Databases Term Project: Use Case Log

Team members: Zhiyu Chen, Yuxin Zhang

Use Case	Query	Description
Public Search		

Search flights	**SELECT** airline_name, flight_num, departure_airport, arrival_airport, departure_time, arrival_time, price
	FROM flight
	WHERE departure_airport = \'{}\' and arrival_airport = \'{}\' and CAST(departure_time AS DATE) =\'{}\'
		Enable users to search for flights with departure airport, arrival airport and desired departure date

	SELECT flight.airline_name, flight.flight_num, flight.departure_airport, flight.arrival_airport, flight.departure_time, flight.arrival_time, flight.price
	FROM flight,airport AS A
	WHERE A.airport_name = flight.departure_airport and flight.departure_airport = \'{}\' and A.airport_city = \'{}\' and flight.arrival_airport = \'{}\' and CAST(flight.departure_time AS DATE) =\'{}\'
		Enable users to search for flights with departure airport, departure city, arrival airport and desired departure date

	SELECT flight.airline_name, flight.flight_num, flight.departure_airport, flight.arrival_airport, flight.departure_time, flight.arrival_time, flight.price
	FROM flight,airport AS A
	WHERE A.airport_name = flight.arrival_airport and flight.departure_airport = \'{}\' and flight.arrival_airport = \'{}\' and A.airport_city = \'{}\' and CAST(flight.departure_time AS DATE) =\'{}\'
		Enable users to search for flights with departure airport, arrival city, arrival airport and desired departure date

	SELECT flight.airline_name, flight.flight_num, flight.departure_airport, flight.arrival_airport, flight.departure_time, flight.arrival_time, flight.price **FROM** flight,airport AS A, airport AS B
	WHERE A.airport_name = flight.departure_airport and B.airport_name = flight.arrival_airport and flight.departure_airport = \'{}\' and A.airport_city = \'{}\' and flight.arrival_airport = \'{}\' and B.airport_city = \'{}\' and CAST(flight.departure_time AS DATE) =\'{}\'
		Enable users to search for flights with departure airport, departure city, arrival airport, arrival city and desired departure date

	SELECT flight.airline_name, flight.flight_num, flight.departure_airport, flight.arrival_airport, flight.departure_time, flight.arrival_time, flight.price **FROM** flight,airport AS A, airport AS B
	WHERE A.airport_name = flight.departure_airport and B.airport_name = flight.arrival_airport and A.airport_city = \'{}\' and flight.departure_airport = \'{}\' and B.airport_city = \'{}\' and CAST(flight.departure_time AS DATE) =\'{}\'
		Enable users to search for flights with departure airport, departure city, arrival city and desired departure date

	SELECT flight.airline_name, flight.flight_num, flight.departure_airport, flight.arrival_airport, flight.departure_time, flight.arrival_time, flight.price **FROM** flight,airport AS A
	WHERE A.airport_name = flight.arrival_airport and A.airport_city = \'{}\' and flight.departure_airport = \'{}\' and CAST(flight.departure_time AS DATE) =\'{}\'
		Enable users to search for flights with departure airport, arrival city and desired departure date

	SELECT flight.airline_name, flight.flight_num, flight.departure_airport, flight.arrival_airport, flight.departure_time, flight.arrival_time, flight.price
	FROM flight,airport AS A, airport AS B
	WHERE A.airport_name = flight.departure_airport and B.airport_name = flight.arrival_airport and A.airport_city = \'{}\' and B.airport_city = \'{}\' and flight.arrival_airport = \'{}\' and CAST(flight.departure_time AS DATE) =\'{}\'
		Enable users to search for flights with departure city, arrival airport, arrival city and desired departure date

	SELECT flight.airline_name, flight.flight_num, flight.departure_airport, flight.arrival_airport, flight.departure_time, flight.arrival_time, flight.price
	FROM flight,airport AS A
	WHERE A.airport_name = flight.departure_airport and A.airport_city = \'{}\' and flight.arrival_airport = \'{}\' and CAST(flight.departure_time AS DATE) =\'{}\'
		Enable users to search for flights with departure city, arrival airport and desired departure date

	SELECT flight.airline_name, flight.flight_num, flight.departure_airport, flight.arrival_airport, flight.departure_time, flight.arrival_time, flight.price
	FROM flight,airport AS A, airport AS B
	WHERE A.airport_name = flight.departure_airport and B.airport_name = flight.arrival_airport and A.airport_city = \'{}\' and B.airport_city = \'{}\' and CAST(flight.departure_time AS DATE) =\'{}\'
		Enable users to search for flights with departure city, arrival city and desired departure date

View flight status	**SELECT** airline_name, flight_num, departure_airport, arrival_airport, departure_time, arrival_time, status
	FROM flight
	WHERE flight_num = \'{}\'
		Enable users to search for flight status with only flight num

	SELECT airline_name, flight_num, departure_airport, arrival_airport, departure_time , arrival_time, status
	FROM flight
	WHERE flight_num = \'{}\' and CAST(departure_time AS DATE) =\'{}\'
		Enable users to search for flight status with flight number and departure date

	SELECT airline_name, flight_num, departure_airport, arrival_airport, departure_time , arrival_time, status
	FROM flight
	WHERE flight_num = \'{}\' and CAST(arrival_time AS DATE) =\'{}\'
		Enable users to search for flight status with flight number and arrival date

	SELECT airline_name, flight_num, departure_airport, arrival_airport,	Enable users to search for

	departure_time , arrival_time, status
	FROM flight
	WHERE flight_num = \'{}\' and CAST(departure_time AS DATE) =\'{}\' and CAST(arrival_time AS DATE) = \'{}\'
		flight status with flight number, departure date and arrival date

	Customer	
Register	**SELECT** * **FROM** customer **WHERE** email = \'{}\'	Authenticate if the user already exists.
	INSERT INTO customer (email,name,password,building_number,street,city,state,phone_number,passport_number,passport_expiration,passport_country,date_of_birth) VALUES (\'{}\',\'{}\',MD5(\'{}\'),\'{}\',\'{}\',\'{}\',\'{}\',\'{}\',\'{}\',\'{}\',\'{}\',\'{}\')	If not, update the customer table. Store password as encoded values.

Log in authentication	**SELECT** *
	FROM customer
	WHERE email = \'{}\' and password = MD5(\'{}\')
		Authenticate if the user provides correct log in values. Password is protected with encoding.

View my flights	**SELECT** ticket_id, airline_name,booking_agent_id,purchase_date
	FROM ticket NATURAL JOIN purchases NATURAL JOIN flight
	WHERE status = 'upcoming' AND customer_email = \'{}\' ORDER BY purchase_date DESC
		Default view all upcoming flights

	SELECT ticket_id, airline_name,booking_agent_id,purchase_date
	FROM ticket NATURAL JOIN purchases NATURAL JOIN flight
	WHERE customer_email = \'{}\' AND purchase_date BETWEEN \'{}\' AND \'{}\' **ORDER BY** purchase_date DESC
		Search for purchased flights based on a range of dates

	SELECT ticket_id, airline_name,booking_agent_id,purchase_date
	FROM ticket NATURAL JOIN purchases NATURAL JOIN flight
	WHERE customer_email = \'{}\' AND purchase_date BETWEEN \'{}\' AND \'{}\' AND departure_airport =\'{}\' AND arrival_airport =\'{}\' **ORDER BY** purchase_date DESC
		Search for purchased flights based on a range of dates, departure airport and arrival airport

	SELECT ticket_id, airline_name,booking_agent_id,purchase_date
	FROM (ticket NATURAL JOIN purchases NATURAL JOIN flight AS A), airport as B, airport as C
	WHERE A.departure_airport = B.airport_name and A.arrival_airport = C.airport_name and customer_email = \'{}\' AND B.airport_city = \'{}\' AND C.airport_city = \'{}\' AND purchase_date BETWEEN \'{}\' AND \'{}\' **ORDER BY** purchase_date DESC
		Search for purchased flights based on a range of dates, departure city and arrival city

	SELECT ticket_id, airline_name,booking_agent_id,purchase_date
	FROM (ticket NATURAL JOIN purchases NATURAL JOIN flight AS A), airport as B, airport as C
	WHERE A.departure_airport = B.airport_name and A.arrival_airport = C.airport_name and customer_email = \'{}\' AND B.airport_city =
		Search for purchased flights based on a range of dates, departure airport, departure city, arrival airport and arrival city

	\'{}\' AND C.airport_city = \'{}\' AND departure_airport = \'{}\' AND arrival_airport = \'{}\' AND purchase_date BETWEEN \'{}\' AND \'{}\' **ORDER BY** purchase_date DESC	

	SELECT ticket_id, airline_name,booking_agent_id,purchase_date
	FROM (ticket NATURAL JOIN purchases NATURAL JOIN flight AS A), airport as B, airport as C
	WHERE A.departure_airport = B.airport_name and A.arrival_airport = C.airport_name and customer_email = \'{}\' AND B.airport_city = \'{}\' AND C.airport_city = \'{}\'
		Search for purchased flights based on departure city and arrival city

	SELECT ticket_id, airline_name,booking_agent_id,purchase_date
	FROM (ticket NATURAL JOIN purchases NATURAL JOIN flight AS A), airport as B, airport as C
	WHERE A.departure_airport = B.airport_name and A.arrival_airport = C.airport_name and customer_email = \'{}\' AND B.airport_city = \'{}\' AND C.airport_city = \'{}\' AND departure_airport = \'{}\' AND arrival_airport = \'{}\'
		Search for purchased flights based on departure airport, departure city, arrival airport and arrival city

	SELECT ticket_id, airline_name,booking_agent_id,purchase_date
	FROM ticket NATURAL JOIN purchases NATURAL JOIN flight
	WHERE customer_email = \'{}\' AND departure_airport = \'{}\' AND arrival_airport = \'{}\'
		Search for purchased flights based on departure airport and arrival airport

Search for flights	Performed in the public search page	/
Purchase tickets	Get seats: **SELECT** seats **FROM** airplane NATURAL JOIN flight **WHERE** airline_name = \'{}\' and flight_num = \'{}\' Get current booked seats: **SELECT** count(ticket_id) **FROM** ticket **WHERE** airline_name = \'{}\' and flight_num = \'{}\' Update ticket table: **INSERT INTO** ticket(ticket_id, airline_name, flight_num) VALUES (\'{}\',\'{}\',\'{}\') Update purchase table: **INSERT INTO** purchases(ticket_id, customer_email, purchase_date) VALUES (\'{}\',\'{}\',\'{}\')	Customers will be able to purchase tickets in the public search page. The purchase button will check for user identity and only allow customers to purchase tickets**.**

Track my spending	**SELECT** SUM(price)
	FROM purchases NATURAL JOIN flight
	WHERE customer_email = \'{}\' AND purchase_date BETWEEN \'{}\' AND \'{}\'
		Total spending in certain range of dates, either the default setting over the last year, or the user-chosen range

	SELECT MONTH(purchase_date), sum(price)
	FROM purchases NATURAL JOIN flight
	WHERE customer_email = \'{}\' AND purchase_date BETWEEN \'{}\' AND \'{}\' **GROUP BY** MONTH(purchase_date)
	ORDER BY MONTH(purchase_date)
		Month wise spending in chosen range of dates, either the default setting over the last year, or the user-chosen range

Booking Agent		
Register	**SELECT** * **FROM** booking_agent **WHERE** email = \'{}\'	Authenticate if the user already exists.
	INSERT INTO booking_agent(email, password,booking_agent_id) VALUES(\'{}\', MD5(\'{}\'),\'{}\')	If not, update the booking_agent table. Store password as encoded values.
Log in authentication	**SELECT** * **FROM** booking_agent **WHERE** email =\'{}\' and booking_agent_id = \'{}\' and password = MD5(\'{}\')	Authenticate if the user provides correct log in values. Password is protected with encoding.

View my flights	**SELECT** ticket_id, airline_name,customer_email,purchase_date
	FROM ticket NATURAL JOIN purchases
	WHERE booking_agent_id = \'{}\' AND status = 'upcoming'
	ORDER BY purchase_date DESC
		Default view all upcoming flights that the agent has booked

	SELECT ticket_id, airline_name,booking_agent_id,purchase_date
	FROM ticket NATURAL JOIN purchases NATURAL JOIN flight
	WHERE booking_agent_id = \'{}\' AND purchase_date BETWEEN \'{}\' AND \'{}\' ORDER BY purchase_date DESC
		View flights based on chosen range of dates

	SELECT ticket_id, airline_name,booking_agent_id,purchase_date
	FROM ticket NATURAL JOIN purchases NATURAL JOIN flight
	WHERE booking_agent_id = \'{}\' AND purchase_date BETWEEN \'{}\' AND \'{}\' AND departure_airport =\'{}\' AND arrival_airport =\'{}\' ORDER BY purchase_date DESC
		View flights based on chosen range of dates, departure airport and arrival airport

	SELECT ticket_id, airline_name,booking_agent_id,purchase_date
	FROM (ticket NATURAL JOIN purchases NATURAL JOIN flight AS A), airport as B, airport as C
	WHERE A.departure_airport = B.airport_name and A.arrival_airport = C.airport_name and booking_agent_id = \'{}\' AND B.airport_city = \'{}\' AND C.airport_city = \'{}\' AND purchase_date BETWEEN \'{}\' AND \'{}\' ORDER BY purchase_date DESC
		View flights based on chosen range of dates, departure city and arrival city

	SELECT ticket_id, airline_name,booking_agent_id,purchase_date
	FROM (ticket NATURAL JOIN purchases NATURAL JOIN flight AS A), airport as B, airport as C
	WHERE A.departure_airport = B.airport_name and A.arrival_airport = C.airport_name and booking_agent_id = \'{}\' AND B.airport_city = \'{}\' AND C.airport_city = \'{}\' AND departure_airport = \'{}\' AND arrival_airport = \'{}\' AND purchase_date BETWEEN \'{}\' AND \'{}\' ORDER BY purchase_date DESC
		View flights based on chosen range of dates, departure airport, departure city, arrival airport and arrival city

	SELECT ticket_id, airline_name,booking_agent_id,purchase_date
	FROM (ticket NATURAL JOIN purchases NATURAL JOIN flight AS A), airport as B, airport as C
		View flights based on departure city and arrival city

	WHERE A.departure_airport = B.airport_name and A.arrival_airport = C.airport_name and booking_agent_id = \'{}\' AND B.airport_city = \'{}\' AND C.airport_city = \'{}\'	
	SELECT ticket_id, airline_name,booking_agent_id,purchase_date **FROM** (ticket NATURAL JOIN purchases NATURAL JOIN flight AS A), airport as B, airport as C **WHERE** A.departure_airport = B.airport_name and A.arrival_airport = C.airport_name and booking_agent_id = \'{}\' AND B.airport_city = \'{}\' AND C.airport_city = \'{}\' AND departure_airport = \'{}\' AND arrival_airport = \'{}\'	View flights based on departure airport, departure city, arrival airport and arrival city
	SELECT ticket_id, airline_name,booking_agent_id,purchase_date **FROM** ticket NATURAL JOIN purchases NATURAL JOIN flight **WHERE** booking_agent_id = \'{}\' AND departure_airport = \'{}\' AND arrival_airport = \'{}\'	View flights based on departure airport and arrival airport
Search for flights	Performed in the public search page	/
Purchase tickets	Get seats: **SELECT** seats **FROM** airplane NATURAL JOIN flight **WHERE** airline_name = \'{}\' and flight_num = \'{}\' Get current booked seats: **SELECT** count(ticket_id) **FROM** ticket **WHERE** airline_name = \'{}\' and flight_num = \'{}\' Update ticket table: **INSERT INTO** ticket(ticket_id, airline_name, flight_num) VALUES (\'{}\',\'{}\',\'{}\') Update purchase table: **INSERT INTO** purchases(ticket_id, customer_email,booking_agent_id, purchase_date) VALUES (\'{}\',\'{}\',\'{}\',\'{}\')	Booking agents will be able to purchase tickets in the public search page. The book button will check for user identity and only allow booking agents to purchase tickets for customers.
View commissions	**SELECT** sum(price)*0.1 **FROM** purchases NATURAL JOIN flight **WHERE** booking_agent_id = \'{}\' AND purchase_date BETWEEN \'{}\' AND \'{}\'	View commissions in chosen range of dates, either the past 30 days (default), or the user-chosen range
View top customers	**SELECT** customer_email,count(ticket_id) **FROM** purchases **WHERE** booking_agent_id = \'{}\' AND purchase_date BETWEEN \'{}\' AND \'{}\' GROUP BY customer_email ORDER BY count(ticket_id) DESC	Top 5 customers based on number of tickets bought from the booking agent in the past 6 months
	SELECT customer_email,sum(price)*0.1 **FROM** purchases NATURAL JOIN flight **WHERE** booking_agent_id = \'{}\' AND purchase_date BETWEEN \'{}\' AND \'{}\' GROUP BY customer_email ORDER BY sum(price) DESC	Top 5 customers based on the amount of commissions received in the last year
	Airline Staff	
Register	**SELECT** * **FROM** airline_staff **WHERE** username = \'{}\' AND	Authenticate if the user

	airline_name = \'{}\'	already exists.
	INSERT INTO airline_staff(username, password, airline_name,first_name,last_name,date_of_birth) VALUES(\'{}\', MD5(\'{}\'),\'{}\',\'{}\', \'{}\',\'{}\')	If not, update the airline_staff table. Store password as encoded values.
Log in authentication	**SELECT** * **FROM** airline_staff **WHERE** username = \'{}\' and password = MD5(\'{}\') and airline_name = \'{}\'	Authenticate if the user provides correct log in values. Password is protected with encoding.
View my flights	**SELECT** airline_name, flight_num, departure_airport, departure_time, arrival_airport, arrival_time, status **FROM** flight **WHERE** airline_name = \'{}\' AND status = 'upcoming' AND CAST(departure_time AS DATE) BETWEEN \'{}\' AND \'{}\' ORDER BY departure_time DESC	Default view upcoming flights for the staff's airline in next 30 days
	SELECT airline_name, flight_num, departure_airport, departure_time, arrival_airport, arrival_time, status **FROM** flight **WHERE** airline_name = \'{}\' AND CAST(departure_time AS DATE) BETWEEN \'{}\' AND \'{}\'	View flights based on chosen range of dates for the particular airline
	SELECT airline_name, flight_num, departure_airport, departure_time, arrival_airport, arrival_time, status **FROM** flight **WHERE** airline_name = \'{}\' AND departure_airport = \'{}\' AND arrival_airport = \'{}\' AND CAST(departure_time AS DATE) BETWEEN \'{}\' AND \'{}\'	View flights based on chosen range of dates, departure airport, arrival airport for the particular airline
	SELECT A.airline_name, A.flight_num, A.departure_airport, A.departure_time, A.arrival_airport, A.arrival_time, A.status **FROM** flight as A, airport as B, airport as C **WHERE** A.departure_airport = B.airport_name AND A.arrival_airport = C.airport_name AND A.airline_name = \'{}\' AND B.airport_city = \'{}\' AND C.airport_city = \'{}\' AND CAST(A.departure_time AS DATE) BETWEEN \'{}\' AND \'{}\'	View flights based on chosen range of dates, departure city, arrival city for the particular airline
	SELECT A.airline_name, A.flight_num, A.departure_airport, A.departure_time, A.arrival_airport, A.arrival_time, A.status **FROM** flight as A, airport as B, airport as C **WHERE** A.departure_airport = B.airport_name AND A.arrival_airport = C.airport_name AND A.airline_name = \'{}\' AND A.departure_airport = \'{}\' and A.arrival_airport = \'{}\' AND B.airport_city = \'{}\' AND C.airport_city = \'{}\' AND CAST(A.departure_time AS DATE) BETWEEN \'{}\' AND \'{}\'	View flights based on chosen range of dates, departure airport, departure city, arrival airport, arrival city for the particular airline
	SELECT A.airline_name, A.flight_num, A.departure_airport, A.departure_time, A.arrival_airport, A.arrival_time, A.status **FROM** flight as A, airport as B, airport as C **WHERE** A.departure_airport = B.airport_name AND A.arrival_airport = C.airport_name AND A.airline_name = \'{}\' AND B.airport_city =	View flights based on departure city, arrival city for the particular airline

	\'{}\' AND C.airport_city = \'{}\'	

	SELECT A.airline_name, A.flight_num, A.departure_airport, A.departure_time, A.arrival_airport, A.arrival_time, A.status
	FROM flight as A, airport as B, airport as C
	WHERE A.departure_airport = B.airport_name AND A.arrival_airport = C.airport_name AND A.airline_name = \'{}\' AND B.airport_city = \'{}\' AND C.airport_city = \'{}\' AND A.departure_airport = \'{}\' AND A.arrival_airport = \'{}\'
		View flights based on departure airport, departure city, arrival airport, arrival city for the particular airline

	SELECT airline_name, flight_num, departure_airport, departure_time, arrival_airport, arrival_time, status
	FROM flight **WHERE** airline_name = \'{}\' AND departure_airport = \'{}\' AND arrival_airport = \'{}\'
		View flights based on departure airport, arrival airport for the particular airline

Create new flights	**SELECT** * **FROM** flight **WHERE** airline_name = \'{}\' AND flight_num = \'{}\'	Check if such flight already exists in our database
	INSERT INTO flight(airline_name,flight_num,departure_airport,departure_time,arrival_airport, arrival_time, price, status, airplane_id) values (\'{}\', \'{}\', \'{}\',\'{}\', \'{}\', \'{}\',\'{}\', \'{}\', \'{}\')	If not, update the flight table with the new flight
Change flight status	**SELECT** * **FROM** flight **WHERE** airline_name = \'{}\' AND flight_num = \'{}\'	Check if such flight does exist
	UPDATE flight SET status = \'{}\' WHERE airline_name = \'{}\' AND flight_num = \'{}\'	If so, update the flight status
Add new airplane	**SELECT** * **FROM** airplane **WHERE** airline_name = \'{}\' AND airplane_id = \'{}\'	Check if such airplane already exists in our database
	INSERT INTO airplane values (\'{}\', \'{}\', \'{}\')	If not, update the airplane table with the new airplane
Add new airport	**SELECT** * **FROM** airport **WHERE** airport_name = \'{}\'	Check if such airport already exists in our database
	INSERT INTO airport values (\'{}\', \'{}\')	If not, update the airport table with the new airport

View booking agents	**SELECT** email, booking_agent_id, count(ticket_id) as total
	FROM purchases NATURAL JOIN ticket NATURAL JOIN booking_agent **WHERE** (purchase_date BETWEEN DATE_SUB(CURRENT_DATE(),INTERVAL 1 MONTH) AND CURRENT_DATE()) AND airline_name = \'{}\' GROUP BY email ORDER BY count(ticket_id) DESC LIMIT 5
		View top 5 booking agents by last month ticket sales

	SELECT email, booking_agent_id, count(ticket_id) as total
	FROM purchases NATURAL JOIN ticket NATURAL JOIN booking_agent **WHERE** (purchase_date BETWEEN
		View top 5 booking agents by last year ticket sales

	DATE_SUB(CURRENT_DATE(),INTERVAL 1 YEAR) AND CURRENT_DATE()) AND airline_name = \'{}\' GROUP BY email ORDER BY count(ticket_id) DESC LIMIT 5	

	SELECT email, booking_agent_id, sum(price) * 0.1 as commission
	FROM ticket NATURAL JOIN flight NATURAL JOIN purchases NATURAL JOIN booking_agent **WHERE** (purchase_date BETWEEN DATE_SUB(CURRENT_DATE(),INTERVAL 1 YEAR) AND CURRENT_DATE()) AND airline_name = \'{}\' GROUP BY email ORDER BY sum(price) * 0.1 DESC LIMIT 5
		View top 5 booking agents by last year commission

View frequent customers	**SELECT** email, name, count(ticket_id) as total
	FROM ticket NATURAL JOIN purchases as T, customer
	WHERE airline_name = \'{}\' AND T.customer_email = customer.email AND (purchase_date BETWEEN DATE_SUB (CURRENT_DATE(),INTERVAL 1 YEAR) AND CURRENT_DATE()) GROUP BY email ORDER BY count(ticket_id) DESC
		View the most frequent customer in the last year based on the number of purchased tickets.

	SELECT airline_name, flight_num, departure_airport, departure_time, arrival_airport, arrival_time, price
	FROM flight NATURAL JOIN ticket NATURAL JOIN purchases
	WHERE customer_email = \'{}\' AND airline_name = \'{}\'
		View all the flights that the most frequent customer has purchased on the staff's airline

View reports	**SELECT** month(purchase_date) as month, count(ticket_id) as num
	FROM purchases NATURAL JOIN ticket
	WHERE purchase_date BETWEEN DATE_SUB (CURRENT_DATE(),INTERVAL 1 YEAR) AND CURRENT_DATE() AND airline_name = \'{}\' GROUP BY month ORDER BY month
		View reports of the last year's tickets sale in month-wise charts

	SELECT month(purchase_date) as month, count(ticket_id) as num
	FROM purchases NATURAL JOIN ticket
	WHERE purchase_date BETWEEN DATE_SUB (CURRENT_DATE(),INTERVAL 1 MONTH) AND CURRENT_DATE() AND airline_name = \'{}\' GROUP BY month ORDER BY month
		View reports of the last month's tickets sale in month-wise charts

	SELECT month(purchase_date) as month, count(ticket_id) as num
	FROM purchases NATURAL JOIN ticket
	WHERE airline_name = \'{}\' AND purchase_date BETWEEN \'{}\' AND \'{}\' GROUP BY month ORDER BY month
		View reports of the tickets sale given the chosen range of dates in month-wise charts

Revenue comparison	**SELECT** sum(price)
	FROM flight NATURAL JOIN ticket NATURAL JOIN purchases
	WHERE booking_agent_id **is null** AND (purchase_date BETWEEN DATE_SUB (CURRENT_DATE(),INTERVAL 1 YEAR) AND CURRENT_DATE()) AND airline_name = \'{}\'
		Total ticket sales via direct customer purchase in last year

	SELECT sum(price)
	FROM flight NATURAL JOIN ticket NATURAL JOIN purchases
	WHERE booking_agent_id **is not null** AND (purchase_date BETWEEN DATE_SUB (CURRENT_DATE(),INTERVAL 1 YEAR) AND
		Total ticket sales via indirect agent booking in last year

	CURRENT_DATE()) AND airline_name = \'{}\'	

	SELECT sum(price)
	FROM flight NATURAL JOIN ticket NATURAL JOIN purchases
	WHERE booking_agent_id **is null** AND (purchase_date BETWEEN DATE_SUB (CURRENT_DATE(),INTERVAL 1 MONTH) AND CURRENT_DATE()) AND airline_name = \'{}\'
		Total ticket sales via direct customer purchase in last month

	SELECT sum(price)
	FROM flight NATURAL JOIN ticket NATURAL JOIN purchases
	WHERE booking_agent_id **is not null** AND (purchase_date BETWEEN DATE_SUB (CURRENT_DATE(),INTERVAL 1 MONTH) AND CURRENT_DATE()) AND airline_name = \'{}\'
		Total ticket sales via indirect agent booking in last month

View top destinations	**SELECT** airport_city, count(ticket_id) as num **FROM** purchases NATURAL JOIN ticket NATURAL JOIN flight, airport
	WHERE airport_name = arrival_airport AND airline_name = \'{}\' AND (purchase_date BETWEEN DATE_SUB (CURRENT_DATE(),INTERVAL 3 MONTH) AND CURRENT_DATE()) GROUP BY airport_city ORDER BY count(ticket_id) DESC LIMIT 3
		View top 3 destinations over the last 3 months

	SELECT airport_city, count(ticket_id) as num **FROM** purchases NATURAL JOIN ticket NATURAL JOIN flight, airport
	WHERE airport_name = arrival_airport AND airline_name = \'{}\' AND (purchase_date BETWEEN DATE_SUB (CURRENT_DATE(),INTERVAL 1 YEAR) AND CURRENT_DATE()) GROUP BY airport_city ORDER BY count(ticket_id) DESC LIMIT 3
		View top 3 destinations over the last year