

Databaseprosjekt del 2

For å bruke applikasjonen kjører du `__main__.py` eller alternativt kan man teste funksjonene med unittestene i `tests.py`.

- a) Databasen skal kunne registrere data om alle jernbanestrekninger i Norge. Dere skal legge inn data for Nordlandsbanen (som vist i figuren). Dette kan gjøres med et skript, dere trenger ikke å programmere støtte for denne funksjonaliteten.
- b) Dere skal kunne registrere data om togruter. Dere skal legge inn data for de tre togrutene på Nordlandsbanen som er beskrevet i vedlegget til denne oppgave. Dette kan gjøres med et skript, dere trenger ikke å programmere støtte for denne funksjonaliteten.

Applikasjonen bruker SQL-skriptene

`init_togdb_tables.sql` og `init_togdb_data.sql`

for å lage tabellene og å legge inn data om Nordlandsbanen.

Dette gjøres automatisk når applikasjonen kjøres.

- c) For en stasjon som oppgis, skal bruker få ut alle togruter som er innom stasjonen en gitt ukedag.
Denne funksjonaliteten skal programmeres.

Kommandoen `finn_ruter` implementerer dette:

```
Skriv kommando: finn_ruter
```

```
Dag: mandag
```

```
Stasjon: Trondheim
```

```
Ruter på mandag fra Trondheim:
```

```
    Rute #0 operert av SJ går kl. 07:49
```

```
    Rute #1 operert av SJ går kl. 23:05
```

```
    Rute #2 operert av SJ går kl. 14:13
```

```
Skriv kommando:
```

- d) Bruker skal kunne søke etter togruter som går mellom en startstasjon og en sluttstasjon, med utgangspunkt i en dato og et klokkeslett. Alle ruter den samme dagen og den neste skal returneres, sortert på tid. Denne funksjonaliteten skal programmeres.

Kommandoen `finn reise`:

```
Skriv kommando: finn reise
Startstasjon: Trondheim
Endestasjon: Steinkjer
Dato (YYYY-MM-DD): 2023-04-03
Tidspunkt (HH:MM): 12:00
Reiser 2023-04-03 og 2023-04-04:
    Rute #1 fra Trondheim til Steinkjer går kl. 23:05
    Rute #0 fra Trondheim til Steinkjer går kl. 07:49
    Rute #1 fra Trondheim til Steinkjer går kl. 23:05
Billetter kan kjøpes med kommandoen 'kjøp billett'
Skriv kommando:
```

- e) En bruker skal kunne registrere seg i kunderegisteret. Denne funksjonaliteten skal programmeres.

Kommandoen `ny kunde`:

```
Skriv kommando: ny kunde
Navn: Ola Nordmann
Epost: ola.nordmann@gmail.com
Telefon: 456 78 901
Kunde #8 lagt til
Skriv kommando:
```

- f) Det skal legges inn nødvendige data slik at systemet kan håndtere billettkjøp for de tre togrutene på Nordlandsbanen, mandag 3. april og tirsdag 4. april i år. Dette kan gjøres med et skript, dere trenger ikke å programmere støtte for denne funksjonaliteten.

I `reset_database` funksjonen på `Database` klassen kalles funksjonen `generate_route_instances` som lager Togruteforekomst-er. Denne kjøres altså automatisk når applikasjonen kjøres.

- g) Registrerte kunder skal kunne finne ledige billetter for en oppgitt strekning på en ønsket togrute og kjøpe de billettene hen ønsker. Denne funksjonaliteten skal programmeres.

Pass på at dere bare selger ledige plasser

Kommandoen `kjøp billett` implementerer dette, her bruker vi rute #1 fra Trondheim til Steinkjer som vi fant tidligere og kjøper den på vegne av `Ola Nordmann`-brukeren vi registrerte tidligere:

```
Skriv kommando: kjøp billett
RuteID: 1
Dato (YYYY-MM-DD): 2023-04-03
Hvilken vogn vil du kjøpe billett til?
Vogn #1 har 12 ledige seter
Vogn #2 har 8 ledige senger
Vognnummer: 1
Hvilken plass vil du kjøpe?
Sete #1 er ledig
Sete #2 er ledig
Sete #3 er ledig
Sete #4 er ledig
Sete #5 er ledig
Sete #6 er ledig
Sete #7 er ledig
Sete #8 er ledig
Sete #9 er ledig
Sete #10 er ledig
Sete #11 er ledig
Sete #12 er ledig
Setenummer: 1
Hva er din kunde-ID? 1
Fra stasjon: Trondheim
Til stasjon: Steinkjer
Billett kjøpt!
Skriv kommando:
```

- h) For en bruker skal man kunne finne all informasjon om de kjøpene hen har gjort for fremtidige

reiser. Denne funksjonaliteten skal programmeres.

Kommandoen se kjøp:

```
Skriv kommando: se kjøp
Kunde-ID: 1
Kunde #1 sine billetter:
    Ordre #1 - Rute #1 fra Trondheim til Steinkjer
    avgang 2023-04-03 kjøpt 2023-03-26 kl. 17:59
Skriv kommando:
```

Legg merke til at hvis denne koden kjøres etter 4. April vil den ikke lenger vise dette kjøpet fordi det vil være i fortiden.

NB: Vi har gjort noen endringer i databasen siden forrige endring fordi det viste seg å gjøre implementasjonen i kode ryddigere og mer intuitiv. Seng og Sete har blitt slått sammen til PassasjerPlass med et felt for type plass. I tillegg har Sovevogn og Sittevogn blitt del av Vogn. PassasjerPlass lagrer hvilken AntallRader og SeterPerRad på Sittevogn, AntallKupéer og SengerPerKupé lagres nå i PassasjerPlass ved Inndeling som representerer enten en Kupé eller en rad seter avhengig av typen vogn.