Lecture : Flask and SQL

```
import os

from flask import Flask, render_template,
                                    request

from sqlalchemy import create_engine

from sqlalchemy.orm import
                    scoped_session, sessionmaker


app = Flask (__name__)

engine = create_engine (os.getenv ("Database
                                    -URL"))

db = scoped_session( sessionmaker
                    (bind = engine))

@ app.route ("/")

def index():

    flights = db.execute(" SELECT * FROM
                        flights"). fetchall()

    return render_template ("index.html",
                        flights = flights)
```

flights ↑ index.html placeholder...

other commands

db.execute(" ").rowcount

⋮

Can use place holders for other queries

also . .

example

db.execute(" SELECT * FROM flights

WHERE id = :id ", {"id": flight_id)

.rowcount

etc

--

place holders in SQL Alchemy also 'escapes' dangerous / malicious input ... (so you don't need to sanitise inputs separately) ...

Use

try:

_____

except ValueError:

_____

;

Flask : if internal server error,
can give a global error message...

if

```
@app.route("/flights/<int:flight_id>")
def flight(flight_id):
```

=

to link to url associated to flight with input flight_id... in html--

```
{{url_for('flight', flight_id=my_input)}}
```