

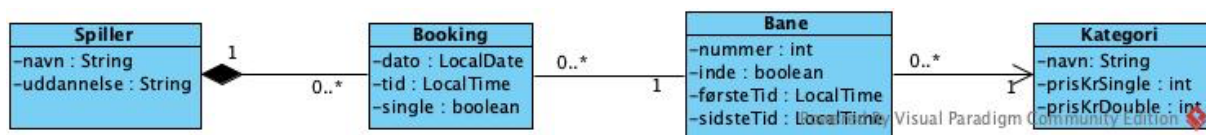
Semesterprøve Programmering juni 2023 (100 points)

Denne opgave omhandler et system, der skal holde styr på bookinger af padel baner på erhvervsakademiets kommende (planlagte) anlæg med padel baner.

En bane kan bookes hver dag fra og med en første tid til og med en sidste tid. Nogle af banerne er indendørs, andre er udendørs. Banerne er inddelt i forskellige kategorier (efter beliggenhed og udstyr), som har forskellige priser, og forskellige priser for singlespil (2 spillere) og doublespil (4 spillere).

En booking afgives af en spiller til en bane på en bestemt dato og tid. En booking er altid på præcis 1 time og starter på et helt timetal og er for enten singlespil eller doublespil. Betaling for bookingen er ikke en del af dette system.

Figuren nedenfor viser et simpelt designklassediagram for systemet. Bemærk, at linkattributter og klassernes metoder ikke er vist på diagrammet.



Opgave S1 (22 point)

Programmér klasserne *Spiller*, *Booking*, *Bane* og *Kategori* i pakken *model*. Klasserne skal have attributter med tilhørende get-metoder, og en konstruktor, som initialiserer alle attributter. Lav ikke set-metoder, men tilføj en set-metode senere, hvis du får brug for den.

Programmér sammenhængen mellem klasserne – både linkattributter og metoder til at opdatere dem. En metode til at fjerne en booking fra en spiller skal ikke laves. Multipliciteter og retninger fremgår af figuren (bemærk, at der er to dobbeltrettede sammenhænge).

Opgave S2 (5 point)

Tilføj til klassen *Bane* metoden *bookedeTimerPåDato(LocalDate dato): int*, der returnerer det samlede antal timer, banen er booket på den givne dato.

Opgave S3 (7 point)

Spillere vil gerne vide, hvad de har betalt for brug af baner i en bestemt kategori. Tilføj til klassen *Spiller* en metode *samletPris(Kategori kategori)*, der returnerer den samlede pris for bookinger af baner i den givne kategori foretaget af spilleren.

Opgave S4 (8 point)

Akademiet er interesseret i at vide, hvornår på dagen de forskellige baner bliver brugt. Tilføj til klassen *Bane* en metode *antalBookningerPrTime()*, der returnerer et *array* med antal bookninger for hver time banen er åben.

Opgave S5 (4 point)

Tilføj klassen *Storage* i pakken *storage*. Klassen skal indeholde lister med systemets spillere, baner og kategorier. Klassen skal også indeholde metoder til at gemme objekter fra klasserne *Spiller*, *Bane* og *Kategori*, og metoder til at hente alle spillere, baner og kategorier. Lav ikke metoder til at slette objekterne fra *Storage*.

Opgave S6 (8 point)

Tilføj klassen *Controller* i pakken *controller*. Klassen skal indeholde metoder til at oprette systemets objekter.

Tilføj klassen *App* med en *main()* metode i pakken *gui*. Klassen skal have en metode *initStorage()*, der under anvendelse af opret-metoderne opretter og gemmer data svarende til nedenstående:

Kategorier:	Navn: <i>Luksus</i>	Single/double pris: 100/200 kr.
	Navn: <i>Mellem</i>	Single/double pris: 50/100 kr.
	Navn: <i>Begynder</i>	Single/double pris: 25/50 kr.

Baner: Der er 6 baner med numrene 1-6. Bane 1-3 er indendørs og bane 4-6 er udendørs. Alle baner kan bookes fra kl. 9 til kl. 17. Bane 1 og 2 er Luksus baner, bane 3 og 4 er Mellem baner, og bane 5 og 6 er Begynder baner.

Spillere:	Navn: <i>Andreas</i>	Uddannelse: <i>DMU</i>
	Navn: <i>Petra</i>	Uddannelse: <i>DMU</i>
	Navn: <i>Henrik</i>	Uddannelse: <i>ITA</i>
	Navn: <i>Ulla</i>	Uddannelse: <i>ITA</i>

Bookinger: Andreas har booket bane 3 den 20/6-23 kl. 10 til single spil og bane 2 den 22/6-23 kl. 10 til doublespil.
Henrik har booket bane 3 den 20/6-23 kl. 11 til double spil.
Ulla har booket bane 3 den 20/6-23 kl. 16. til double spil og bane 5 den 23/6-23 kl. 17 til single spil.

Opgave S7 (8 point)

Tilføj til klassen *ControlLer* metoden

samletBooketDoubleTid(String uddannelse, Kategori kategori): int

Metoden skal returnere den samlede tid for bookinger til double spil foretaget af spillere på den givne uddannelse på baner af den givne kategori.

Opgave S8 (4 point)

Tilføj til klassen *Bane* en metode *tidLedig*. Metoden skal tage en dato og en tid som parametre og returnere true eller false.

Bemærk: Det er et krav at metoden implementeres under anvendelse af søgeskabelonen. *break* må ikke bruges, og metoden må kun indeholde én *return* sætning.

Opgave S9 (5 point)

Tilføj til klassen *ControlLer* en metode *findLedigBane*. Metoden skal tage en dato, en tid og en kategori som parametre og søge efter en ledig bane med den givne kategori på det angivne tidspunkt. Metoden skal returnere den ledige bane, hvis en sådan findes, ellers returneres *null*.

Bemærk: Det er et krav at metoden implementeres under anvendelse af søgeskabelonen. *break* må ikke bruges, og metoden må kun indeholde én *return* sætning.

Opgave S10 (7 point)

Tilføj en metode til klassen *Controller*. Metoden skal i en tekstfil udskrive alle bookings af baner. Udskriften skal se ud som vist herunder (for objekterne oprettet i *initStorage()*):

```
2 Bane nr: 3, dag: 2023-06-20 kl. 10:00, spil: single, spiller: Andreas
  Bane nr: 3, dag: 2023-06-22 kl. 10:00, spil: double, spiller: Andreas
  Bane nr: 3, dag: 2023-06-20 kl. 11:00, spil: double, spiller: Henrik
  Bane nr: 3, dag: 2023-06-20 kl. 16:00, spil: double, spiller: Ulla
  Bane nr: 5, dag: 2023-06-23 kl. 17:00, spil: single, spiller: Ulla
```

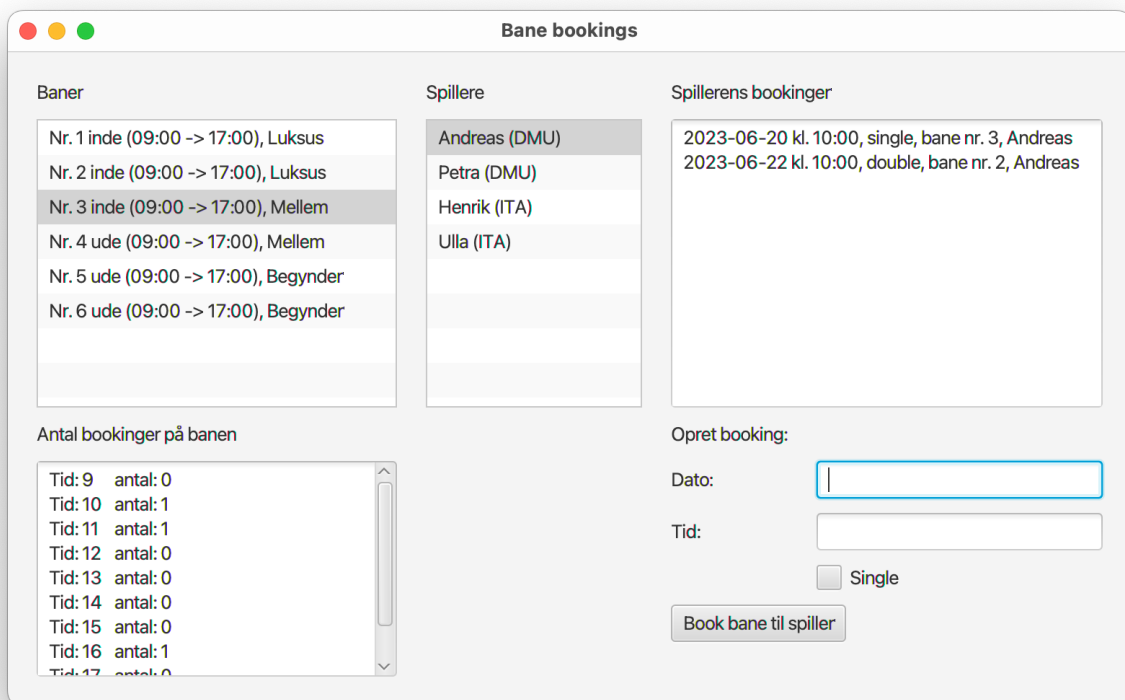
Metoden skal som parameter have navnet på den tekstfil, der skal skrives til. Relevante exceptions skal fanges i selve metoden og resultere i en fejludskrift.

Opgave S11 (22 point)

Programmér en grænseflade i pakken *gui* under anvendelse af JavaFX, så følgende use cases kan udføres under anvendelse af controller, model og storage:

- Vise en liste over alle baner (i et ListView).
- Vise en banes antal bookinger fordelt på dagens tidspunkter (i et TextArea).
- Vise en liste over alle spillere (i et ListView).
- Vise alle bookinger af en bane foretaget af en spiller (i et TextArea).
- Booke en bane til en spiller på en given dag og tid (husk at opdatere visningen af bookinger). Hvis banen ikke er ledig, skal der vises et lille vindue med en fejlmeddelelse.

Et vindue der indeholder ovenstående kunne for eksempel se ud som vist i figuren herunder.



Du skal aflevere én zip fil på Wiseflow indeholdende pakkerne med alle de programmerede klasser (kun pakkerne, ikke src-mappen). Filen skal navngives med dit **fornavn**.