Skrevet af Casper E. Jensen

Opgave: Udvikling af et Ejendomsselskabs Administrationssystem - 2

Formål:
Formålet med denne udvidelse er at introducere brugen af interfaces, en simpel brugergrænseflade (menu) i programmet, og en metode til at gemme og hente data fra en tekstfil. Programmet skal kunne håndtere ejendomme via et fælles interface og give brugeren mulighed for at interagere med systemet gennem en menu til at tilføje ejendomme og lejere samt vise eksisterende ejendomme.

OBS: Opgave C er en ekstra opgave hvis du har mod på den.

Opgavebeskrivelse

Forarbejde: Udarbejdelse af en Domænemodel

- Udarbejd en domænemodel, der viser entiteter som Lejer, Bolig, Lejlighed, Villa, Rækkehus og EjendomsPorteføljе.

Del A: Introduktion af et Interface for Boliger

1. **Opgave A1: Opret et interface Bolig**
    - Opret et interface kaldet Bolig, som de tre boligtyper (Lejlighed, Villa, Rækkehus) skal implementere.
    - Interfacet skal indeholde følgende metode-signaturer:
        - getAdresse(), getAreal(), getVærdi(), getUdståendeGæld(), getLejer().
        - setAdresse(), setAreal(), setVærdi(), setUdståendeGæld(), setLejer().
        - toString(): returnerer en beskrivende tekststreng med boligens information.
2. **Opgave A2: Implementér interface Bolig i boligklasserne**
    - Implementér interfacet i klasserne Lejlighed, Villa og Rækkehus.
    - Hver af de tre klasser skal indeholde de samme minimumsegenskaber: adresse, areal, værdi, udestående gæld og lejer.
    - Du kan tilføje ekstra egenskaber for hver boligtype.

Del B: Implementér en Menu i main()

3. **Opgave B1: Implementér en simpel menu**
    - Implementér en menu i main(), hvor brugeren kan:
        - Se alle ejendomme i porteføljen.
        - Tilføje en ny ejendom (lejlighed, villa eller rækkehus).
        - Tilføje en lejer til en eksisterende ejendom.
        - Afslutte programmet.
4. **Opgave B2: Håndtering af ejendomsportefølje**
    - Implementér en klasse kaldet EjendomsPorteføljе, der indeholder en liste over Bolig-objekter.
    - Tilføj metoder til at:
        - Tilføje en bolig (tilføjBolig).
        - Fjerne en bolig (fjernBolig).
        - Liste alle boliger (listAlleBoliger).

Del C: Lagring af data til og fra en tekstfil

5. **Opgave C1: Implementér en "repo"-klasse til at gemme og hente data**
    - Opret en klasse kaldet BoligRepo, som skal håndtere gemning og indlæsning af boliger fra en tekstfil.
    - Implementér følgende metoder:
        - gemBoligerTilFil(String filnavn, List<Bolig> boliger): Gemmer alle boliger fra porteføljen i en tekstfil.
        - hentBoligerFraFil(String filnavn): Henter boliger fra en tekstfil og returnerer dem som en liste af Bolig-objekter.
    - Sørg for, at filen kan indeholde data om både adresser, areal, værdi, udestående gæld og lejeren for hver bolig. Håndtér også specifikke boligattributter som etager eller have.
    - Brug simple tekstformater, hvor hver bolig gemmes som en linje i filen, og attributterne adskilles af fx kommaer eller semikolon.
6. **Opgave C2: Tilpas menuen til at bruge BoligRepo**
    - Tilføj funktionalitet i menuen til at:
        - Gemme alle nuværende boliger til en fil, når brugeren vælger det.
        - Hente boliger fra en fil, når programmet starter.

Aflevering og Vurdering

- **Aflevering:** Opgaven skal afleveres ved at uploade koden til et GitHub-repository (offentligt eller privat). Inkludér en README-fil, der beskriver, hvad der gik godt, og hvad der kunne forbedres i opgaven.
- **Vurdering:** Opgaven vurderes på korrekt implementering af de beskrevne funktioner, kodens strukturering samt brug af objektorienterede principper. Dokumentation og kodekvalitet vil også blive vurderet.
-