# Public info

1. Search based on departure city, arrival city, departure date and/or arrival date

if departure\_city:

        if query == '':

            query = 'SELECT airline\_name, flight\_num, departure\_airport, departure\_time, arrival\_airport, arrival\_time, price, status, airplane\_id FROM (flight JOIN airport AS ad ON(flight.departure\_airport=ad.airport\_name)) JOIN airport AS aa ON(flight.arrival\_airport=aa.airport\_name) WHERE ad.airport\_city=\'{}\''.format(departure\_city)

        else:

            query += " AND ad.airport\_city=\'{}\'".format(departure\_city)

    if arrival\_city:

        if query == '':

            query = 'SELECT airline\_name, flight\_num, departure\_airport, departure\_time, arrival\_airport, arrival\_time, price, status, airplane\_id FROM (flight JOIN airport AS ad ON(flight.departure\_airport=ad.airport\_name)) JOIN airport AS aa ON(flight.arrival\_airport=aa.airport\_name) WHERE aa.airport\_city=\'{}\''.format(arrival\_city)

        else:

            query += ' AND aa.airport\_city=\'{}\''.format(arrival\_city)

    if departure\_date:

        if query == '':

            query = 'SELECT \* FROM flight WHERE DATE\_FORMAT(departure\_time, \'%Y-%m-%e\')=DATE\_FORMAT(\'{}\', \'%Y-%m-%e\')'.format(departure\_date)

        else:

            query += ' AND DATE\_FORMAT(departure\_time, \'%Y-%m-%e\')=DATE\_FORMAT(\'{}\', \'%Y-%m-%e\')'.format(departure\_date)

    if arrival\_date:

        if query == '':

            query = 'SELECT \* FROM flight WHERE DATE\_FORMAT(arrival\_time, \'%Y-%m-%e\')=DATE\_FORMAT(\'{}\', \'%Y-%m-%e\')'.format(arrival\_date)

        else:

            query += ' AND DATE\_FORMAT(arrival\_time, \'%Y-%m-%e\')=DATE\_FORMAT(\'{}\', \'%Y-%m-%e\')'.format(arrival\_date)

query += ' AND status=\'Upcoming\''

1. Abstract search

query = 'SELECT airline\_name, flight\_num, departure\_airport, departure\_time, arrival\_airport, arrival\_time, price, status, airplane\_id FROM (flight JOIN airport AS ad ON(flight.departure\_airport=ad.airport\_name)) JOIN airport AS aa ON(flight.arrival\_airport=aa.airport\_name) WHERE status=\'Upcoming\' AND (departure\_airport LIKE \'%{}%\' OR arrival\_airport LIKE \'%{}%\' OR ad.airport\_city LIKE \'%{}%\' OR aa.airport\_city LIKE \'%{}%\')'.format(searchText, searchText, searchText, searchText)

1. Search based flight number

query = 'SELECT \* FROM flight WHERE airline\_name=\'{}\' AND flight\_num={}'.format(airline\_name,flight\_number)

# Login

1. Check whether password is correct for customer

query = 'SELECT md5(\'{}\')'.format(password)

query = 'SELECT name FROM customer WHERE email=\'{}\' AND password=\'{}\''.format(email,password)

1. Check whether password is correct for agent

query = 'SELECT md5(\'{}\')'.format(password)

query = 'SELECT email FROM booking\_agent WHERE email=\'{}\' AND password=\'{}\' AND booking\_agent\_id=\'{}\''.format(email,password, id)

1. Check whether password is correct for staff

query = 'SELECT md5(\'{}\')'.format(password)

query = 'SELECT username FROM airline\_staff WHERE username=\'{}\' AND password=\'{}\''.format(email,password)

# Register

1. Password encoding

query = 'SELECT md5(\'{}\')'.format(password)

1. Check whether the customer email is already used

query = 'SELECT email FROM customer WHERE email=\'{}\''.format(email)

1. Register for a customer

query = 'INSERT INTO customer VALUES(\'{}\', \'{}\', \'{}\', \'{}\', \'{}\', \'{}\', \'{}\', {}, \'{}\', \'{}\', \'{}\', \'{}\')'.format(email,name,password,building\_number,street,city,state,phone\_number,passport\_number,passport\_expiration,passport\_country,date\_of\_birth)

query = 'CREATE USER \'customer\_{}\'@\'localhost\' IDENTIFIED BY \'{}\''.format(email,password)

query = 'GRANT customer\_role TO \'customer\_{}\'@\'localhost\''.format(email)

1. Check whether the agent email is used

query = 'SELECT email FROM booking\_agent WHERE email=\'{}\''.format(email)

1. Check whether the agent id is used

query = 'SELECT email FROM booking\_agent WHERE booking\_agent\_id=\'{}\''.format(booking\_agent\_id)

1. Register for an agent

query = 'INSERT INTO booking\_agent VALUES(\'{}\', \'{}\', {})'.format(email,password,booking\_agent\_id)

query = 'CREATE USER \'agent\_{}\'@\'localhost\' IDENTIFIED BY \'{}\''.format(email,password)

query = 'GRANT agent\_role TO \'agent\_{}\'@\'localhost\''.format(email)

1. Check whether the staff username is used

query = 'SELECT username FROM airline\_staff WHERE username=\'{}\''.format(username)

1. Register for a staff

query = 'INSERT INTO airline\_staff VALUES(\'{}\', \'{}\', \'{}\', \'{}\', \'{}\', \'{}\')'.format(username,password,first\_name,last\_name,date\_of\_birth,airline\_name)

query = 'CREATE USER \'staff\_{}\'@\'localhost\' IDENTIFIED BY \'{}\''.format(username,password)

query = 'GRANT staff\_role TO \'staff\_{}\'@\'localhost\''.format(username)

1. Logout

# Customer page

1. View my flight

query = 'SELECT airline\_name, flight\_num, departure\_airport, departure\_time, arrival\_airport, arrival\_time, price, status, airplane\_id FROM ((customer JOIN purchases ON(customer.email=purchases.customer\_email)) NATURAL JOIN ticket) NATURAL JOIN flight WHERE status=\'Upcoming\' AND email=\'{}\''.format(username[9:])

1. Track a customer’s spending in a time interval

query= """

SELECT

DATE\_FORMAT(p.purchase\_date, '%%Y-%%m') AS month,

COALESCE(SUM(f.price), 0) AS spending

FROM

purchases p

LEFT JOIN

ticket t ON p.ticket\_id = t.ticket\_id

LEFT JOIN

flight f ON t.flight\_num = f.flight\_num

WHERE

p.customer\_email = %s

AND p.purchase\_date BETWEEN %s AND %s

GROUP BY

DATE\_FORMAT(p.purchase\_date, '%%Y-%%m')

ORDER BY

month;

"""

1. Track a customer’s recent spending

query = """

SELECT

DATE\_FORMAT(NOW() - INTERVAL (n.n + 1) MONTH, '%%Y-%%m') AS month,

COALESCE(SUM(f.price), 0) AS spending

FROM

ticket t

LEFT JOIN

purchases p ON t.ticket\_id = p.ticket\_id

LEFT JOIN

flight f ON t.flight\_num = f.flight\_num

CROSS JOIN

(SELECT 0 AS n UNION SELECT 1 UNION SELECT 2 UNION SELECT 3 UNION SELECT 4 UNION SELECT 5) AS n

WHERE

p.customer\_email = %s

AND DATE\_FORMAT(p.purchase\_date, '%%Y-%%m') = DATE\_FORMAT(NOW() - INTERVAL (n.n + 1) MONTH, '%%Y-%%m')

GROUP BY

DATE\_FORMAT(NOW() - INTERVAL (n.n + 1) MONTH, '%%Y-%%m')

ORDER BY

month DESC;

"""

1. Search based on departure city, arrival city, departure date and/or arrival date

if departure\_city:

        if query == '':

            query = 'SELECT airline\_name, flight\_num, departure\_airport, departure\_time, arrival\_airport, arrival\_time, price, status, airplane\_id FROM (flight JOIN airport AS ad ON(flight.departure\_airport=ad.airport\_name)) JOIN airport AS aa ON(flight.arrival\_airport=aa.airport\_name) WHERE ad.airport\_city=\'{}\''.format(departure\_city)

        else:

            query += " AND ad.airport\_city=\'{}\'".format(departure\_city)

    if arrival\_city:

        if query == '':

            query = 'SELECT airline\_name, flight\_num, departure\_airport, departure\_time, arrival\_airport, arrival\_time, price, status, airplane\_id FROM (flight JOIN airport AS ad ON(flight.departure\_airport=ad.airport\_name)) JOIN airport AS aa ON(flight.arrival\_airport=aa.airport\_name) WHERE aa.airport\_city=\'{}\''.format(arrival\_city)

        else:

            query += ' AND aa.airport\_city=\'{}\''.format(arrival\_city)

    if departure\_date:

        if query == '':

            query = 'SELECT \* FROM flight WHERE DATE\_FORMAT(departure\_time, \'%Y-%m-%e\')=DATE\_FORMAT(\'{}\', \'%Y-%m-%e\')'.format(departure\_date)

        else:

            query += ' AND DATE\_FORMAT(departure\_time, \'%Y-%m-%e\')=DATE\_FORMAT(\'{}\', \'%Y-%m-%e\')'.format(departure\_date)

    if arrival\_date:

        if query == '':

            query = 'SELECT \* FROM flight WHERE DATE\_FORMAT(arrival\_time, \'%Y-%m-%e\')=DATE\_FORMAT(\'{}\', \'%Y-%m-%e\')'.format(arrival\_date)

        else:

            query += ' AND DATE\_FORMAT(arrival\_time, \'%Y-%m-%e\')=DATE\_FORMAT(\'{}\', \'%Y-%m-%e\')'.format(arrival\_date)

query += ' AND status=\'Upcoming\''

1. Abstract search

query = 'SELECT airline\_name, flight\_num, departure\_airport, departure\_time, arrival\_airport, arrival\_time, price, status, airplane\_id FROM (flight JOIN airport AS ad ON(flight.departure\_airport=ad.airport\_name)) JOIN airport AS aa ON(flight.arrival\_airport=aa.airport\_name) WHERE status=\'Upcoming\' AND (departure\_airport LIKE \'%{}%\' OR arrival\_airport LIKE \'%{}%\' OR ad.airport\_city LIKE \'%{}%\' OR aa.airport\_city LIKE \'%{}%\')'.format(searchText, searchText, searchText, searchText)

1. Search with flight number

query = 'SELECT \* FROM flight WHERE airline\_name=\'{}\' AND flight\_num={}'.format(airline\_name,flight\_number)

1. Purchase ticket

query = '''

SELECT t.ticket\_id

FROM ticket t

LEFT JOIN purchases p ON t.ticket\_id = p.ticket\_id

WHERE t.airline\_name = %s AND t.flight\_num = %s AND p.ticket\_id IS NULL

'''

1. Logout

def logout():

session.clear()

return redirect(url\_for('search'))

# Agent page

1. View my flight

query = 'SELECT customer\_email, airline\_name, flight\_num, departure\_airport, departure\_time, arrival\_airport, arrival\_time, price, status, airplane\_id FROM ((purchases NATURAL JOIN booking\_agent) NATURAL JOIN ticket) NATURAL JOIN flight WHERE status=\'Upcoming\' AND email=\'{}\''.format(email)

1. Total tickets

query = 'SELECT sum(price)\*0.1 AS total\_commission, avg(price)\*0.1 AS avg\_commission, COUNT(ticket\_id) AS total\_tickets FROM ((booking\_agent NATURAL JOIN purchases) NATURAL JOIN ticket) NATURAL JOIN flight WHERE datediff(now(), purchase\_date)<=30 AND email=\'{}\''.format(email)

1. Rank the customers by tickets in a time interval

query = """

SELECT customer\_email AS name, COUNT(ticket\_id) AS number\_of\_tickets

FROM ((booking\_agent NATURAL JOIN purchases) NATURAL JOIN ticket) NATURAL JOIN flight

WHERE email = %s AND purchase\_date >= %s AND purchase\_date <= %s

GROUP BY customer\_email

ORDER BY COUNT(ticket\_id)

LIMIT 5;

"""

1. Rank the customers by recent tickets

query = """

SELECT customer\_email AS name, COUNT(ticket\_id) AS number\_of\_tickets

FROM ((booking\_agent NATURAL JOIN purchases) NATURAL JOIN ticket) NATURAL JOIN flight

WHERE email = %s AND period\_diff(date\_format(now(), '%Y%m'), date\_format(purchase\_date, '%Y%m')) < 6

GROUP BY customer\_email

ORDER BY COUNT(ticket\_id)

LIMIT 5;

"""

1. Rank the customers by commission in a time interval

query = """

SELECT customer\_email AS name, SUM(price)\*0.05 AS spending

FROM ((booking\_agent NATURAL JOIN purchases) NATURAL JOIN ticket) NATURAL JOIN flight

WHERE email = %s AND purchase\_date >= %s AND purchase\_date <= %s

GROUP BY customer\_email

ORDER BY SUM(price)\*0.05 DESC

LIMIT 5;

"""

1. Rank the customers by recent commission

query = """

SELECT customer\_email AS name, SUM(price)\*0.05 AS spending

FROM ((booking\_agent NATURAL JOIN purchases) NATURAL JOIN ticket) NATURAL JOIN flight

WHERE email = %s AND period\_diff(date\_format(now(), '%Y%m'), date\_format(purchase\_date, '%Y%m')) <= 12

GROUP BY customer\_email

ORDER BY SUM(price)\*0.05 DESC

LIMIT 5;

"""

1. Check the airline(s)

query = 'SELECT airline\_name FROM booking\_agent\_work\_for WHERE email=\'{}\''.format(username[6:])

1. Search based on departure city, arrival city, departure date and/or arrival date

if departure\_city:

        if query == '':

            query = 'SELECT airline\_name, flight\_num, departure\_airport, departure\_time, arrival\_airport, arrival\_time, price, status, airplane\_id FROM (flight JOIN airport AS ad ON(flight.departure\_airport=ad.airport\_name)) JOIN airport AS aa ON(flight.arrival\_airport=aa.airport\_name) WHERE ad.airport\_city=\'{}\''.format(departure\_city)

        else:

            query += " AND ad.airport\_city=\'{}\'".format(departure\_city)

    if arrival\_city:

        if query == '':

            query = 'SELECT airline\_name, flight\_num, departure\_airport, departure\_time, arrival\_airport, arrival\_time, price, status, airplane\_id FROM (flight JOIN airport AS ad ON(flight.departure\_airport=ad.airport\_name)) JOIN airport AS aa ON(flight.arrival\_airport=aa.airport\_name) WHERE aa.airport\_city=\'{}\''.format(arrival\_city)

        else:

            query += ' AND aa.airport\_city=\'{}\''.format(arrival\_city)

    if departure\_date:

        if query == '':

            query = 'SELECT \* FROM flight WHERE DATE\_FORMAT(departure\_time, \'%Y-%m-%e\')=DATE\_FORMAT(\'{}\', \'%Y-%m-%e\')'.format(departure\_date)

        else:

            query += ' AND DATE\_FORMAT(departure\_time, \'%Y-%m-%e\')=DATE\_FORMAT(\'{}\', \'%Y-%m-%e\')'.format(departure\_date)

    if arrival\_date:

        if query == '':

            query = 'SELECT \* FROM flight WHERE DATE\_FORMAT(arrival\_time, \'%Y-%m-%e\')=DATE\_FORMAT(\'{}\', \'%Y-%m-%e\')'.format(arrival\_date)

        else:

            query += ' AND DATE\_FORMAT(arrival\_time, \'%Y-%m-%e\')=DATE\_FORMAT(\'{}\', \'%Y-%m-%e\')'.format(arrival\_date)

query += ' AND status=\'Upcoming\''

1. Abstract search

query = 'SELECT airline\_name, flight\_num, departure\_airport, departure\_time, arrival\_airport, arrival\_time, price, status, airplane\_id FROM (flight JOIN airport AS ad ON(flight.departure\_airport=ad.airport\_name)) JOIN airport AS aa ON(flight.arrival\_airport=aa.airport\_name) WHERE status=\'Upcoming\' AND (departure\_airport LIKE \'%{}%\' OR arrival\_airport LIKE \'%{}%\' OR ad.airport\_city LIKE \'%{}%\' OR aa.airport\_city LIKE \'%{}%\')'.format(searchText, searchText, searchText, searchText)

1. Search with flight number

query = 'SELECT \* FROM flight WHERE airline\_name=\'{}\' AND flight\_num={}'.format(airline\_name,flight\_number)

1. Search result about commission in a time interval

query = 'CALL agent\_ranged\_commission(\'{}\', \'{}\', \'{}\')'.format(email,commission\_from, commission\_to)

1. Search result about recent commission

query = 'SELECT sum(price)\*0.1 AS total\_commission, avg(price)\*0.1 AS avg\_commission, COUNT(ticket\_id) AS total\_tickets FROM ((booking\_agent NATURAL JOIN purchases) NATURAL JOIN ticket) NATURAL JOIN flight WHERE datediff(now(), purchase\_date)<=30 AND email=\'{}\''.format(email)

1. Check customer email

query = 'SELECT email FROM customer WHERE email=\'{}\''.format(agent\_customer\_email)

1. Purchase tickets for customers

query = 'CALL customer\_purchase(\'{}\', \'{}\')'.format(airline\_name,flight\_num)

query = 'INSERT INTO purchases VALUES({}, \'{}\', \'{}\', \'{}\')'.format(ticket\_id,customer\_email,booking\_agent\_id,today)

1. Logout

def logout():

session.clear()

return redirect(url\_for('search'))

# Staff page

1. View my flight

query = 'SELECT airline\_name, flight\_num, departure\_airport, departure\_time, arrival\_airport, arrival\_time, price, status, airplane\_id FROM flight NATURAL JOIN airline\_staff WHERE status=\'Upcoming\' AND username=\'{}\' AND datediff(departure\_time, now())<=30'.format(username)

1. Check permissions

query = 'SELECT permission\_type FROM permission WHERE username=\'{}\''.format(username)

1. Rank agents by ticket sales last month

query = 'SELECT booking\_agent\_id AS id FROM ((purchases NATURAL JOIN ticket) NATURAL JOIN flight) NATURAL JOIN airline\_staff WHERE username=\'{}\' AND booking\_agent\_id IS NOT NULL AND date\_format((now()-INTERVAL 1 month), \'%m\')=date\_format(purchase\_date, \'%m\') GROUP BY booking\_agent\_id ORDER BY COUNT(ticket\_id) DESC LIMIT 5'.format(username)

1. Rank agents by ticket sales last year

query = 'SELECT booking\_agent\_id AS id FROM ((purchases NATURAL JOIN ticket) NATURAL JOIN flight) NATURAL JOIN airline\_staff WHERE username=\'{}\' AND booking\_agent\_id IS NOT NULL AND date\_format((now()-INTERVAL 1 year), \'%Y\')=date\_format(purchase\_date, \'%Y\') GROUP BY booking\_agent\_id ORDER BY COUNT(ticket\_id) DESC LIMIT 5'.format(username)

1. Rank agents by commission last year

query = 'SELECT booking\_agent\_id AS id FROM ((purchases NATURAL JOIN ticket) NATURAL JOIN flight) NATURAL JOIN airline\_staff WHERE username=\'{}\' AND booking\_agent\_id IS NOT NULL AND date\_format((now()-INTERVAL 1 year), \'%Y\')=date\_format(purchase\_date, \'%Y\') GROUP BY booking\_agent\_id ORDER BY SUM(price) DESC LIMIT 5'.format(username)

1. Rank customers by tickets last year

query = """

WITH A AS (

SELECT customer\_email AS email, COUNT(ticket\_id) AS num\_of\_flights

FROM ((purchases NATURAL JOIN ticket) NATURAL JOIN flight) NATURAL JOIN airline\_staff

WHERE username = %s AND date\_format((now()-INTERVAL 1 year), '%%Y') = date\_format(purchase\_date, '%%Y')

GROUP BY customer\_email

)

SELECT email, name, num\_of\_flights

FROM A NATURAL JOIN customer

WHERE 1

ORDER BY num\_of\_flights DESC

LIMIT 5

"""

1. Rank destination by tickets last 3 months

query = 'SELECT airport\_city FROM ((purchases NATURAL JOIN ticket) NATURAL JOIN (flight JOIN airport ON(flight.arrival\_airport=airport.airport\_name))) NATURAL JOIN airline\_staff WHERE username=\'{}\' AND period\_diff(date\_format(now(), \'%Y%m\'), date\_format(purchase\_date, \'%Y%m\'))<=3 GROUP BY airport\_city ORDER BY COUNT(ticket\_id) DESC LIMIT 3'.format(username)

1. Rank destinations by tickets last year

query = 'SELECT airport\_city FROM ((purchases NATURAL JOIN ticket) NATURAL JOIN (flight JOIN airport ON(flight.arrival\_airport=airport.airport\_name))) NATURAL JOIN airline\_staff WHERE username=\'{}\' AND date\_format((now()-INTERVAL 1 year), \'%Y\')=date\_format(purchase\_date, \'%Y\') GROUP BY airport\_city ORDER BY COUNT(ticket\_id) DESC LIMIT 3'.format(username)

1. Number of tickets sold last year

query = 'SELECT COUNT(ticket\_id) AS amount FROM (purchases NATURAL JOIN ticket) NATURAL JOIN flight WHERE date\_format(purchase\_date, \'%Y\')=date\_format((now() - INTERVAL (1) year), \'%Y\') AND airline\_name=\'{}\''.format(airline\_name)

1. Number of tickets sold last month

query = 'SELECT COUNT(ticket\_id) AS amount FROM (purchases NATURAL JOIN ticket) NATURAL JOIN flight WHERE date\_format(purchase\_date, \'%Y-%m\')=date\_format((now() - INTERVAL (1) month), \'%Y-%m\') AND airline\_name=\'{}\''.format(airline\_name)

1. Revenue of direct sale last month

query = """

SELECT SUM(price) AS revenue

FROM (purchases NATURAL JOIN ticket) NATURAL JOIN flight

WHERE date\_format((now()-INTERVAL 1 month), '%Y-%m') = date\_format(purchase\_date, '%Y-%m')

AND booking\_agent\_id IS NULL

AND airline\_name IN (

SELECT airline\_name

FROM airline\_staff

WHERE username = %s

)

"""

1. Revenue of indirect sale (through agents) last month

query = """

SELECT SUM(price) AS revenue

FROM (purchases NATURAL JOIN ticket) NATURAL JOIN flight

WHERE date\_format((now()-INTERVAL 1 month), '%Y-%m') = date\_format(purchase\_date, '%Y-%m')

AND booking\_agent\_id IS NOT NULL

AND airline\_name IN (

SELECT airline\_name

FROM airline\_staff

WHERE username = %s

)

"""

1. Revenue of direct sale last year

query = """

SELECT SUM(price) AS revenue

FROM (purchases NATURAL JOIN ticket) NATURAL JOIN flight

WHERE date\_format((now()-INTERVAL 1 year), '%Y') = date\_format(purchase\_date, '%Y')

AND booking\_agent\_id IS NULL

AND airline\_name IN (

SELECT airline\_name

FROM airline\_staff

WHERE username = %s

)

"""

1. Revenue of indirect sale (through agents) last year

query = """

SELECT SUM(price) AS revenue

FROM (purchases NATURAL JOIN ticket) NATURAL JOIN flight

WHERE date\_format((now()-INTERVAL 1 year), '%Y') = date\_format(purchase\_date, '%Y')

AND booking\_agent\_id IS NOT NULL

AND airline\_name IN (

SELECT airline\_name

FROM airline\_staff

WHERE username = %s

)

"""

1. Ticket sales stats by month in a time interval

query = """

WITH A AS (

SELECT date\_format((now() - INTERVAL (cast(help\_topic\_id as signed) - 1) month), '%Y-%m') AS month

FROM mysql.help\_topic

WHERE date\_format((now() - INTERVAL (cast(help\_topic\_id as signed) - 1) month), '%Y-%m') >= date\_format(%s, '%Y-%m')

AND date\_format((now() - INTERVAL (cast(help\_topic\_id as signed) - 1) month), '%Y-%m') <= date\_format(%s, '%Y-%m')

),

B AS (

SELECT date\_format(purchase\_date, '%Y-%m') AS month, COUNT(ticket\_id) AS num\_ticket\_bar

FROM (purchases NATURAL JOIN ticket) NATURAL JOIN flight

WHERE date\_format(purchase\_date, '%Y-%m') >= date\_format(%s, '%Y-%m')

AND date\_format(purchase\_date, '%Y-%m') <= date\_format(%s, '%Y-%m')

AND airline\_name IN (

SELECT airline\_name

FROM airline\_staff

WHERE username = %s

)

GROUP BY date\_format(purchase\_date, '%Y-%m')

)

SELECT month, coalesce(num\_ticket\_bar, 0) AS num\_ticket

FROM A NATURAL LEFT OUTER JOIN B

ORDER BY month

"""

1. Ticket sales stats by month recently

query = """

WITH A AS (

SELECT date\_format((now() - INTERVAL (2) year + INTERVAL (cast(help\_topic\_id as signed) - 1) month), '%Y-%m') AS month

FROM mysql.help\_topic

WHERE date\_format((now() - INTERVAL (2) year + INTERVAL (cast(help\_topic\_id as signed) - 1) month), '%Y') = date\_format((now() - INTERVAL (1) year), '%Y')

),

B AS (

SELECT date\_format(purchase\_date, '%Y-%m') AS month, COUNT(ticket\_id) AS num\_ticket\_bar

FROM (purchases NATURAL JOIN ticket) NATURAL JOIN flight

WHERE date\_format(purchase\_date, '%Y') = date\_format((now() - INTERVAL (1) year), '%Y')

AND airline\_name IN (

SELECT airline\_name

FROM airline\_staff

WHERE username = %s

)

GROUP BY date\_format(purchase\_date, '%Y-%m')

)

SELECT month, coalesce(num\_ticket\_bar, 0) AS num\_ticket

FROM A NATURAL LEFT OUTER JOIN B

ORDER BY month

"""

1. Search based on departure city, arrival city, departure date and/or arrival date

if departure\_city:

        if query == '':

            query = 'SELECT airline\_name, flight\_num, departure\_airport, departure\_time, arrival\_airport, arrival\_time, price, status, airplane\_id FROM (flight JOIN airport AS ad ON(flight.departure\_airport=ad.airport\_name)) JOIN airport AS aa ON(flight.arrival\_airport=aa.airport\_name) WHERE ad.airport\_city=\'{}\''.format(departure\_city)

        else:

            query += " AND ad.airport\_city=\'{}\'".format(departure\_city)

    if arrival\_city:

        if query == '':

            query = 'SELECT airline\_name, flight\_num, departure\_airport, departure\_time, arrival\_airport, arrival\_time, price, status, airplane\_id FROM (flight JOIN airport AS ad ON(flight.departure\_airport=ad.airport\_name)) JOIN airport AS aa ON(flight.arrival\_airport=aa.airport\_name) WHERE aa.airport\_city=\'{}\''.format(arrival\_city)

        else:

            query += ' AND aa.airport\_city=\'{}\''.format(arrival\_city)

    if departure\_date\_from:

        if query == '':

            query = 'SELECT \* FROM flight WHERE DATE\_FORMAT(departure\_time, \'%Y-%m-%e\')>=DATE\_FORMAT(\'{}\', \'%Y-%m-%e\')'.format(departure\_date\_from)

        else:

            query += ' AND DATE\_FORMAT(departure\_time, \'%Y-%m-%e\')>=DATE\_FORMAT(\'{}\', \'%Y-%m-%e\')'.format(departure\_date\_from)

    if departure\_date\_to:

        if query == '':

            query = 'SELECT \* FROM flight WHERE DATE\_FORMAT(departure\_time, \'%Y-%m-%e\')<=DATE\_FORMAT(\'{}\', \'%Y-%m-%e\')'.format(departure\_date\_to)

        else:

            query += ' AND DATE\_FORMAT(departure\_time, \'%Y-%m-%e\')<=DATE\_FORMAT(\'{}\', \'%Y-%m-%e\')'.format(departure\_date\_to)

    if arrival\_date\_from:

        if query == '':

            query = 'SELECT \* FROM flight WHERE DATE\_FORMAT(arrival\_time, \'%Y-%m-%e\')>=DATE\_FORMAT(\'{}\', \'%Y-%m-%e\')'.format(arrival\_date\_from)

        else:

            query += ' AND DATE\_FORMAT(arrival\_time, \'%Y-%m-%e\')>=DATE\_FORMAT(\'{}\', \'%Y-%m-%e\')'.format(arrival\_date\_from)

    if arrival\_date\_to:

        if query == '':

            query = 'SELECT \* FROM flight WHERE DATE\_FORMAT(arrival\_time, \'%Y-%m-%e\')<=DATE\_FORMAT(\'{}\', \'%Y-%m-%e\')'.format(arrival\_date\_to)

        else:

            query += ' AND DATE\_FORMAT(arrival\_time, \'%Y-%m-%e\')<=DATE\_FORMAT(\'{}\', \'%Y-%m-%e\')'.format(arrival\_date\_to)

query += ' AND airline\_name=\'{}\''.format(airline\_name)

1. Abstract search

query = 'SELECT airline\_name, flight\_num, departure\_airport, departure\_time, arrival\_airport, arrival\_time, price, status, airplane\_id FROM (flight JOIN airport AS ad ON(flight.departure\_airport=ad.airport\_name)) JOIN airport AS aa ON(flight.arrival\_airport=aa.airport\_name) WHERE (status LIKE \'%{}%\' OR departure\_airport LIKE \'%{}%\' OR arrival\_airport LIKE \'%{}%\' OR ad.airport\_city LIKE \'%{}%\' OR aa.airport\_city LIKE \'%{}%\')'.format(searchText, searchText, searchText, searchText, searchText)

query += ' AND airline\_name=\'{}\''.format(airline\_name)

1. Search with flight number

query = 'SELECT \* FROM flight WHERE airline\_name=\'{}\' AND flight\_num={}'.format(airline\_name,flight\_number)

1. Update flight status (with operator permission)

query = 'UPDATE flight SET status=\'{}\' WHERE airline\_name=\'{}\' AND flight\_num=\'{}\''.format(status,airline\_name,flight\_num)

1. View customers in a flight

query = 'SELECT email, name FROM (purchases JOIN customer ON(purchases.customer\_email=customer.email)) NATURAL JOIN ticket WHERE airline\_name=\'{}\' AND flight\_num=\'{}\''.format(airline\_name,flight\_num)

1. View flights of a customer

query = 'SELECT airline\_name FROM airline\_staff WHERE username=\'{}\''.format(username[6:])

1. Logout

def logout():

session.clear()

return redirect(url\_for('search'))

# Admin page

1. Check if an airplane exists (which will be used for creating new airplanes or new flights)

query = 'SELECT \* FROM airplane WHERE airline\_name=\'{}\' AND airplane\_id={}'.format(airline\_name,plane\_id)

1. Check if a flight number exists (which will be used for creating new flights)

query = 'SELECT flight\_num FROM flight WHERE flight\_num={} AND airline\_name=\'{}\''.format(flight\_num,airline\_name)

1. Check if an airport exists (which will be used for creating new flights or new airports)

query = 'SELECT a.airport\_name FROM airport AS a, airport AS b WHERE a.airport\_name=\'{}\' AND b.airport\_name=\'{}\''.format(departure\_airport, arrival\_airport)

1. Check if there exists time conflict between the new flight and the existing ones

query = """

SELECT airline\_name

FROM flight

WHERE airline\_name = %s AND airplane\_id = %s

AND (

(%s >= departure\_time AND %s <= arrival\_time) OR

(%s >= departure\_time AND %s <= arrival\_time) OR

(departure\_time >= %s AND departure\_time <= %s) OR

(arrival\_time >= %s AND arrival\_time <= %s)

)

"""

1. Create new flights

query = 'INSERT INTO flight VALUES(\'{}\', {}, \'{}\', \'{}\', \'{}\', \'{}\', \'{}\', \'{}\', {})'.format(airline\_name,flight\_num,departure\_airport,departure\_time,arrival\_airport,arrival\_time,price,status,plane\_id)

query = 'SELECT seats FROM airplane WHERE airline\_name=\'{}\' AND airplane\_id={}'.format(airline\_name,plane\_id)

query = 'INSERT INTO ticket VALUES({}, \'{}\', {})'.format(ticket\_id\_start+i,airline\_name, flight\_num)

1. Create new planes

query = 'INSERT INTO airplane VALUES(\'{}\', {}, {})'.format(airline\_name, airplane\_id, seats)

1. Create new airports

query = 'INSERT INTO airport VALUES(\'{}\', \'{}\')'.format(airport\_name, airport\_city)

1. Check if the staff exists in the flight company

query = 'SELECT username FROM airline\_staff WHERE username=\'{}\' AND airline\_name=\'{}\''.format(target\_username,airline\_name)

1. Check the permission one already has

query = 'SELECT permission\_type FROM permission WHERE username=\'{}\''.format(target\_username)

1. Grant permissions

query = 'GRANT admin\_role TO \'staff\_{}\'@\'localhost\' WITH ADMIN OPTION'.format(target\_username)

query = 'GRANT operator\_role TO \'staff\_{}\'@\'localhost\' WITH ADMIN OPTION'.format(target\_username)

query = 'GRANT operator\_role TO \'staff\_{}\'@\'localhost\''.format(target\_username)

1. Check if the agent exists

query = 'SELECT email FROM booking\_agent WHERE email=\'{}\''.format(email)

1. Check if the agent is already working for the airline

query = 'SELECT email FROM booking\_agent\_work\_for WHERE email=\'{}\' AND airline\_name=\'{}\''.format(email,airline\_name)

1. Add new agents to the airline

query = 'INSERT INTO booking\_agent\_work\_for VALUES(\'{}\', \'{}\')'.format(email, airline\_name)

1. Logout

def logout():

session.clear()

return redirect(url\_for('search'))