

SEMINAR 9

Aplicație Gestionare Seriale (extindere aplicație anterioară pentru a lucra cu mai multe entități).

Serial: id, denumire, an aparitie, numar episoade

Client: id, nume, varsta

Rating: serial, client, number of stars

TO DO:

STATISTICI/RAPOARTE (DTOs + RepoRatingInMemory) [SEMINAR8_211_1_week9.zip]

1. Pentru un client dat, să se afișeze serialele cu cele mai bune rating-uri (primele 3). Se va afișa numele clientului, numele serialului, și rating-ul (number_of_stars).
2. Să se afișeze numărul de rating-uri pentru fiecare serial și media de varsta a clientilor care au evaluat serialul. Se va afișa titlul serialului, numărul de rating-uri, media rating-ului (average number of stars) și media de vârstă.

SHOW FILE REPO - MOSTENIRE [SEMINAR9_211.zip]

1. Sa se implementeze clasa ShowFileRepo, varianta cu moștenire (repo care permite salvare/incarcare din fisier, si care mosteneste din clasa ShowRepoInMemory)
2. **RepoClientFile, RepoRatingFile, varianta cu mostenire** - rapoarte 1,2: DTOs + RepoRatingFile

UNIT TESTING

3. Sa se scrie teste folosind modulul **unittest**.

- import unittest
- Creare clasa de test derivata din unittest.TestCase
- Metode speciale: se executa inainte (setUp) si dupa (tearDown) fiecare metoda de test
 - setUp
 - tearDown
- Metode de test: numele metodelor trebuie inceapa cu **test** ca sa fie identificate ca teste
- Simple asserts inlocuite in clasa de test cu:
 - (self).assertEqual
 - (self).assertTrue, assertFalse
 - (self).assertRaises
 - ◆ [Sintaxa apel assertRaises](#)
 - More like assertIn, assertNotIn, assertNotEqual, assertIsInstance, etc:
<https://docs.python.org/3/library/unittest.html>

Extra (nu se cere la lab-uri/examene): code coverage

How much of the code (i.e. what percent of the written lines) did my tests cover?

PyCharm Professional: [Running with coverage](#)

PyCharm Community Edition: [PyCharm Community Edition Coverage Tools](#)

Eclipse + PyDev: [Unit test coverage results with Eclipse + PyDev](#)