

Database for jernbanen i Norge

=====

Teste funksjoner

For å teste funksjoner, kan du kopiere og lime inn følgende kommandoer

```
python3 C_Tog_Innom_Stasjon.py --stasjon "Mo i Rana"
```

```
python3 D_Check_Tain.py --fromm "Trondheim S" --to "Mo i Rana" --timeUNIX 1680476400
```

```
python3 E_Join_Operator.py --kundeID 3 --IDoperator 2 --passord "admin"
```

```
python3 G_kjop_billett.py --fromm "Trondheim S" --to "Mosjoen" --kundeID 1 --passord "admin" --timeUNIX 1680476400
```

```
python3 H_check_bught_tickets.py --kundeID 1 --passord "admin"
```

Starte databasen fra scratch

Dette er en kjapp tutorial for hvordan kommandoene brukes for databasen. Alle filene som krever argumenter støtter --help kommando. --help kommandoen gir detaljert svar på kriteriene til argumentene.

eksempelvis:

```
python3 G_kjop_billett.py --help
```

☐ ✓

```
usage: Kjøp billett [-h] --fromm FROMM --to TO [--timeUNIX TIMEUNIX] --kundeID KUNDEID --passord PASSORD
```

Kjøp billett

optional arguments:

-h, --help show this help message and exit

--fromm FROMM Skriv inn navnet på stasjonen du vil FRA. Lukk disse med " "

--to TO Skriv inn navnet på stasjonen du vil TIL. Lukk disse med " "

--timeUNIX TIMEUNIX Skriv inn tiden du vil resie fra, dette oppgis i unix tid med GMT+1(Norsk tid). oppgir du ingen klokkeslett, vil det bli satt til nåtiden august 00:00:00 er: 1680476400

--kundeID KUNDEID Skriv tallet for din kunde ID

--passord PASSORD passord, dette skal være lukket innen " ", eksepem "Dette er mitt passord"

Her kan du bestille billetter info eller full info. en må sende med informasjon med argumenter, se --help eller -h for guide det er også en med annen kundeID. Dette må bli gjort av samme bruker. En sng vil også være utigjengelig om den brukes av noen andre på hele

A. inisiere databasen

For å starte så kan en kjøre følgende kommando for å starte på scratch. Dette vil slette databasen.

```
python3 dropAll.py
```

Deretter kan en sette opp tables i SQLite ved å kjøre kommandoen under.

```
sqlite3 bane-vest.db < databaseInit.sql
```

For å registrere Jernbanestasjonene, Banestrekning og Delstrekninger. så kjører en kommandoen under.

```
python3 A_Register_Jernbanestasjon.py
```

Linjen over vil gi CLI print av hva som kjøres til bane-vest.db. Dette ser ut som følgende.

```
INSERT INTO Banestrekning VALUES(1, 'Nordlandsbanen');
INSERT INTO Jernbanestasjon VALUES(1, 'Trondheim S', 1.5);
INSERT INTO Jernbanestasjon VALUES(1, 'Steinkjer', 3.6);
INSERT INTO Delstrekning VALUES(1,120,True,True,'Trondheim S','Steinkjer');
INSERT INTO Jernbanestasjon VALUES(1, 'Mosjoen', 6.8);
INSERT INTO Delstrekning VALUES(2,280,False,False,'Steinkjer','Mosjoen');
INSERT INTO Jernbanestasjon VALUES(1, 'Mo i Rana', 3.5);
INSERT INTO Delstrekning VALUES(3,90,False,False,'Mosjoen','Mo i Rana');
INSERT INTO Jernbanestasjon VALUES(1, 'Fauske', 34.0);
INSERT INTO Delstrekning VALUES(4,170,False,False,'Mo i Rana','Fauske');
INSERT INTO Jernbanestasjon VALUES(1, 'Bodo', 4.1);
INSERT INTO Delstrekning VALUES(5,60,False,False,'Fauske','Bodo');
```

B. Registrere data om togruter

For å initiere togrutene i Togrute tabellen, kjøres kommandoen under. Kommandoen tar inn argumentet --start, start argumentet må være en Mandag klokka 00:00:00 i UNIX format med tidskoden GMT+1. Koden vil deretter generere en uke med strekninger for togruten. For å få F_Generer_Okkuperte_seter_for_bilett.py til å fungere riktig. Må to uker med strekninger genereres. Dette kan gjøres ved å kjøre følgende kommandoer. Disse linjene vil generere for uken som med mandag 27. Mars og 3. April.

```
python3 B_Register_togruter.py --start 1680476400
```

og

```
python3 B_Register_togruter.py --start 1679871600
```

Hvis en skriver inn ugyldig tidspunkt, vil en få CLI beskjed om kriteriene til argumentet --start. Hvis gjort riktig, får en CLI print av kommandoene som blir kjørt. Disse er å finne på [Pastebin \(https://pastebin.com/u8sSDGpa\)](https://pastebin.com/u8sSDGpa), og ser ut som følgende

```
You have typed in beginning date: 2023-04-03 01:00:00
1 1
Ny Dag
Ny delstrek
2023-04-03 08:49:00
2023-04-03 10:48:00
True
INSERT INTO Strekning VALUES(2, 1, 1680504540, 1680511680, True, 1, 24);
Ny delstrek
2023-04-03 10:51:00
2023-04-03 14:17:00
False
INSERT INTO Strekning VALUES(3, 1, 1680511860, 1680524220, False, 2, 24);
Ny delstrek
2023-04-03 14:20:00
2023-04-03 15:28:00
False
```

C. Togruter innom stasjon

For å se hvilken togruter som er innom en stasjon, kan kommandoen under kjøres

```
python3 C_Tog_Innom_Stasjon.py --stasjon "Mo i Rana"
```

Kommandoen over vil gi følgende CLI print:

```
Følgende tog kjører gjennom Mo i Rana :  
TogID 1  
TogID 2  
TogID 3
```

D. Søke etter togrute

Følgende togrute, så må en oppgi til og fra med følgende argument, --to --fromm. I tillegg kan en velge å oppgi tid i unixformat GMT+1 for å søke etter et spesifikt tidspunkt. eksempel på bruk, vil være følgende:

```
python3 D_Check_Tain.py --fromm "Trondheim S" --to "Mo i Rana" --timeUNIX 1680476400
```

Den utgir søkesvaret som CLI print og ser ut som følgende:

```
Tog fra 1 til 3  
Dette tilsier at ruten følge følgende strekninger [1, 2, 3]  
Togruter som har lik baneretning [(1,),(2,)]  
  
Du kan ta følge ruter  
  
TogRute 1  
Avgang      Trondheim S      Tid: 2023-04-03 08:49:00  
Ankomst     Mo i Rana      Tid: 2023-04-03 15:28:00  
  
TogRute 2  
Avgang      Trondheim S      Tid: 2023-04-04 00:05:00  
Ankomst     Mo i Rana      Tid: 2023-04-04 06:52:00  
  
TogRute 1  
Avgang      Trondheim S      Tid: 2023-04-04 08:49:00  
Ankomst     Mo i Rana      Tid: 2023-04-04 15:28:00  
  
TogRute 2  
Avgang      Trondheim S      Tid: 2023-04-05 00:05:00  
Ankomst     Mo i Rana      Tid: 2023-04-05 06:52:00  
  
TogRute 1  
Avgang      Trondheim S      Tid: 2023-04-05 08:49:00  
Ankomst     Mo i Rana      Tid: 2023-04-05 15:28:00
```

E. Registrere brukere og operatører

for å starte registrering, må operatører genereres. dette kan gjøres med følgende kode:

```
sqlite3 bane-vest.db < E_Operator_inisiering.sql
```

Deretter kan en generere brukere med under. For brukeres sikkerhet, blir passord laget som sha256.

```
python3 E_register Bruker.py --Navn "Pom Pom Purins" --Epost "Owner@breached.vc" --Telefon 87654321 --Passord "admin"
```

Dette skal gi ut følgende:

```
Gratulerer som vår første bruker! Bruk rabattkode: TDT4145 for 41.45 kroner av ditt første kjøp  
Gratulerer Simon Dargahi, du er nå registrert med kundeID: 1, du kan finne kundeID på hjemmesiden(som ikke eksisterer foreløpig ;)) d
```

```
python3 E_register Bruker.py --Navn "Pom Pom Purins" --Epost "Owner@breached.vc" --Telefon 87654321 --Passord "admin"  
Gratulerer Pom Pom Purins, du er nå registrert med kundeID: 1, du kan finne kundeID på hjemmesiden(som ikke eksisterer foreløpig ;))
```

Med operatør og bruker på plass, kan en som bruker legge seg inn i registeret til operatøren. Dette kan gjøres med kommandoen:

```
python3 E_Join_Operator.py --kundeID 1 --IDoperator 1 --passord "admin"
```

Her kan en få flere feilmeldinger hvis noe er gjort feil. Disse er

- passord ikke korrekt
- Allerede registrert i register

Men hvis det er gjort riktig, så skal en få tilbakemelding på dette, dette ser ut som følgende

- Du er nå registrert

F. Genere nødvendig data for billettkjøp

for å genere riktig data, må OkkuperteSeter i databasen fylles opp. Dette gjøres med følgende kommando:

```
python3 F_Generer_okuperte_seter_for_bilett.py
```

Her er det viktig å bemerke seg at koden er hardcodet til å kun fungere med totalt to uker av togruter. disse trenger ikke å være kontinuelig. Når Python scriptet over er kjørt, vil en få noe som ser ut som følgende:

```
INSERT INTO OkuperteSeter(IDstrekning, Plassnr, Sengeplass) VALUES (71,6,False)
INSERT INTO OkuperteSeter(IDstrekning, Plassnr, Sengeplass) VALUES (71,7,False)
INSERT INTO OkuperteSeter(IDstrekning, Plassnr, Sengeplass) VALUES (71,8,False)
INSERT INTO OkuperteSeter(IDstrekning, Plassnr, Sengeplass) VALUES (71,9,False)
INSERT INTO OkuperteSeter(IDstrekning, Plassnr, Sengeplass) VALUES (71,10,False)
INSERT INTO OkuperteSeter(IDstrekning, Plassnr, Sengeplass) VALUES (71,11,False)
INSERT INTO OkuperteSeter(IDstrekning, Plassnr, Sengeplass) VALUES (71,12,False)
INSERT INTO OkuperteSeter(IDstrekning, Plassnr, Sengeplass) VALUES (72,1,False)
```

Det vil være 2800+ linjer med print. for å se hele, må kanskje CLI innstillinger justeres for å støtte dette.

G. kjøp billett

for å kjøpe billett må kan følgende kommando kjøres. Hvis en ikke oppgir tid, så blir øyeblikkelig tid brukt.

```
python3 G_kjop_bilett.py --fromm "Trondheim S" --to "Mosjoen" --kundeID 1 --passord "admin" --timeUNIX 1680476400
```

Her vil det være flere mulige valg å velge mellom.

H. Se billetter kjøpt

En kan se kjøpte billetter med følgende kommando

```
python3 H_check_bught_tickets.py --kundeID 1 --passord "admin"
```

Det en får ut ser som følgende:

OrdreID: 3 Plass på toget: 12
Avgang: 2023-03-23 08:11:00 Mo i Rana
Annkunst: 2023-03-23 14:10:00 Trondheim S

OrdreID: 4 Plass på toget: 11
Avgang: 2023-03-23 08:11:00 Mo i Rana
Annkunst: 2023-03-23 14:10:00 Trondheim S

OrdreID: 6 Plass på toget: 5
Avgang: 2023-03-23 08:11:00 Mo i Rana
Annkunst: 2023-03-23 14:10:00 Trondheim S

OrdreID: 5 Plass på toget: 5
Avgang: 2023-03-24 08:11:00 Mo i Rana
Annkunst: 2023-03-24 14:10:00 Trondheim S

OrdreID: 8 Plass på toget: 7
Avgang: 2023-03-27 00:05:00 Trondheim S
Annkunst: 2023-03-27 06:52:00 Mo i Rana

OrdreID: 13 Plass på toget: 15
Avgang: 2023-03-27 00:05:00 Trondheim S
Annkunst: 2023-03-27 06:52:00 Mo i Rana

OrdreID: 11 Plass på toget: 4
Avgang: 2023-04-03 08:49:00 Trondheim S
Annkunst: 2023-04-03 15:28:00 Mo i Rana

OrdreID: 12 Plass på toget: 6
Avgang: 2023-04-03 08:49:00 Trondheim S
Annkunst: 2023-04-03 15:28:00 Mo i Rana