

FestivalManager – Laboratoare

- **Laborator 1: Output, Tipuri de Valori, Functii**

- Output: Folosit in 'Main.java' pentru afisarea meniului si rezultatelor prin 'System.out.println'.
- Tipuri de valori: Utilizare tipuri 'int', 'double', 'String' in toate clasele, precum 'Artist.java', 'Bilet.java'.
- Functii: Toate metodele definite in clasa, cum ar fi 'afiseazaArtisti()' in 'FestivalService.java'.

- **Laborator 2: Input, For, While, Switch, If**

- Input: Folosit in 'Main.java' pentru citirea datelor utilizand 'Scanner'.
- For: Utilizat in 'FestivalService.java' in metoda 'adaugaBilete' pentru a adauga bilete.
- While: Utilizat in 'Main.java' pentru implementarea meniului principal.
- Switch: Utilizat in 'Main.java' pentru navigarea intre optiuni.
- If: Folosit in 'Bilet.java' pentru verificarea starii unui bilet.

- **Laborator 3: Colectii Java**

- List: Utilizat in 'FestivalService.java' pentru gestionarea listei de artisti si bilete.
- Map: Utilizat in 'FestivalService.java' in metoda 'afiseazaBilete' pentru gruparea biletelor dupa tip si pret.

- **Laborator 4: Clase Java**

- Clase create: 'Artist', 'Bilet', 'Organizator', 'FestivalService', 'Main'.
- Atribute si metode: Utilizate in toate clasele, exemplu: 'nume', 'genMuzical' si metode ca 'afiseazaRol()'.

- **Laborator 5: Mostenire si Clase Abstracte**

- Mostenire: 'Artist' si 'Organizator' mostenesc clasa abstracta 'Persoana'.

- Clasa abstracta: `'Persoana'` definita cu metoda abstracta `'afiseazaRol()'`.

- **Laborator 6: Interfete in Java**

- Interfata: `'Performant.java'` implementata in clasa `'Artist'`.

- **Laborator 7: Teste pentru fiecare metoda**

- Testele au fost implementate in `'FestivalServiceTest.java'` folosind JUnit.

- Exemple: `'testAdaugareBilete()'`, `'testVanzareBileteSucces()'`.

- **Laborator 8: Persistenta datelor**

- Persistenta datelor: Implementata in `'FestivalService.java'` prin metodele `'incarcaDateDinFisier()'` si `'salveazaDateInFisier()'` folosind biblioteca `'Gson'`.

- **Laborator 9: Diagrame UML**

- Diagrame UML: Generate automat de extensia din IntelliJ – PlantUML Integration.