alege una dintre cele trei operații: introducere, afișare sau ieșire. O astfel de selecție este cel mai bine realizată folosind structura `if-elif-else` sau `match-case`.

3. **Crearea logicii pentru introducerea unui nou film (serializarea)** – de la utilizator trebuie preluate datele despre film, folosind funcția `input()` și apoi trebuie să se creeze un obiect de tip `Movie` pe baza acestor date. Obiectul creat trebuie salvat într-un fișier. Acest lucru este cel mai ușor de făcut folosind modulul `pickle`. Cu toate acestea, dacă găsiți în mod independent o modalitate de a efectua serializarea, soluția respectivă va fi valabilă atâta timp cât programul face ceea ce i se cere să facă.
4. **Crearea logicii pentru afișarea filmelor introduse anterior (dezerializarea)** – trebuie să citiți datele filmelor salvate anterior (desigur, dacă există date introduse anterior). Din nou, cel mai bine este să folosiți `pickle` și să deserializați pur și simplu datele despre film scrise anterior. Apoi, iterând prin lista tuturor filmelor, trebuie să afișați datele acestora.
5. **Crearea logicii pentru ieșirea din program** – trebuie să permiteți utilizatorului să iasă din program prin introducerea unei anumite comenzi (indiferent că este un număr special, caracter sau cuvânt). În acest caz, trebuie să opriți bucla `while` creată anterior și să listați un mesaj de rămas bun utilizatorului.