

## SEMINAR 3

### TO COVER:

- Funcții
- Specificațiile funcțiilor
- Scriere de teste și procesul TDD

Creați o aplicație cu interfață de tip consolă care gestionează o listă de cântece. Fiecare cântec are patru atribute:

**titlu:** titlul melodiei (str, cel puțin 2 caractere)  
**artist:** artistul melodiei (str, cel puțin 1 caracter)  
**gen:** genul melodiei (str, poate fi doar unul dintre: *rock, pop, hip-hop, folk*)  
**durata:** durata melodiei, sub forma mm.ss (float; numărul de minute trebuie să fie între 1 și 15, iar numărul de secunde între 1 și 59)

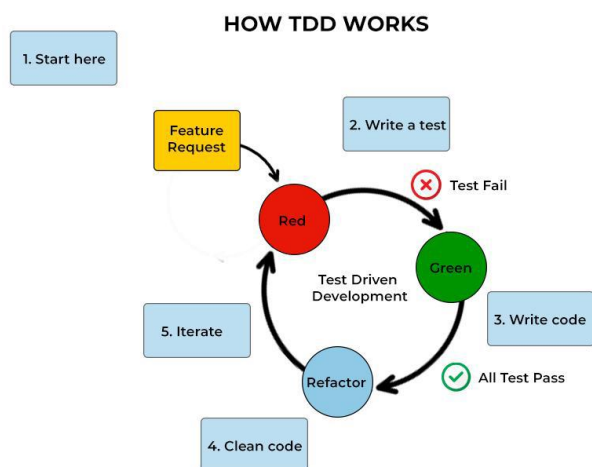


Aplicația permite următoarele:

1. Adăugarea unui cântec în listă. Informațiile pentru cele patru câmpuri se citesc de la tastatură.
2. Căutarea unui cântec după titlu și artist.
3. Ștergerea unui cântec după titlu și artist.
4. Eliminarea tuturor cântecelor dintr-un anumit gen din listă.
5. Afișarea cântecelor care au durată între două durate date de la tastatură.
6. Afișarea unor statistici cu privire la lista de cântece
  - a) Numărul de cântece care conțin un string dat de la tastatură în titlu.
  - b) Numărul de cântece ale fiecărui artist (dintre cei existenți în listă).
  - c) Numărul de cântece din fiecare gen (dintre toate cele 4 menționate; dacă nu există cântec dintr-un anumit gen, se va afișa valoarea 0).
7. Undo pentru operațiile care modifică lista.

Se va implementa și posibilitatea de printare a listei de cântece și de adăugare a unor cântece default. De asemenea, datele introduse cu privire la cântece se vor valida.

Q:



- 1) Cum reprezentăm datele din problemă?
- 2) Cum implementăm soluția pentru această problemă urmând un proces TDD? La ce este util TDD?
- 3) Cum implementăm operația de undo?
- 4) Cum implementăm validarea datelor?
  - a) Validare input
  - b) Validare în cadrul listei

## THE PLAN:

SEMINAR 3		Funcții, funcții de test, TDD
SEMINAR 4		More TDD, Modularizare
SEMINAR 5		Aspecte legate de scrierea codului ușor testabil, ușor modificabil

Pentru toate funcționalitățile identificate:

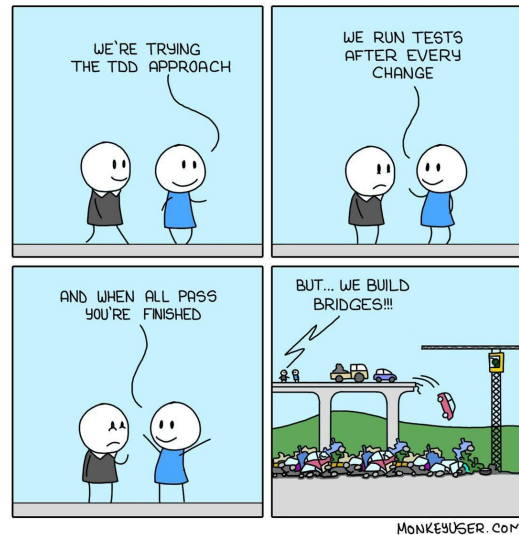
**Modelare** → **Activități** → **Cazuri de test (funcții de test)**

**Modelare:** scenarii de rulare

**Activități:** tabel de task-uri

**Teste:** input-expected output, funcții cu assert

## APPLIED TDD



R:

(1) Răspuns exercițiu de asociere Seminar1+2:

1-E  
2-J  
3-D  
4-K

5-A  
6-L  
7-B, F  
8-G

9-I  
10-H

(2) Recapping exercises

A. Ce se printează în urma execuției următoarelor secvențe de cod?

```
a = [1, 2, 4, 10]
b = a
a[2] = 'hello!'
print(b)
```

A.1. \_\_\_\_\_

```
a = [1, 2, 4, 10]
b = a
b[0] = 'hello!'
print(a)
```

A.2. \_\_\_\_\_

```
lst = [-3, 6, 1, 2, 4, 10, 18, 23]
print(lst[2:6])
```

A.3. \_\_\_\_\_

```
print(['a'] * 5)
```

A. 4. \_\_\_\_\_

```
lst = list(range(5, 0, -1))
print(lst)
```

A. 5. \_\_\_\_\_

```
lst1 = ['Ana are']
lst2 = ['mere']
lst1.extend(lst2)
print(lst1)
```

A. 6. \_\_\_\_\_

```
t = (1, 2, 3, 5)
a, b, c, d = t
print(c)
```

A.7. \_\_\_\_\_

```
numbers = [1, 19, 20, -5]
print(numbers[-1])
```

A.8. \_\_\_\_\_

```
participants = ('A', 'B', 'V', 'E')
print('E' in participants)
```

A.9. \_\_\_\_\_

```
lst = [1, [9], 2, 20, 9]
lst.remove(9)
print(lst)
```

A.10. \_\_\_\_\_

B. Se dă funcția de mai jos împreună cu specificația sa. Completați codul lipsă (sunt 6 locuri de completat) astfel încât testele (instrucțiunile `assert`) să treacă cu succes.

```
def count_character_frequencies(sentence: 1) -> 2:
    """
        Funcția calculează frecvența caracterelor dintr-o propoziție dată
        :param sentence: propoziția dată
        :return un dicționar care conține perechi de caracter-frecvență
    """
    frequency_dict = 3
    for character in sentence.4:
        if character in frequency_dict:
            frequency_dict[5] += 1
        else:
            frequency_dict[character] = 6
    return frequency_dict

test_sentence1 = "Ana are mere."
freq_dict_test1 = count_character_frequencies(test_sentence1)
assert (freq_dict_test1['a'] == 3)
assert (freq_dict_test1[' '] == 2)

test_sentence2 = ""
freq_dict_test2 = count_character_frequencies(test_sentence2)
assert (len(freq_dict_test2) == 0)
```