

დავალბების დაწყებამდე გაეცანით multicast delegate-ს და delegate chain-ს

### 1. Consolidated Logging

- a. შექმენით LogMessage დელეგატირომელიც მიიღებს string message-ს და არ ექნება return value
- b. განსაზღვრეთ ორი ტიპის ლოგირების ქცევა: კონსოლში დაბეჭდვის და ფაილში ჩაწერის
- c. Multicast დელეგატის და delegate chain-ის გამოყენებით გააერთიანეთ ეს ორი ლოგირების მეთოდი
- d. გამოიძახეთ multicast delegate, რომლის შემდეგ უნდა დაილოგოს მესიჯი, როგორც კონსოლში, ასევე ფაილში.

### 2. Sequential Calculations

- a. შექმენით MathOperation ტიპის დელეგატი, რომელიც მიიღებს ორ decimal-ს და დააბრუნებს decimal-ს
- b. შექმენით მარტივი კალკულაციის მეთოდები: დამატება, გამოკლება, გამრავლება და გაყოფა
- c. შექმენით სხვადასხვა delegate chain-ები
- d. გამოიძახეთ ეს დელეგატები და დაბეჭდეთ მიღებული შედეგები

### 3. Chained Validation

- a. შექმენით book ტიპის კლასი შემდეგი property-ებით
  - i. Title (აუცილებელი, მხოლოდ ასოები,  $1 < \text{სიგრძე} < 255$ )
  - ii. Author (აუცილებელი, მხოლოდ ასოები,  $3 < \text{სიგრძე} < 128$ )
  - iii. ISBN (მხოლოდ რიცხვი, სიგრძე 13 ციფრი)
  - iv. Publisher (აუცილებელი,  $2 < \text{სიგრძე} < 64$ )
  - v. PublicationDate (არ არის აუცილებელი, წარსულში უნდა იყოს)
  - vi. Genre (enum გამოიყენეთ, აუცილებელი ველია)
  - vii. NumberOfPages (აუცილებელი,  $>0$ )
  - viii. IsAvailable (აუცილებელია)
  - ix. Price (არ არის აუცილებელი, თუ არის მაშინ  $>0$ )
- b. შექმენით Validator ტიპის დელეგატი, რომელიც მიიღებს Book-ს და დააბრუნებს bool ტიპს.
- c. შექმენით ვალიდაციები ზემოთ აღწერილი წესების მიხედვით, თუ გინდათ შეცვალოთ კოდეც
- d. შექმენით სხვადასხვა delegate chain ამ ვალიდაციებით

e. დააბრუნეთ ყველა ის ველი, რომელიც არ არის ვალიდური

#### 4. Custom Data Pipelines

a. დააგენერირეთ წიგნების list

b. დელეგატების გამოყენებით გაფილტრეთ list რომელიმე ველის/ველების მიხედვით (გამოცემული რომელიმე წლის მერე, X გვერდზე ნაკლები, Y ჟანრის და ა.შ.)

c. დელეგატის გამოყენებით გააკეთეთ Book-ების ტრანსფორმაცია სხვა ტიპის ობიექტში BookDto:

i. Title

ii. Author

iii. Genre

iv. IsAvailable

v. Price

d. **მინიშნება:** შექმენით `DataPipeline<T>`, რომელსაც ექნება ფილტრების და ტრანსფორმაციების delegate chain, ასევე `TResult Process(IEnumerable<T> input)` მეთოდი, რომელიც გააკეთებს ყველა ოპერაციას.