FlyRes User Manual

Payton Wiscovitch

Informatics Department Spring 23B

Graduate Project

Table of Contents

Introduction

The following document is to serve as a user level document for the database administrator/manager and customer representatives to perform user level transactions based on the project requirements. This is a companion document to the overall project technical specifications document. This document will start by walking the user through the phpMyAdmin user interface and then explain how to perform necessary tasks. Based on the project requirements, users need to be able to perform the following tasks/transactions:

- List all flights
- List flights by airport
- View reservations
- View reservations by customer
- Create a reservation
- Revenue report by month
- Revenue generated by customer
- Revenue generated by employee (customer representative)
- Add, edit, and delete passenger/passenger information
- Create a customer mailing list
- View passenger(s) reserved for a specific flight
- Make suggestions for flights based on customer reservations

phpMyAdmin – User Interface

PhpMyAdmin was chosen for this project for the ease of use and the GUI provided for non-technical users. The

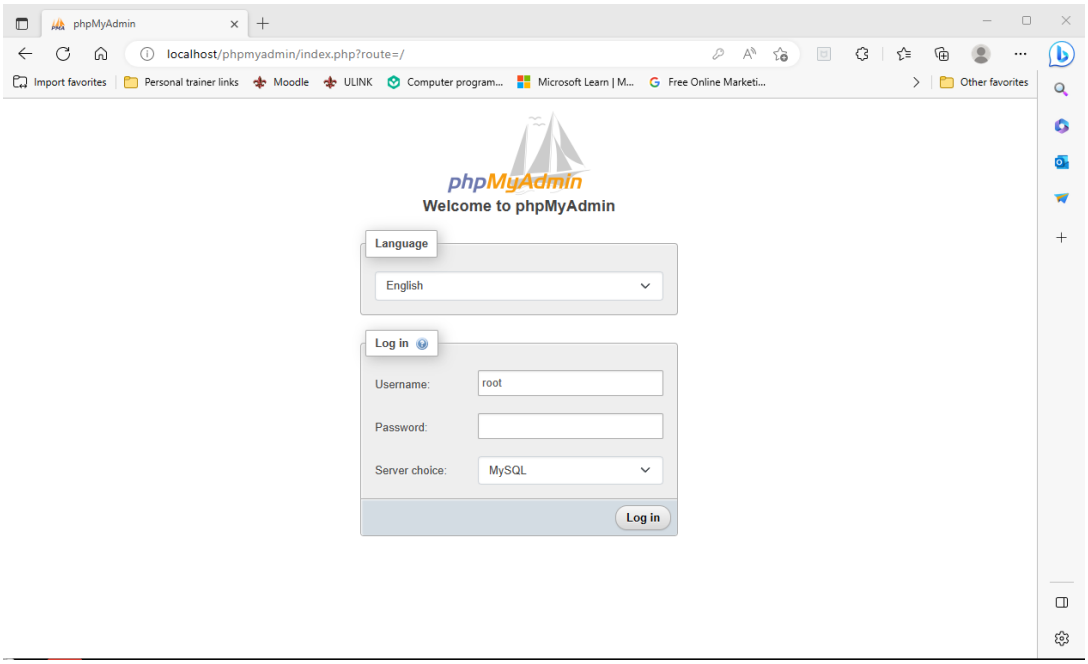
> "phpMyAdmin is a free software tool written in PHP, intended to handle the administration of MySQL over the Web. phpMyAdmin supports a wide range of operations on MySQL and MariaDB. Frequently used operations (managing databases, tables, columns, relations, indexes, users, permissions, etc) can be performed via the user interface, while you still have the ability to directly execute any SQL statement."[1]

interface allows for both traditional use of MySQL commands and queries, as well as simple point and click administration. The program can be downloaded directly from www.phpmyadmin.net.

GUI Layout

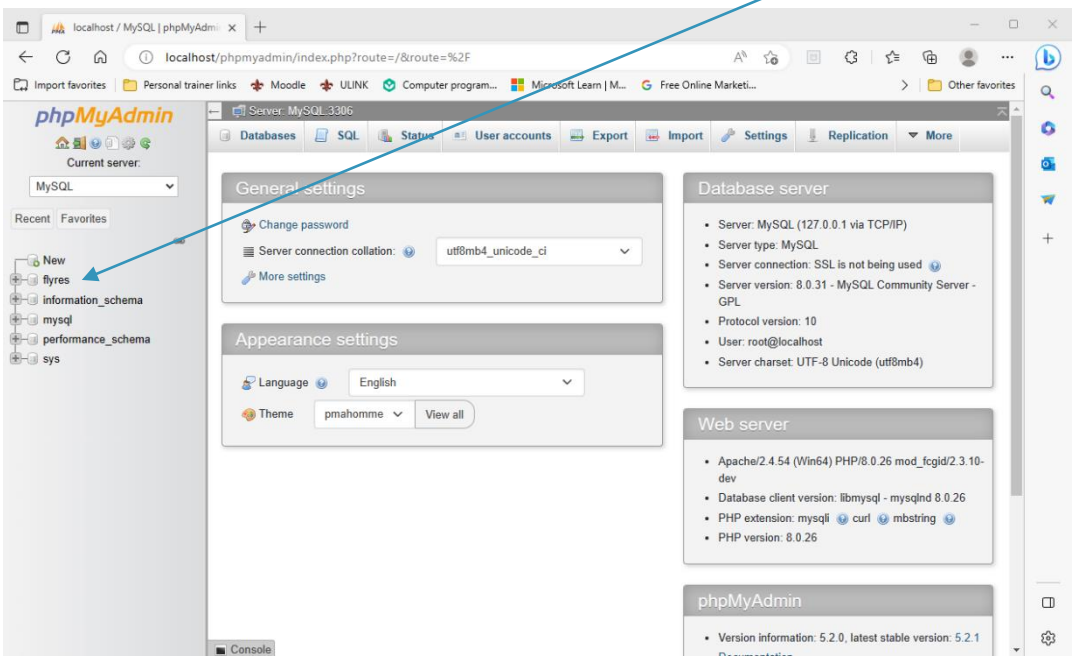
Login Screen

Once the phpMyAdmin software is downloaded and opened a login screen appears. An administrator user name and password for the FlyRes database has been established. FlyRes management will login using the user name: manager and password: flyresmanager. You will notice that it opens as a part of the user's web browser as it is a web-based application.



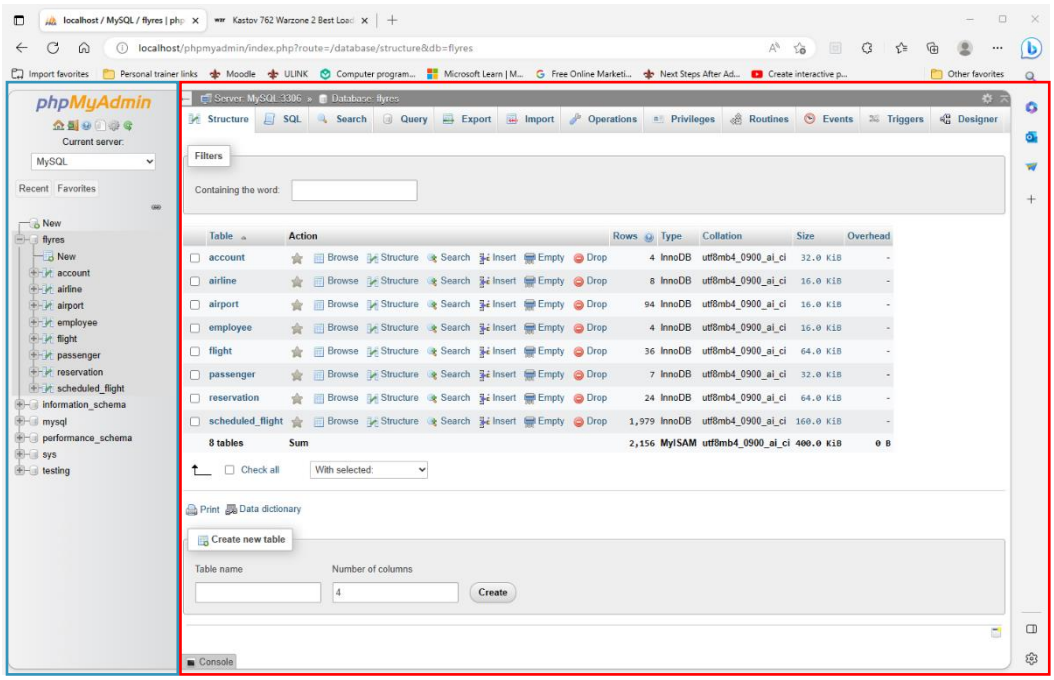
Basic GUI Layout

This is the main screen of the phpMyAdmin GUI. On the left side panel are all the databases listed. The database we will be working with is flyres. Across the top toolbar the administrator will see the option to view all databases, enter SQL commands, view server status, add/remove/edit user accounts, export information, import information, view and edit settings, and more. The first thing we want to do is select the flyres database so that we will be working within that database.



FlyRes Database Main Page

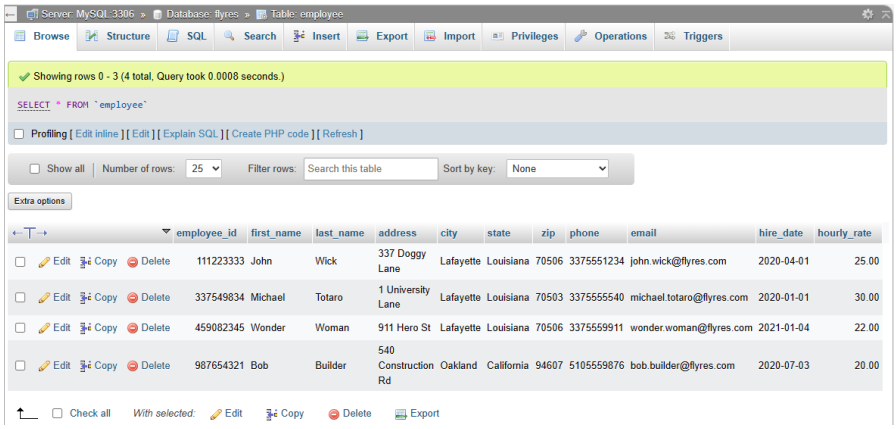
After opening the FlyRes database, a main page screen will appear showing all tables within the database.



All tables are listed in the left side panel under the flyres database as well as the center console. In the center of the screen the tables are listed along with the available functions such as browse, view table structure, table search, insert information into the table, empty the table, or drop/delete the table. The main functions the administrator will use are browse, search, and insert.

Browse

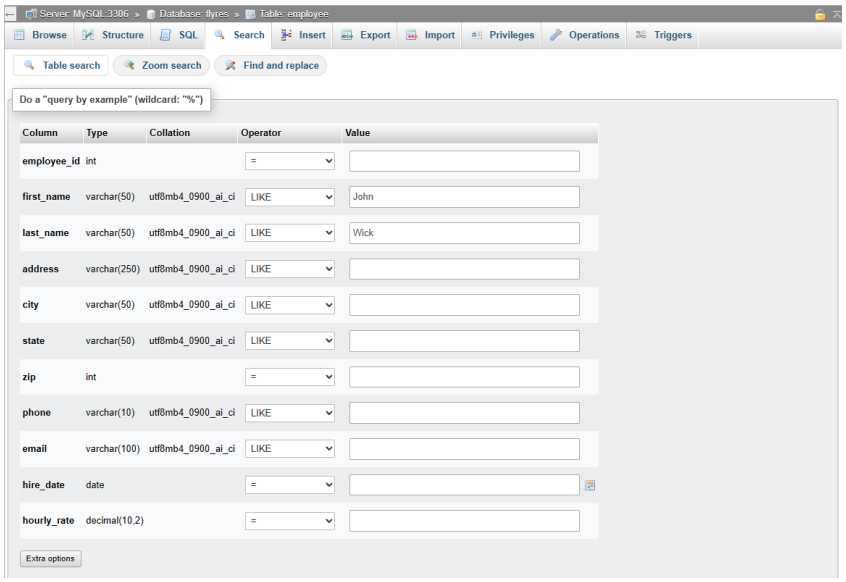
The browse function performs a simple SELECT all FROM <table> command. Which returns all entries for a given table. This is a great function to use to see all the information available for any given table. For example, if we wanted to see all the employees in the employee table, we could simply select browse for the employee table.



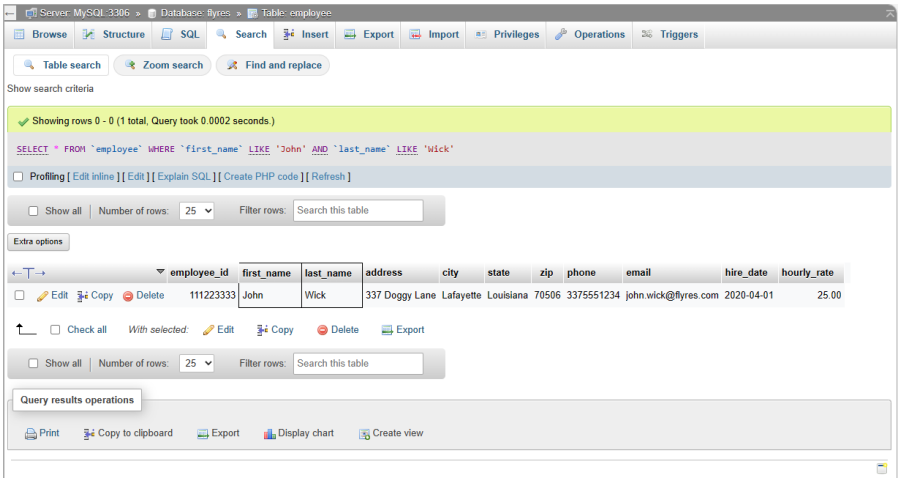
Search

The search function allows us to search a table for a specific record using the available variables. For example, if we wanted to search for a specific employee by first and last name, we could click search under the employee table and then search this information.

1. Search screen for employee table: enter employee first name and last name to search and click 'Go'.



2. Search results are generated and displayed as follows.

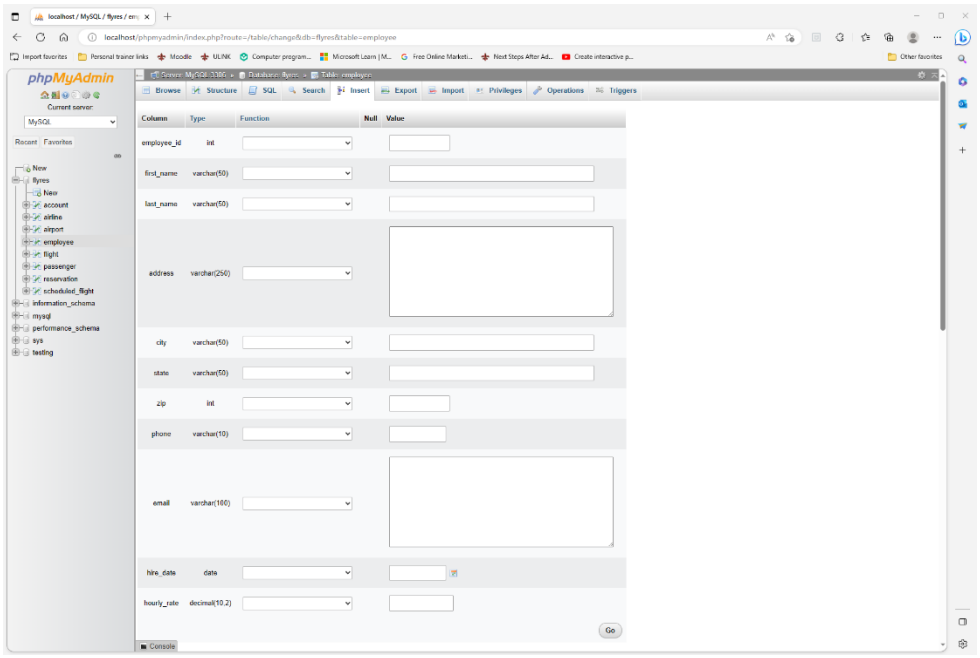


3. Based on the purpose of the search the administrator can then use the information, edit the information, or delete the information.

Insert

The last function that will be performed from the main database screen is the insert function. For example, if the administrator needs to add a new employee. They will select insert on the employee table row.

1. After selecting insert on the employee table the following input screen will appear.



2. All information is required to be filled out before submitting using the 'Go' button to create a new record. Specifics for adding new employees will be detailed later under the adding/editing/deleting employee section.

GUI Overview Wrap-up

The best way to familiarize yourself with the phpMyAdmin graphical user interface is to follow this guide and explore the interface hands on. The interface is designed with ease of use in mind and combining the user-friendly design with the detailed instructions contained in this manual will help the flyres administrator to correctly perform necessary functions.

Flight Information

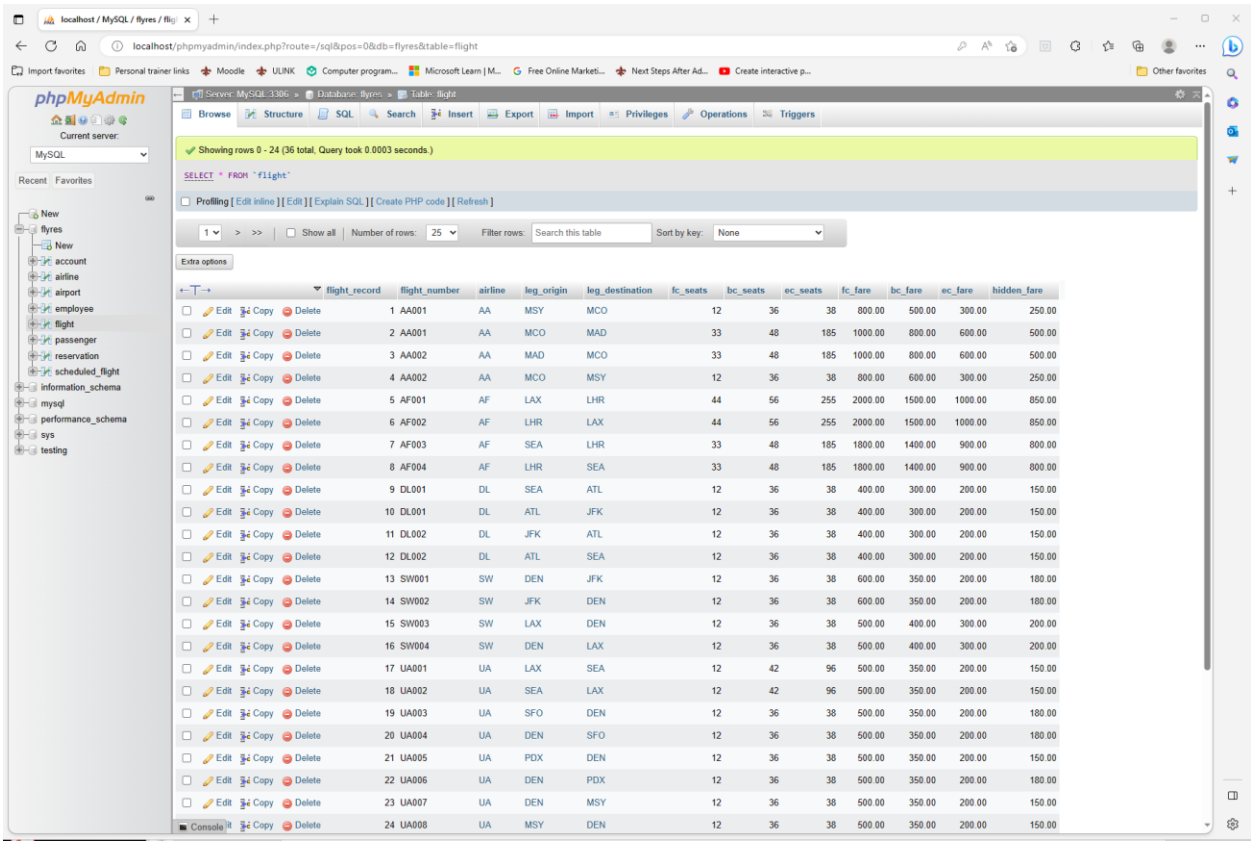
Overview

The following section will explain in detail how to access flight information. This will include how to view all flights, view all scheduled flights, looking up airport codes, viewing all flights by airport code, viewing flights by date, and viewing which passengers are reserved on a scheduled flight. It is important to know that the flight list contains general information about flights as it pertains to the route that the flight operates on and the associated fares for the flight. The scheduled_flight table lists all scheduled flights with dates and times for the flights contained within the flight table. So, the flight table will be used to gather information about routes while the scheduled_flight table is used to search for the information we gathered using the flight table. Flights are referenced using their flight record associated with specified origin and destination.

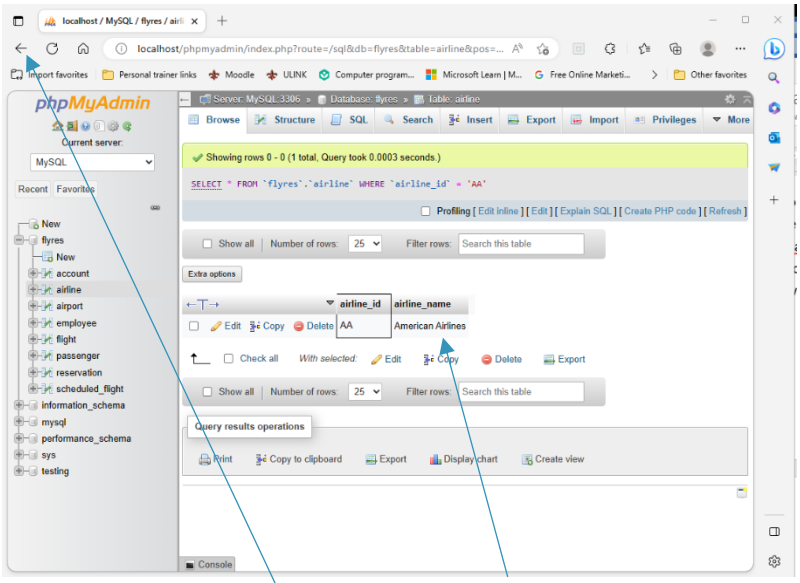
Flight Table

View All Flights

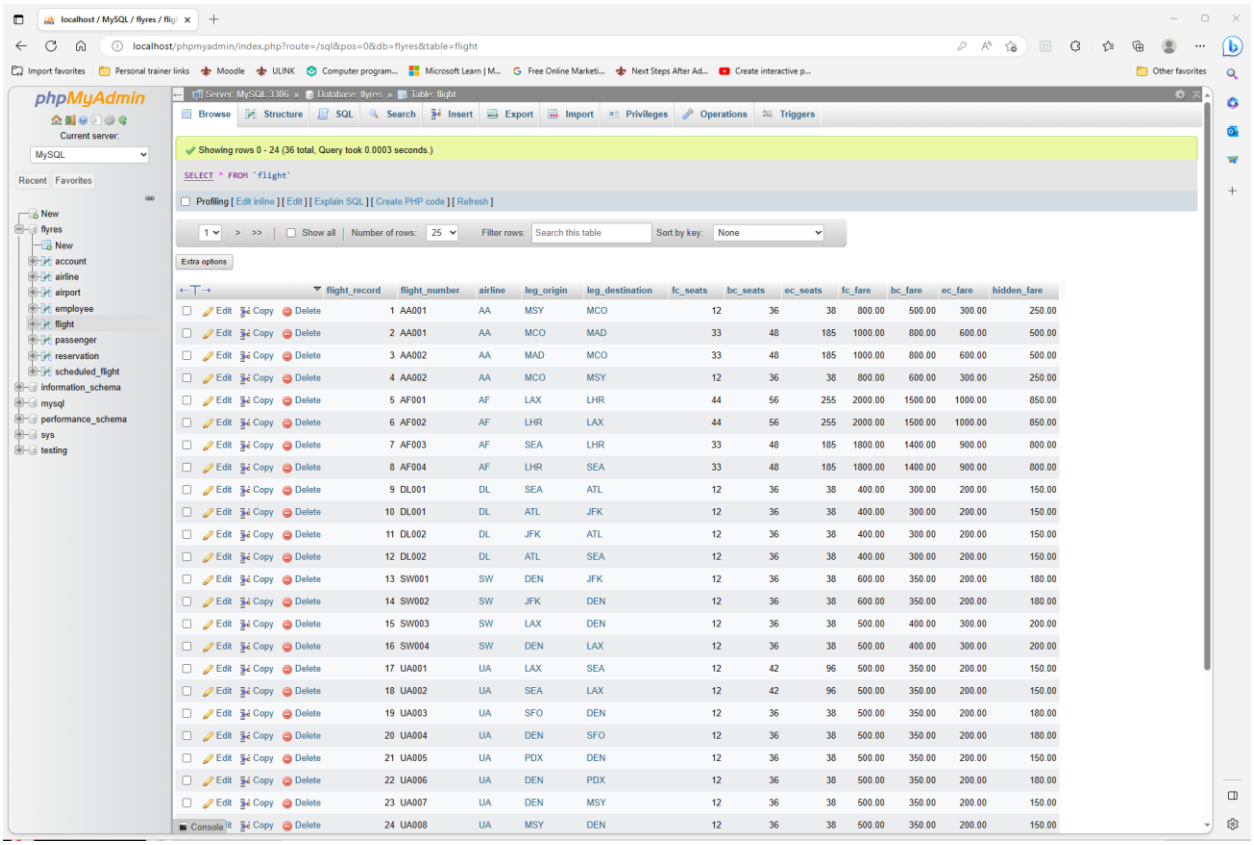
In order to view all flights, expand/open the flyres database as explained in the GUI introduction section by clicking on flyres in the left-hand side panel of phpMyAdmin. Then click on the flight table listed on the main page. This will perform a SELECT * FROM flight command and return a list of all flights and the corresponding information as shown below.



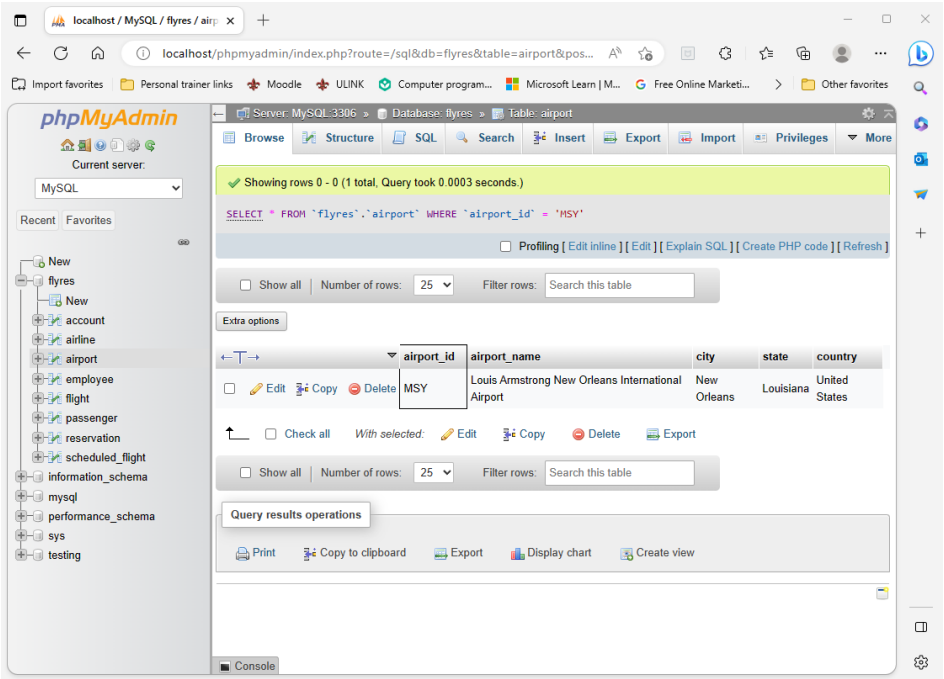
All flights have a flight number, an airline, origin, destination, number of seats in each class, and the fares associated with each class as well as a hidden fare which is the lowest economy class fare that the airline would accept in a passenger bid. Airline, leg_origin, and leg_destination are linked to the airline and airport tables respectively. To get further information on the airline or airport, simply click the blue text. For example, flight AA001 is operated by airline AA, if we click the blue text AA, we get the following result.



The result shows that flight AA001 is operated by airline AA = American Airlines. To return to the previous screen and view all flights, simply click the back button on the web browser.



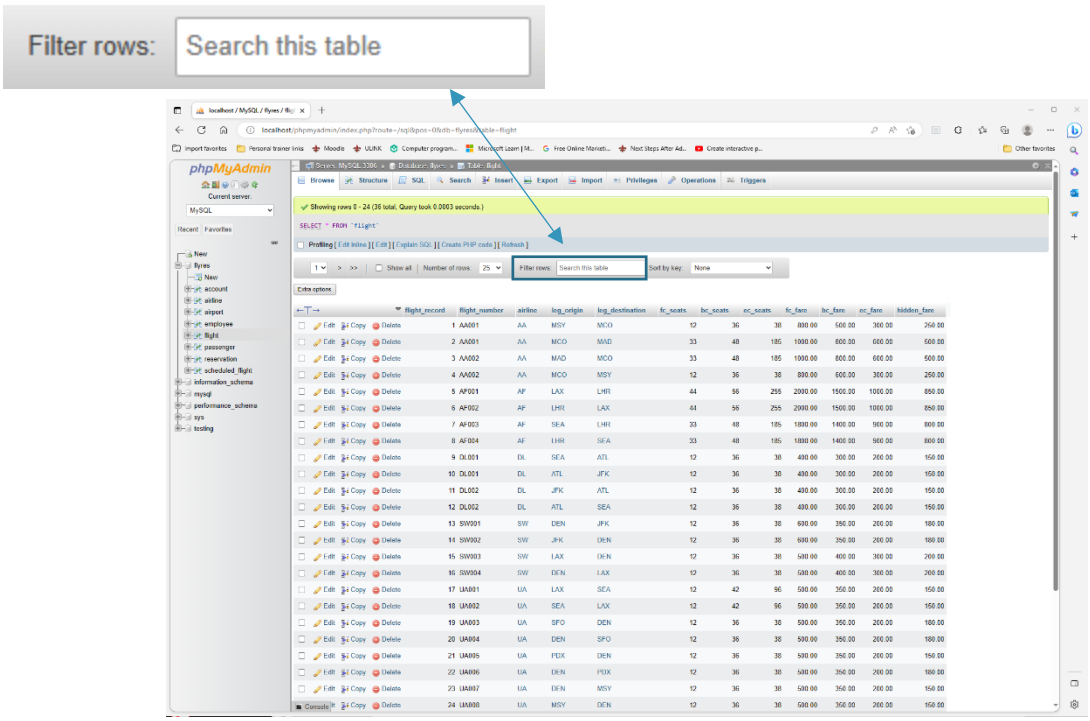
Once back at the main flight screen we can also click through the origin or destination to see information regarding the airports. For this example, we will click through the MSY link to view information for this airport code.



By clicking through the leg_origin airport code we see the information about this airport. MSY is the airport code for Louis Armstrong New Orleans International Airport in New Orleans, Louisiana, United States. To return to the previous flight list we simply click the back button on the web browser to return to the previous page.

Flights by Airport Code

From the main flight list screen, we can search all flights that operate into or out of a specified airport or all flights operated by a specified airline. From the main flight list, above the flight table there is an input to filter rows. In this input field we can enter information to filter the results shown. Let's say we wanted to search all flights departing from or arriving at MSY. We would simply type MSY into this search box.



Type search constraints into the input field to filter the table. For this example, we will type MSY into the box which returns the following result.

1 > >> | Show all | Number of rows: 25 Filter rows: MSY Sort by key: None

Extra options

	flight_record	flight_number	airline	leg_origin	leg_destination	fc_seats	bc_seats	ec_seats	fc_fare	bc_fare	ec_fare	hidden_fare
Edit Copy Delete	1	AA001	AA	MSY	MCO	12	36	38	800.00	500.00	300.00	250.00
Edit Copy Delete	4	AA002	AA	MCO	MSY	12	36	38	800.00	600.00	300.00	250.00
Edit Copy Delete	23	UA007	UA	DEN	MSY	12	36	38	500.00	350.00	200.00	150.00
Edit Copy Delete	24	UA008	UA	MSY	DEN	12	36	38	500.00	350.00	200.00	150.00

We can see from the screen shot that there are two flights that depart from MSY and two flights which arrive at MSY. We will return to the main flight list by clicking the back button in the web browser.

Flights by Airline

Once back at the main flight list we can repeat the steps above using an airline code. So, in the filter rows input we can type in AA for American Airlines to view all flights operated by American Airlines. The results show that American Airlines operated four flights and all the information for these four flights.

1 > >> | Show all | Number of rows: 25 Filter rows: AA Sort by key: None

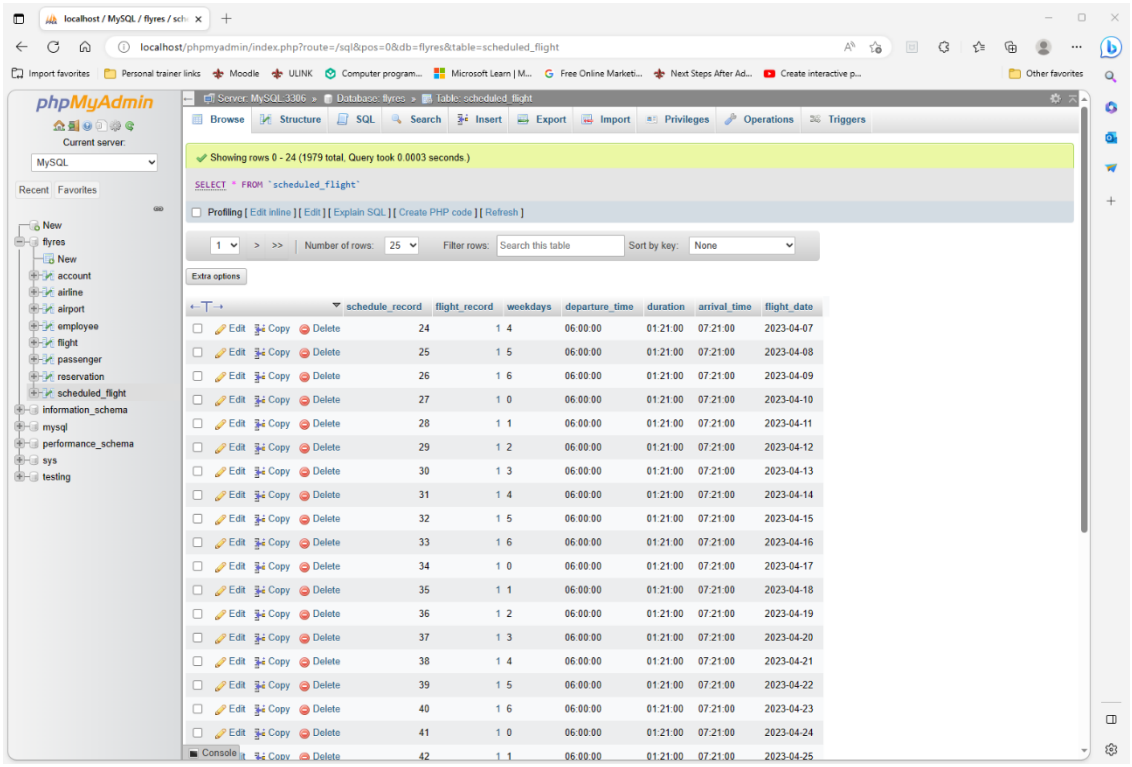
Extra options

	flight_record	flight_number	airline	leg_origin	leg_destination	fc_seats	bc_seats	ec_seats	fc_fare	bc_fare	ec_fare	hidden_fare
Edit Copy Delete	1	AA001	AA	MSY	MCO	12	36	38	800.00	500.00	300.00	250.00
Edit Copy Delete	2	AA001	AA	MCO	MAD	33	48	185	1000.00	800.00	600.00	500.00
Edit Copy Delete	3	AA002	AA	MAD	MCO	33	48	185	1000.00	800.00	600.00	500.00
Edit Copy Delete	4	AA002	AA	MCO	MSY	12	36	38	800.00	600.00	300.00	250.00

Scheduled_flight Table

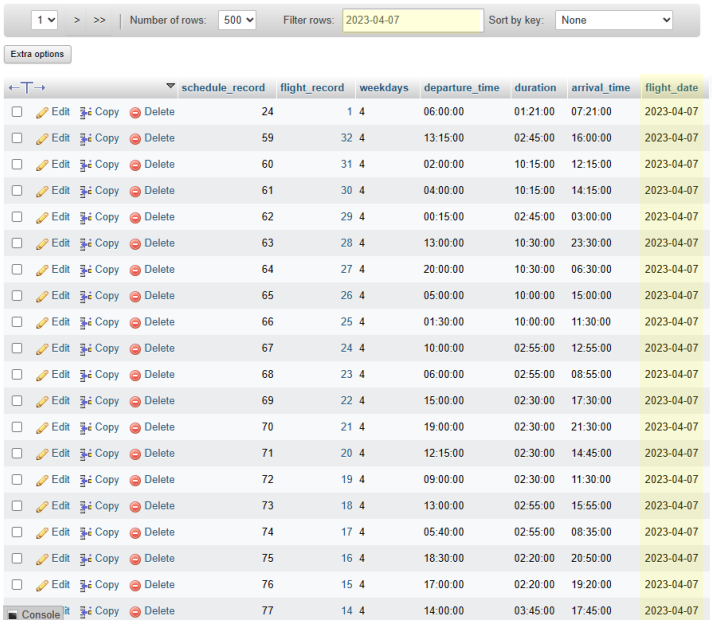
View All Scheduled Flights

The scheduled flight table contains flight schedule information such as flight date, departure times, flight duration, and scheduled arrival times. To view all of the scheduled flights we simply click on scheduled_flight from the main page of the flyres database.



View Scheduled Flights by Date

From the scheduled_flight list we can filter and sort results based on our needs. Let's say for example we want to see all flights departing on 2023-04-07. We simply type this information into the filter by search input and get the following results.



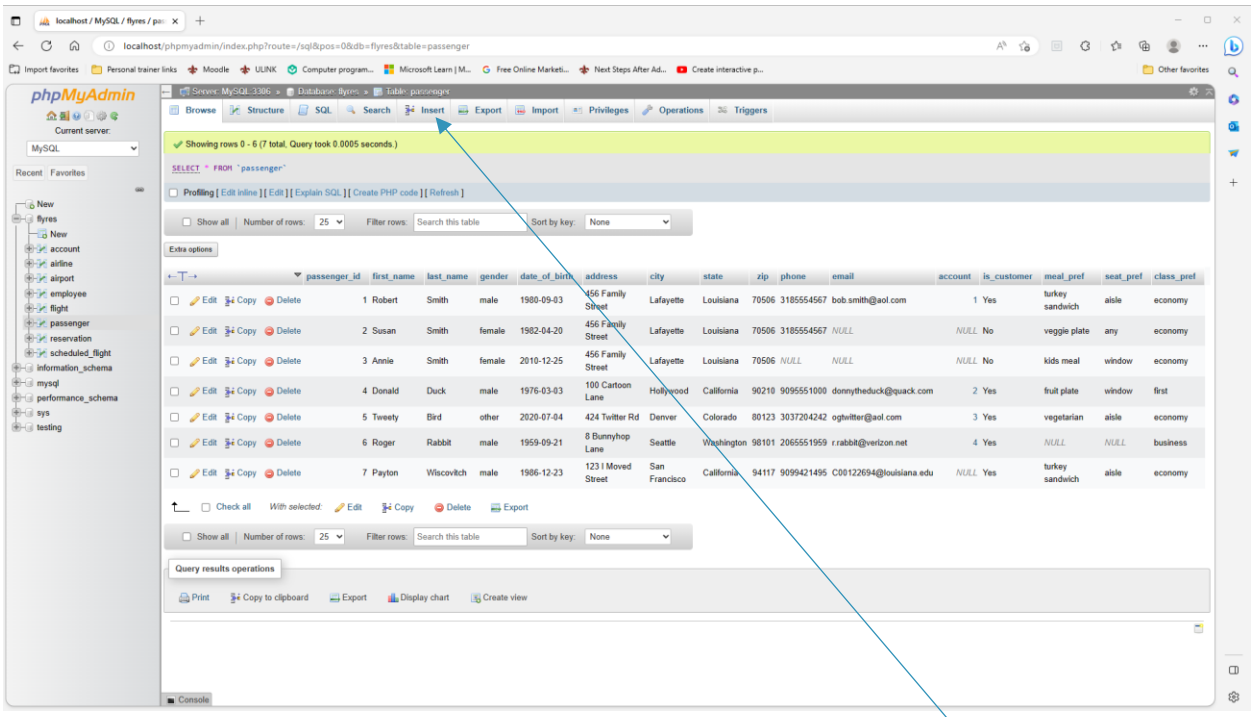
There are a number of ways to sort and filter flight information. The best way is to follow the information provided in this guide and adjust the input information for your specific scenario.

Customers, Passengers, and Accounts

Passenger Table

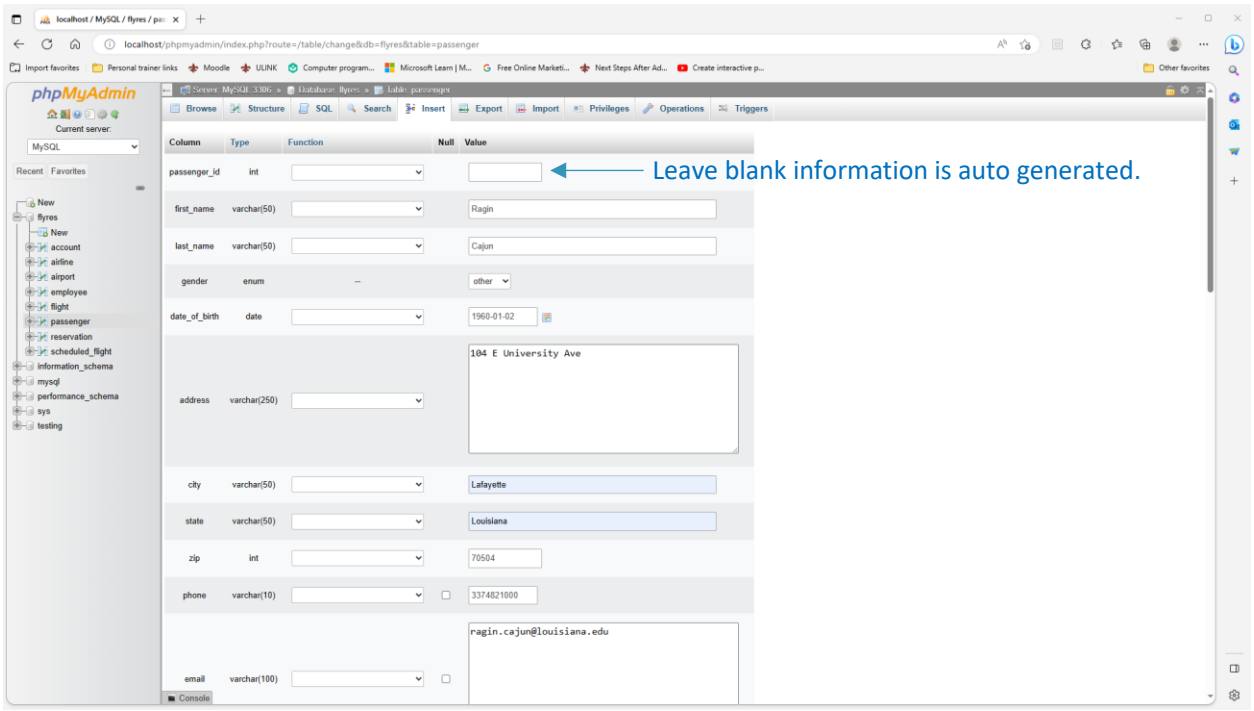
View All Passengers

The passenger table contains information about passengers. Not all passengers are customers. This is the case when one family member is a customer and makes reservations for flights for their family members. Any person traveling on a flight will have an entry in the passenger table. We can view the passenger table in its entirety by clicking on the passenger table from the main page of the flyres database or at any time selecting it from the left side panel under flyres.



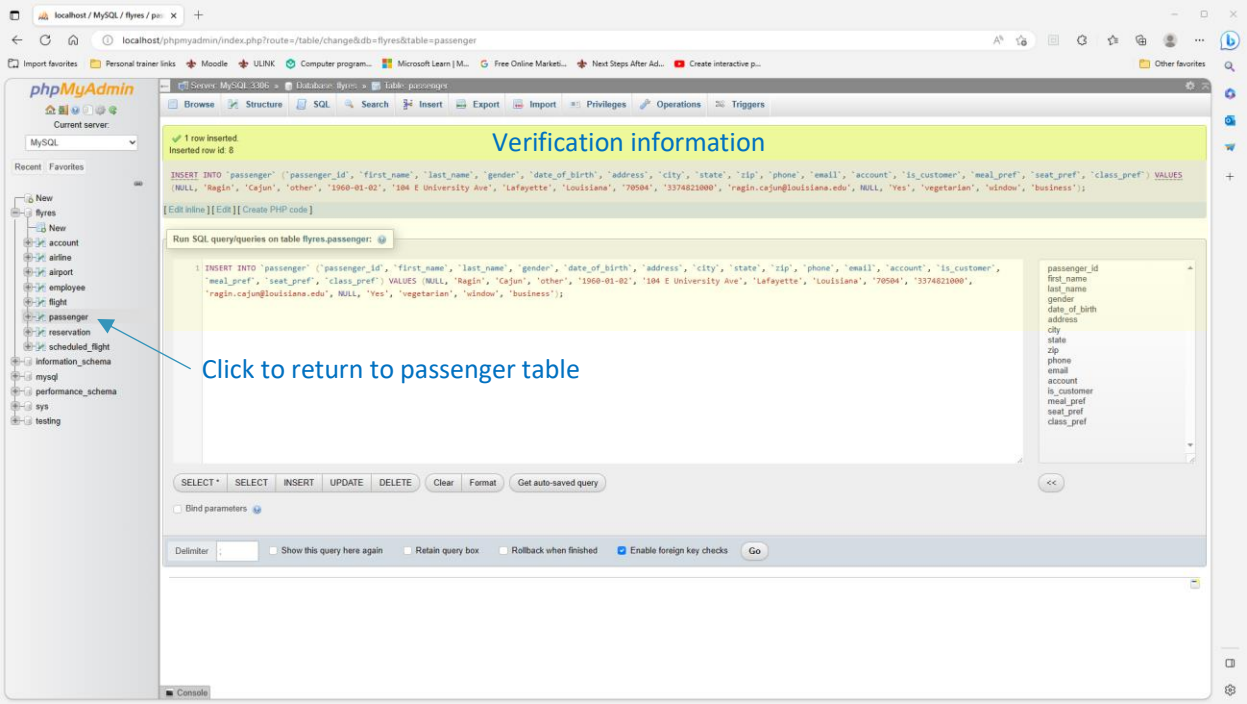
Adding a Passenger

From the main passenger table list we can add a new passenger by selecting the insert tab across the top toolbar. After selecting insert we are presented with the following screen for inputting the passenger information.



Enter the information as described leaving the passenger_id field blank as this information is automatically generated. Leave the account field blank as we will address creating a new account and linking it to the

customer later in this section. Under the is_customer field select ‘Yes’ if the passenger is the one creating the reservation. Select ‘No’ if this entry is for a passenger who is not the individual creating the reservation. When finished, click ‘Go’ to save the entry in the passenger table. After selecting ‘Go’ we are presented with the following screen which confirms all of the information we just entered. To see the newly updated passenger table we can again click on the passenger table on the left side panel under flyres to view the entire passenger table.

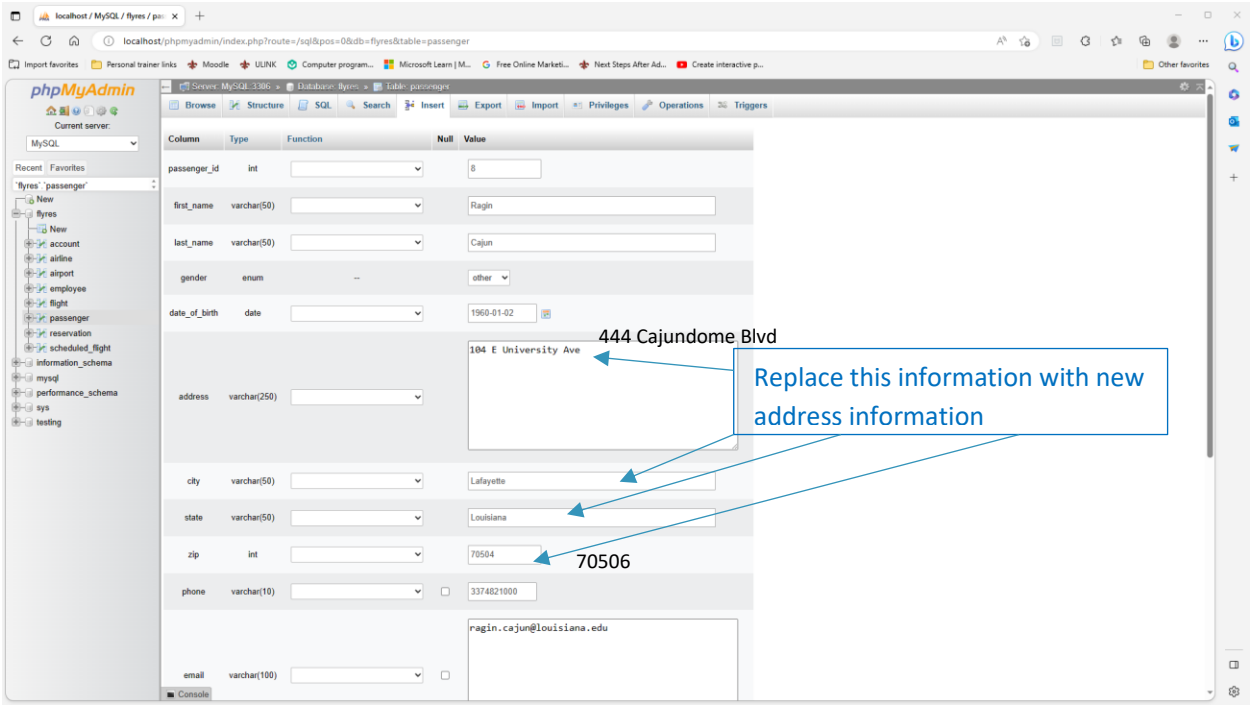


The updated passenger table with our new customer/passenger is shown below.

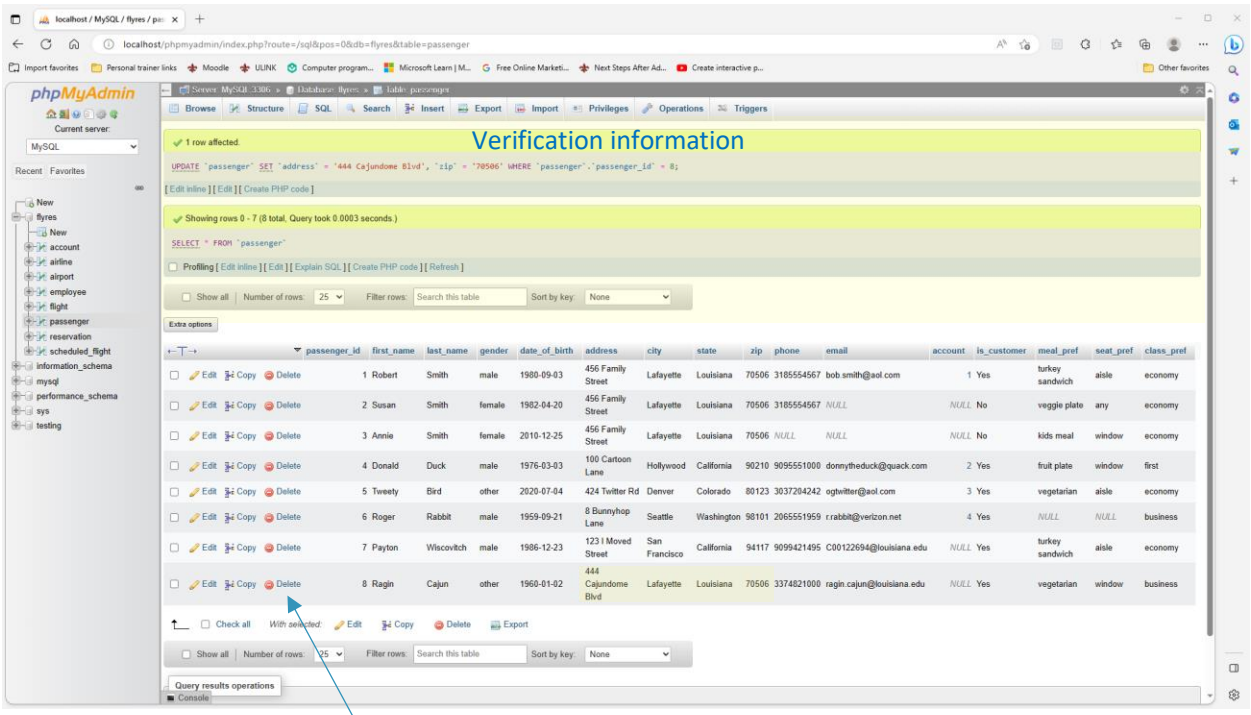
	passenger_id	first_name	last_name	gender	date_of_birth	address	city	state	zip	phone	email	account	is_customer	meal_pref	seat_pref	class_pref
Edit Copy Delete	1	Robert	Smith	male	1980-09-03	456 Family Street	Lafayette	Louisiana	70506	3185554567	bob.smith@aol.com	1	Yes	turkey sandwich	aisle	economy
Edit Copy Delete	2	Susan	Smith	female	1982-04-20	456 Family Street	Lafayette	Louisiana	70506	3185554567	NULL	NULL	No	veggie plate	any	economy
Edit Copy Delete	3	Annie	Smith	female	2010-12-25	456 Family Street	Lafayette	Louisiana	70506	NULL	NULL	NULL	No	kids meal	window	economy
Edit Copy Delete	4	Donald	Duck	male	1976-03-03	100 Cartoon Lane	Hollywood	California	90210	9095551000	donnytheduck@quack.com	2	Yes	fruit plate	window	first
Edit Copy Delete	5	Tweety	Bird	other	2020-07-04	424 Twitter Rd	Denver	Colorado	80123	3037204242	ogtwitter@aol.com	3	Yes	vegetarian	aisle	economy
Edit Copy Delete	6	Roger	Rabbit	male	1959-09-21	8 Bunnyhop Lane	Seattle	Washington	98101	2065551959	r.rabbit@verizon.net	4	Yes	NULL	NULL	business
Edit Copy Delete	7	Payton	Wiscovitch	male	1986-12-23	123 I Moved Street	San Francisco	California	94117	9099421495	C00122694@louisiana.edu	NULL	Yes	turkey sandwich	aisle	economy
Edit Copy Delete	8	Ragin	Cajun	other	1960-01-02	104 E University Ave	Lafayette	Louisiana	70504	3374821000	ragin.cajun@louisiana.edu	NULL	Yes	vegetarian	window	business

Edit Passenger Information

From the main passenger table, we can edit passenger information. Let’s take our newest passenger Ragin Cajun for example. The passenger has moved to a new address, and we need to change the associated information. We can edit the passenger information by clicking the Edit link located next to the pencil icon to the left of the passenger entry. This brings us back to our insert page where we can adjust the information accordingly and select ‘Go’ to save.



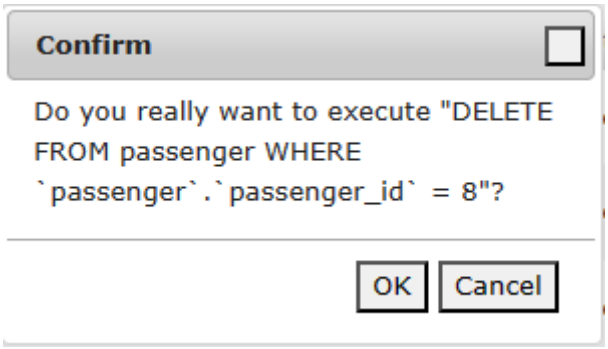
The entry has now been updated with the new address for the passenger.



Deleting a Passenger

Passengers can be deleted from the passenger table using the Delete link to the left of the passenger entry. However, once this information is deleted it cannot be recovered. So, it is important that we are sure this is

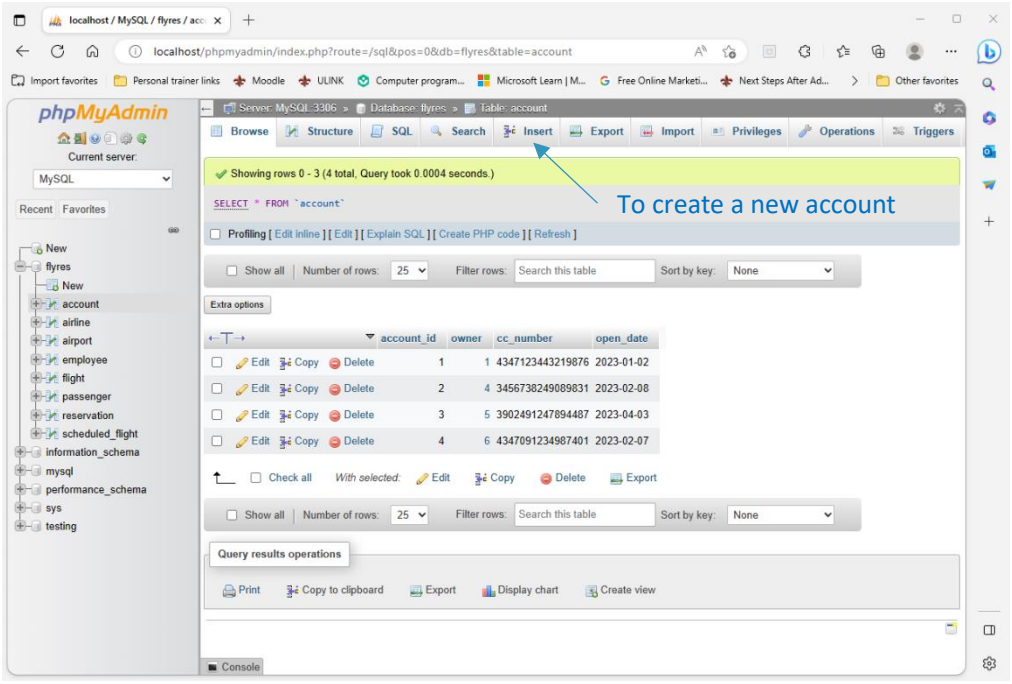
necessary before performing a delete procedure. We are presented with the following prompt when clicking Delete.



Again, it is important to be sure that we want to delete the selected information before confirming using the OK button shown above. Once OK is clicked the record will be permanently deleted.

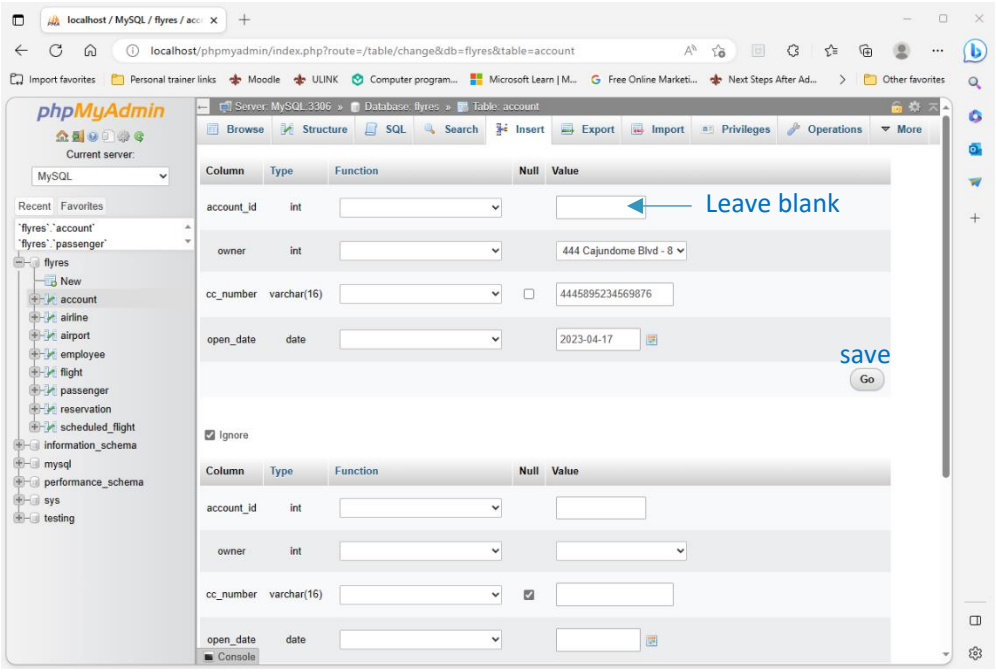
Creating an Account

Each customer can have one or more accounts. This may be the case if the same customer makes reservations for both leisure travel and business travel. We can create a new account by first opening the account table by selecting account from the left side panel or from the main page of the flyres database.

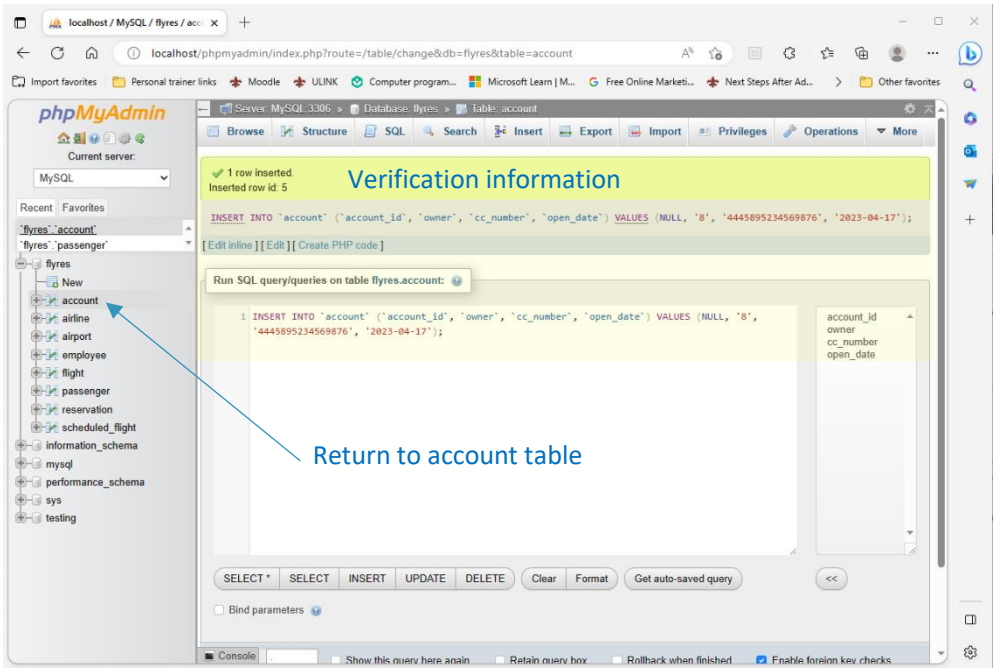


From here, we can select the Insert tab along the top toolbar to create a new account. We need to know the passenger_id of the person we are creating the account for to link the account to the owner. In this example, we are creating an account for our newest customer Ragin Cajun passenger_id 8. You will enter the information as described, leaving the account_id field blank as this information is autogenerated. Under owner, select the correct passenger_id from the drop-down menu (in this case 8 with the corresponding street address of Ragin

Cajun). Type in the customer's credit card number and then select the current date from the calendar and select 'Go' to save.



We are then taken to a screen which verifies the information we have just entered. To return to the account table select account from the left side panel under flyres.

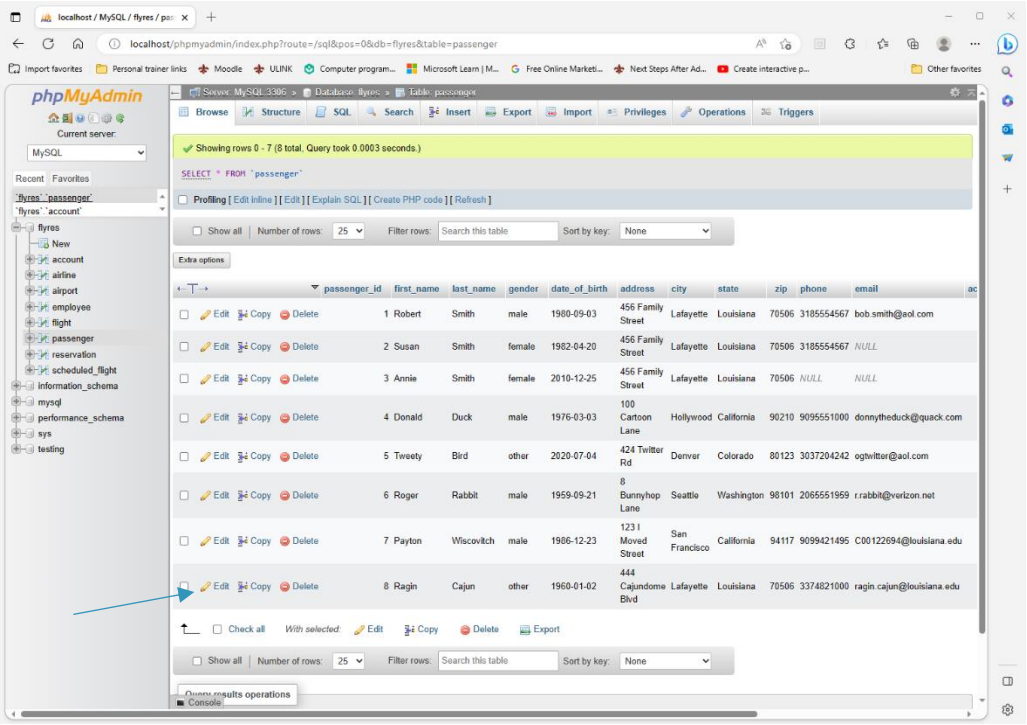


Our newest entry is now shown at the bottom of the account table.

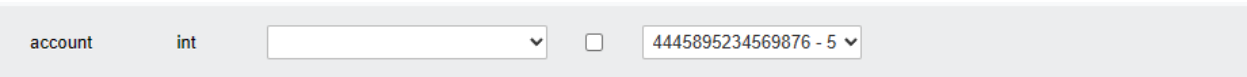
	account_id	owner	cc_number	open_date
Edit Copy Delete	1	1	4347123443219876	2023-01-02
Edit Copy Delete	2	4	3456738249089831	2023-02-08
Edit Copy Delete	3	5	3902491247894487	2023-04-03
Edit Copy Delete	4	6	4347091234987401	2023-02-07
Edit Copy Delete	5	8	4445895234569876	2023-04-17

Linking an Account

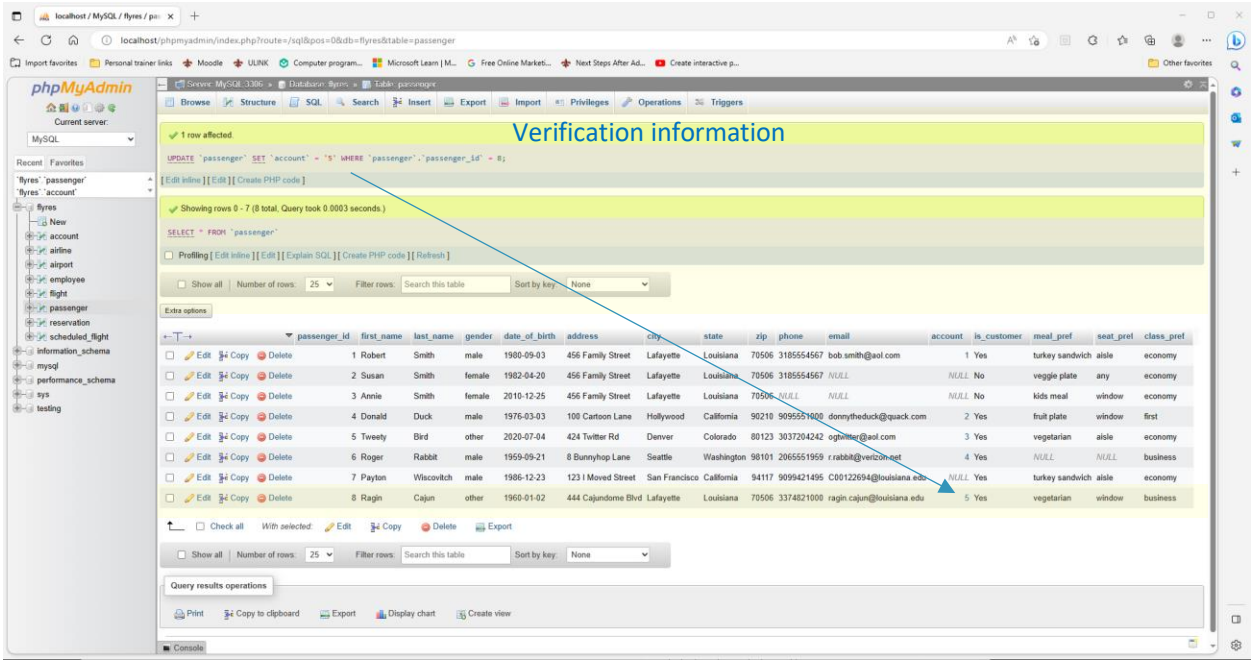
Now that we have created a new account, we need to link the account to the passenger table. In our working example, our new account_id is 5 and belongs to passenger_id 8. We need to go back to the passenger table and edit the entry for passenger_id 8 to link the new account to the customer that owns the account. We open the passenger table by selecting passenger from the left side panel under flyres. We then select Edit for passenger_id 8.



Under the insert page we select account_id 5 from the drop-down menu to associate account 5 with passenger 8.



Then we select ‘Go’ to save this information.



We have now created a new passenger/customer entry, edited the passenger information, created an account for the new customer, and linked the account to the customer.

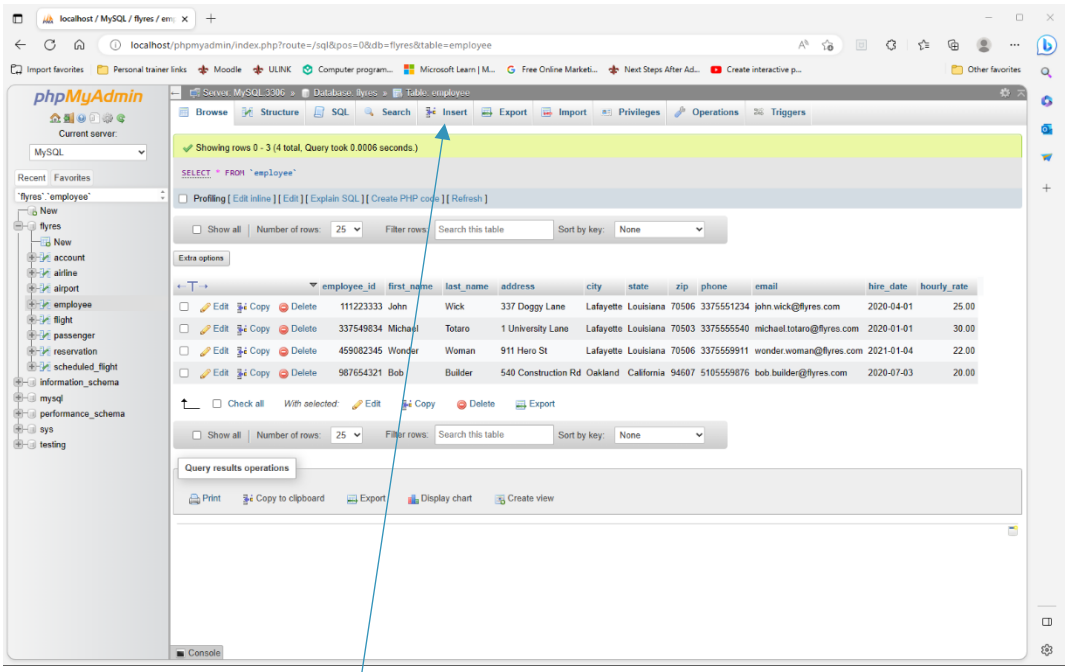
Employees

Employee Table

The employee table has information about the customer representatives employed by FlyRes. The employee_id field is the employee's social security number and is a unique identifier for the employee table. All fields within the employee table are required fields and will require input for adding and saving new employees to the table.

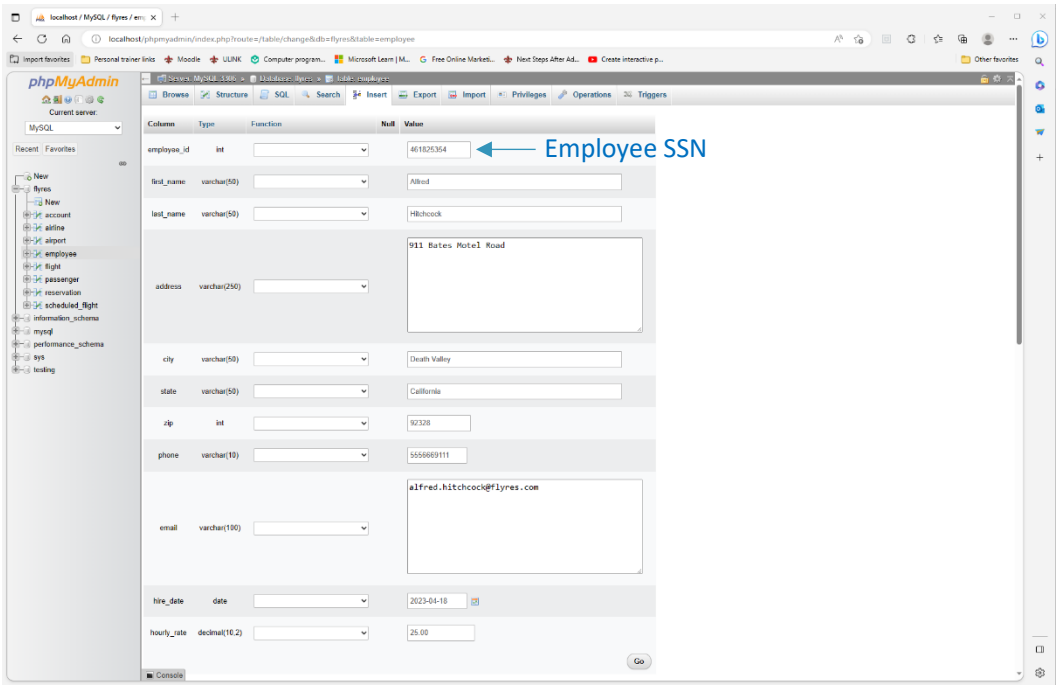
View All Employees

To view all employees, we can select the employee table by clicking on employee in the left side panel or from the main page of the flyres database.

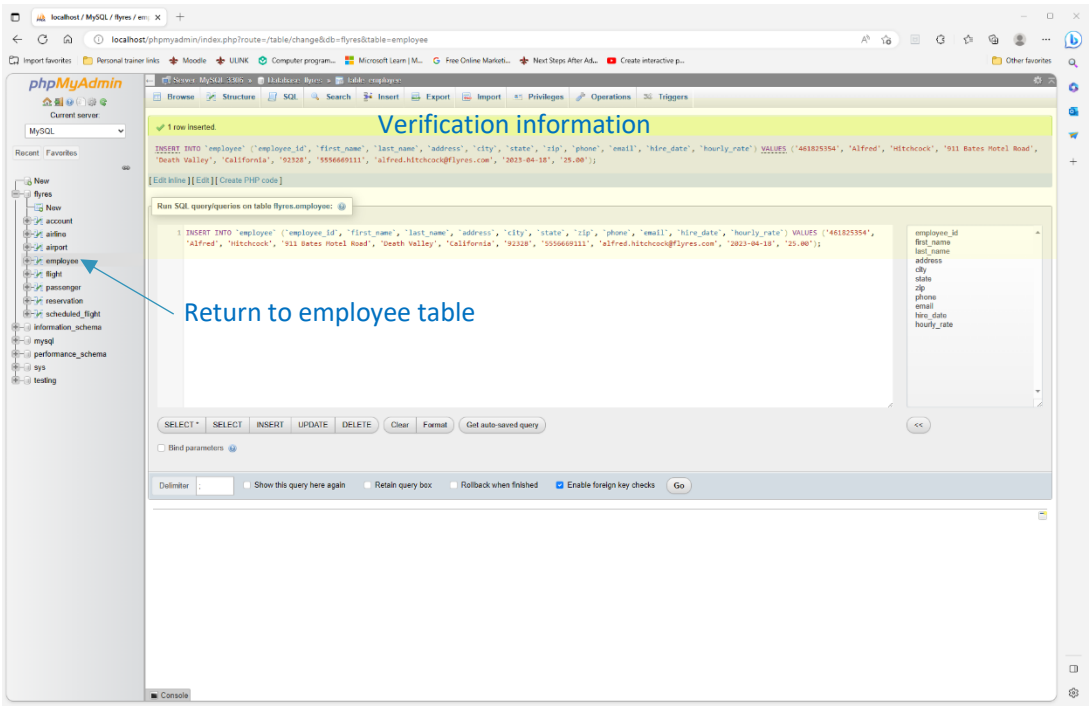


Adding a New Employee

After selecting the employee table from the left side panel or the main page of the flyres database, a new employee can be added by clicking the insert tab from the employee table main page.

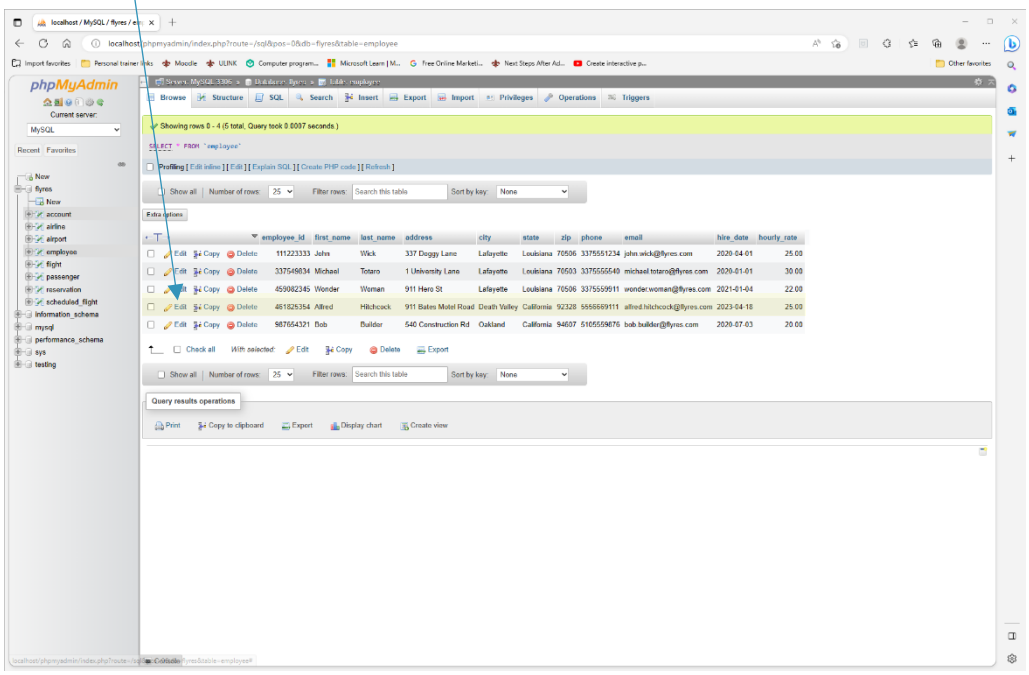


The above input screen appears for adding a new employee. All fields are required. Enter all information as indicated and click ‘Go’ to save.

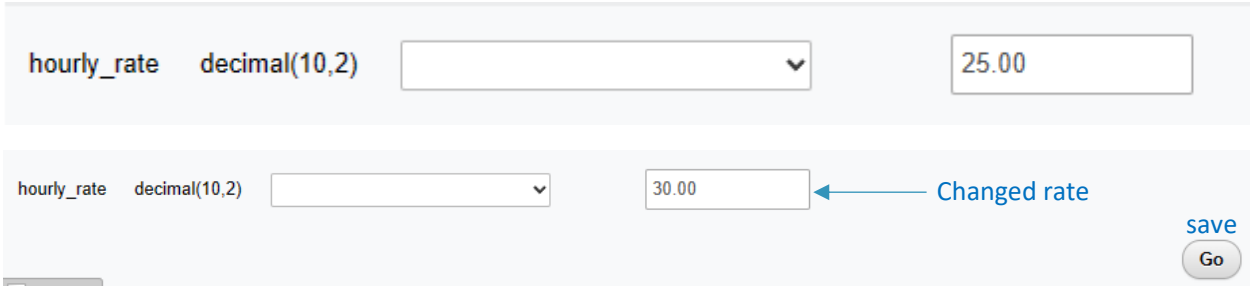


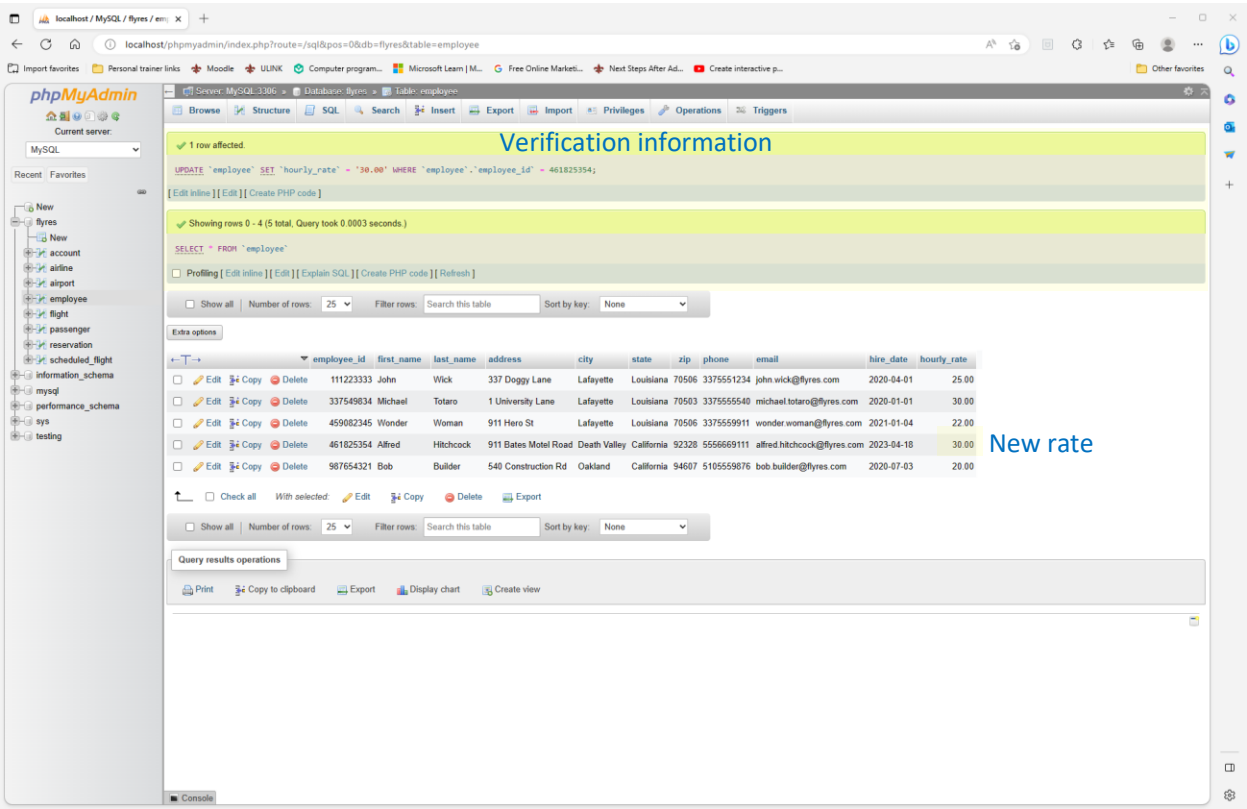
Editing Employee Information

Employee information can be edited by selecting Edit from the employee table main page. For example, if one of our existing employees has received a raise and we need to change their hourly pay rate. To demonstrate this we will change the pay rate of our newest employee Alfred Hitchcock by selecting Edit next to the entry for this employee.



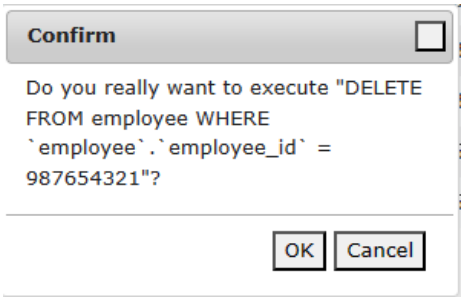
Once selecting edit next to the employee's entry, we are taken back to the insert screen where we can edit the current information. We will change the hourly rate and then select 'Go' to save the change.





Deleting an Employee

From the main page of the employee table an employee can be permanently deleted by selecting the Delete link next to the desired entry. This action is permanent, and we need to be certain that this is the desired outcome before completing this action. After selecting Delete we will be presented with the following prompt:



Before selecting OK it is important to be absolutely sure that the desired result is to permanently remove all information for this employee from the database.

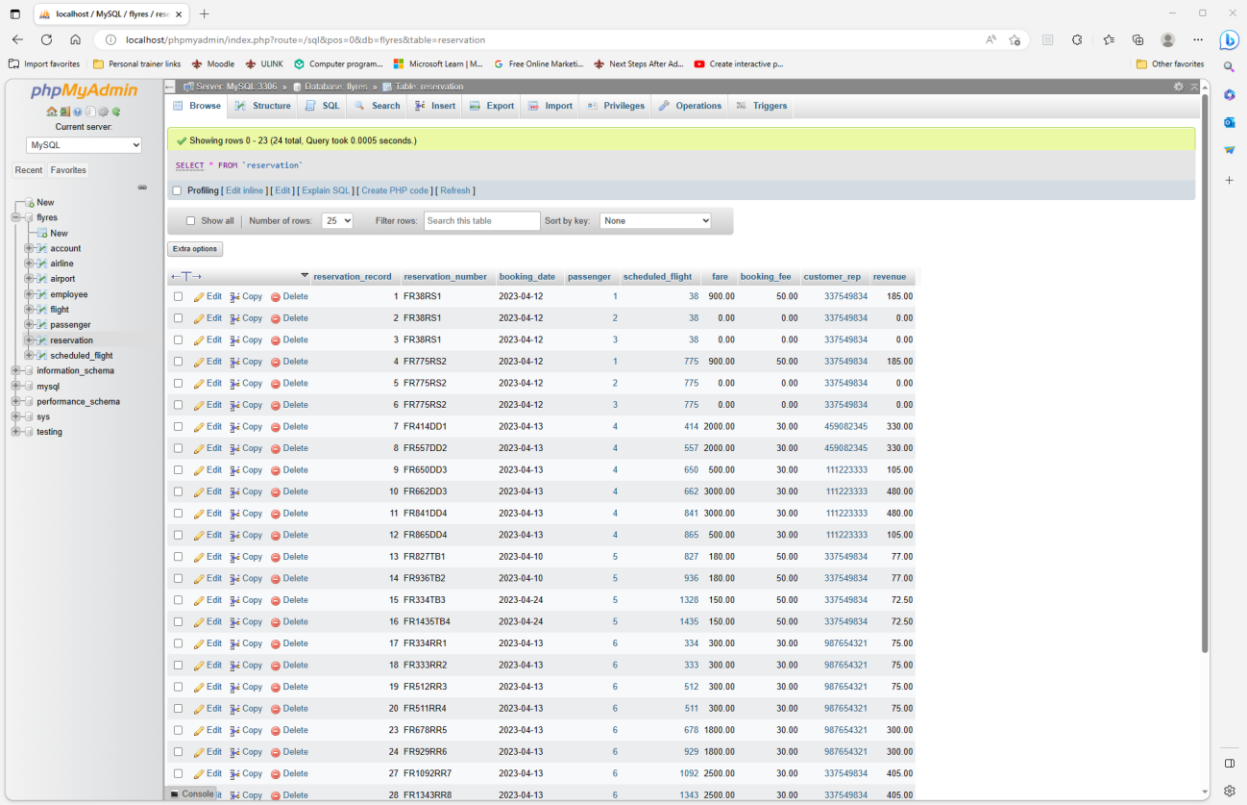
Reservations

Reservation Table

The reservation table shows all reservations in the FlyRes system. The reservation table houses information from multiple different tables such as the passenger table, scheduled flight table, and the employee table. Each reservation is for a passenger, for a specific scheduled flight, and created by a customer representative. Reservations linked to passengers who are also customers will have fare information, booking fees, and total revenue gained from this specific reservation. A reservation record is a specific instance of a reservation while a reservation number can be shared among passengers traveling together under one reservation. For example, if a family is traveling together the reservation number is shared among all passengers in this group.

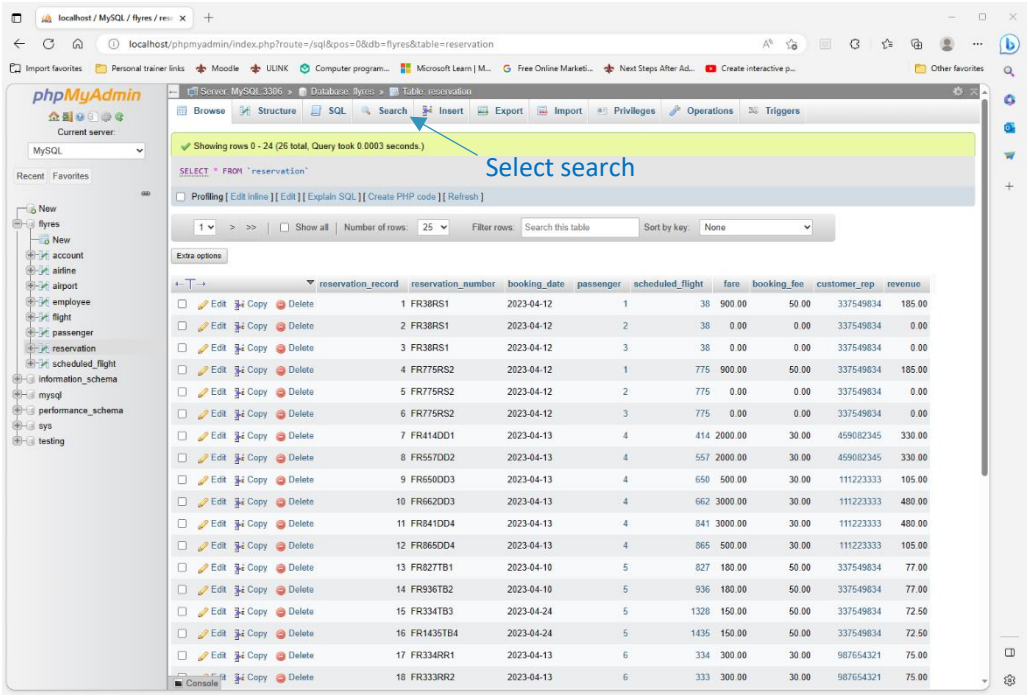
View All Reservations

In order to view all reservations in the system, we can simply select the reservation table under flyres from the left side panel or from the main page of the flyres database.

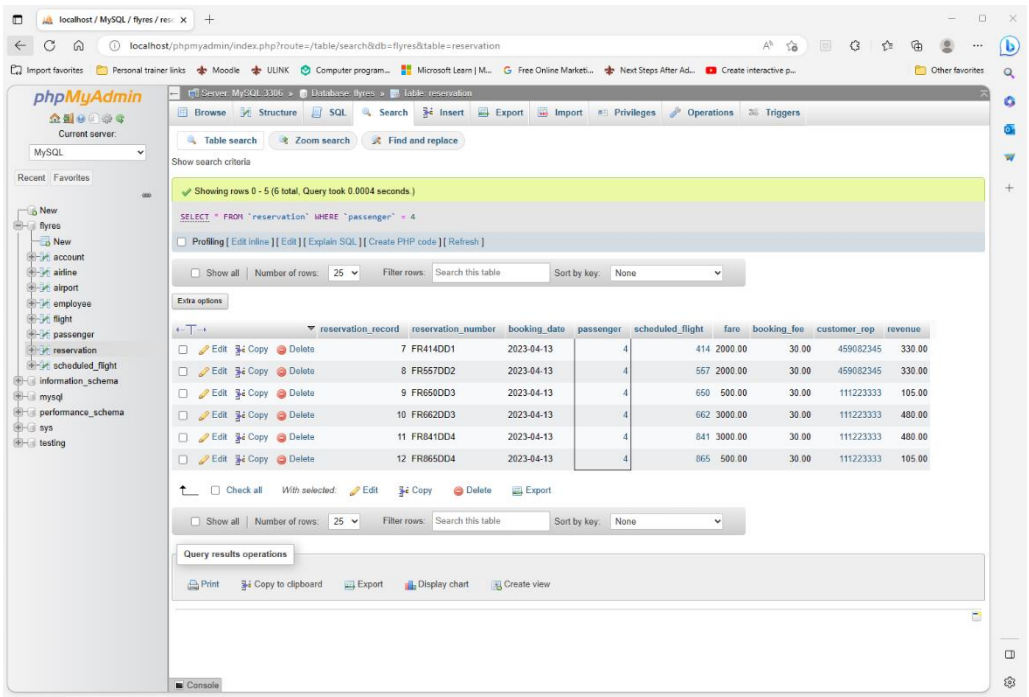


View Reservation by [parameter]

We can view all the reservations for a particular parameter by searching for the parameter in the reservation table. For example, if we want to view all reservations for a specific passenger. We have a passenger named Donald Duck his passenger_id is 4. We can view all of Donald's reservations by selecting the reservation table in the left side panel and then using the search tab along the top toolbar of the reservation table.



On the search input the passenger_id by selecting from the drop down menu for passenger. Then, select 'Go' to execute the search which will return all reservations for passenger_id 4, Donald Duck.



Creating a Reservation

Creating a reservation requires gathering information from multiple tables. Each reservation will require knowing the passenger_id of each passenger, the flight_record for the preferred flight and flight date, and the employee_id of the customer representative creating the reservation. We will also need to note the fare for

each flight reserved. Booking fees are based on the fare. Any fare < 500 has a $50 booking fee while any fare > 500 has a booking fee of $30. Fares and booking fees are only charged to the customer. So, passengers that are not also customers will not have any charges associated with their reservation.

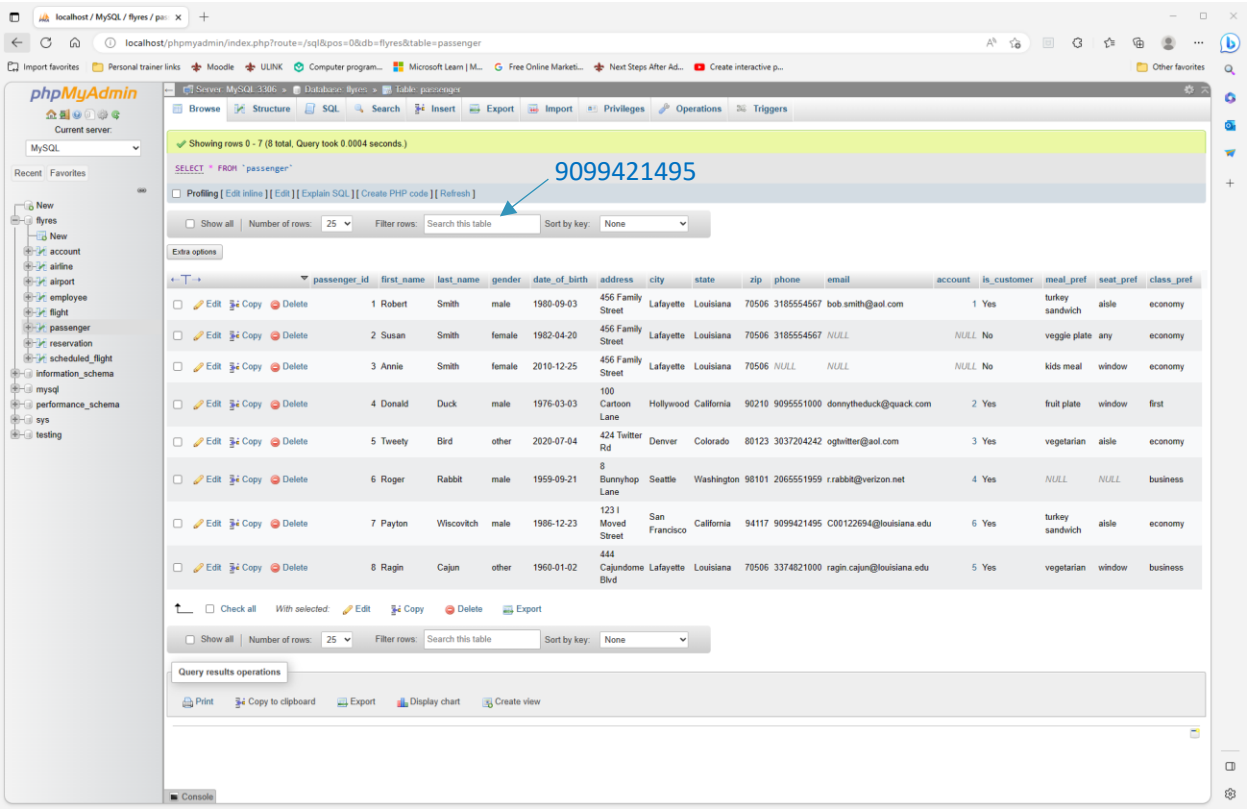
Reservation Scenario

Passenger Payton Wiscovitch wants to make a roundtrip reservation from San Francisco to Denver and back leaving on 5/01/2023 and returning 5/07/2023.

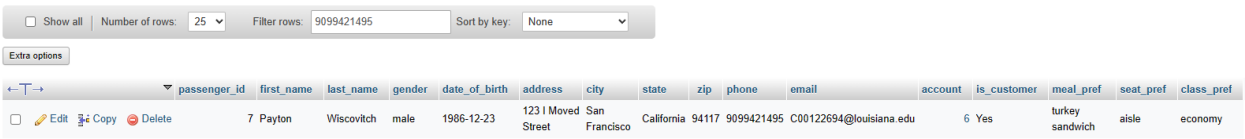
Passenger Look-up

We need to gather the required information about our passenger. We will open the passenger table and search for our passenger. We can use the passenger's name to look them up, but there may be multiple passengers with the same name. So, we will try a unique identifier such as the phone number or email address associated with the passenger. In this case, Payton Wiscovitch is a unique name, but for demonstration purposes, let's look him up using the phone number 9099421495.

1. Open the passenger table by selecting passenger on the left side panel listed under flyres.
2. In the Filter rows: input where it says "Search this table" type in the passenger phone number.



3. Note the results of the passenger information.



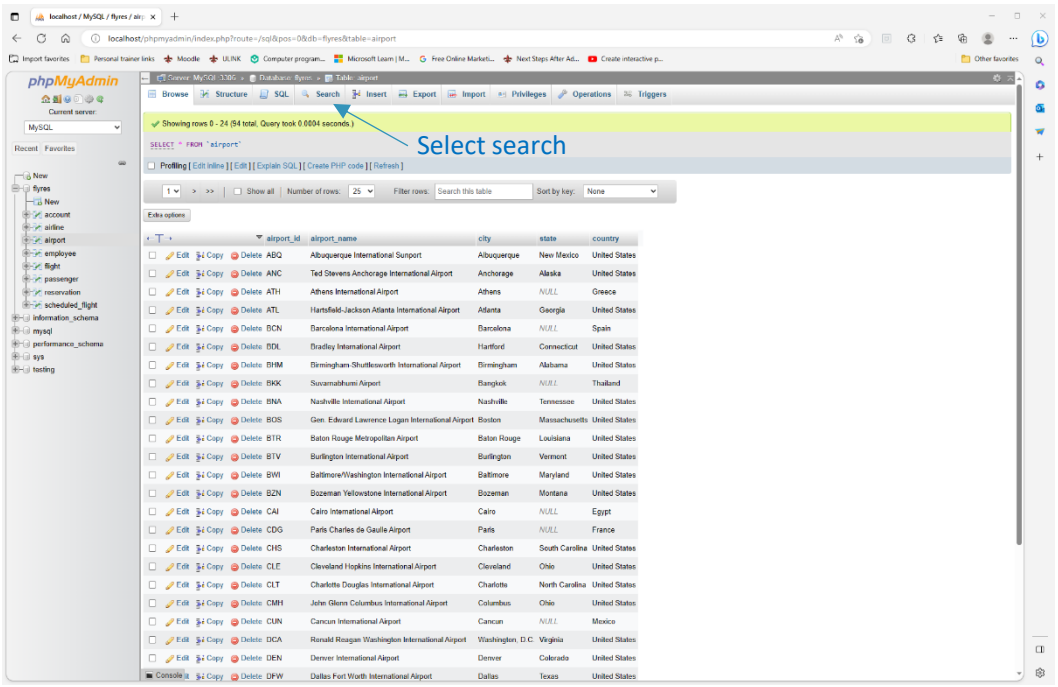
4. Passenger_id = 7, class preference is economy

Gathering Flight Information

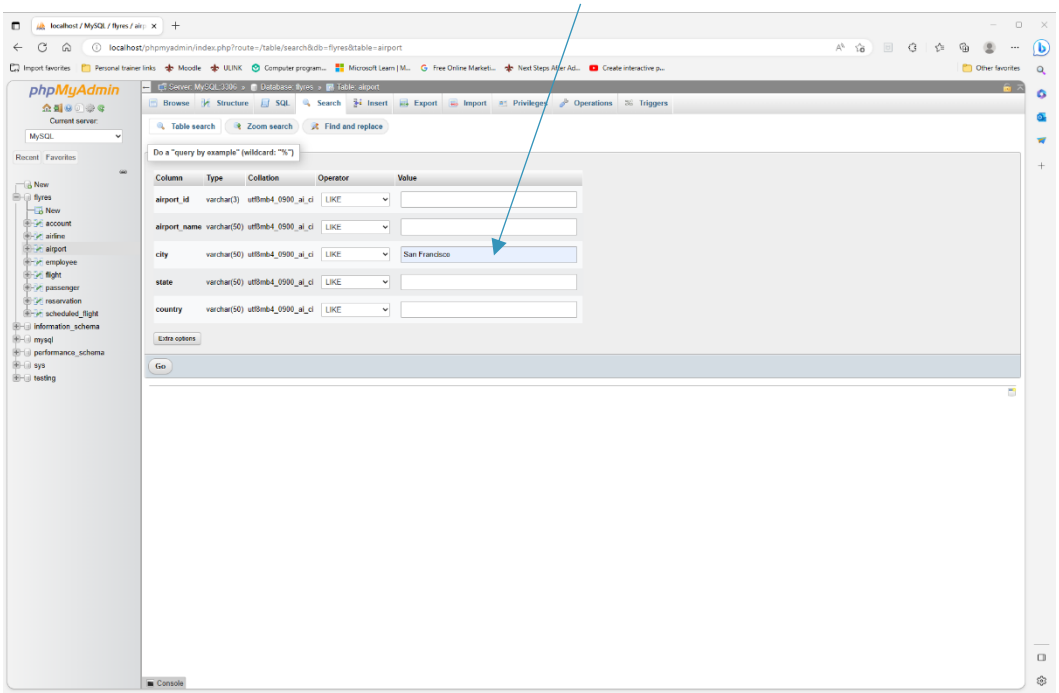
We are looking for flights from San Francisco to Denver departing on 5/1/2023 and flights from Denver to San Francisco departing on 5/7/2023.

Airport Code

1. The first step is noting the airport codes for San Francisco and Denver. If you know the airport codes skip to the next section. To do this, open the airport table by selecting airport from the left side panel.

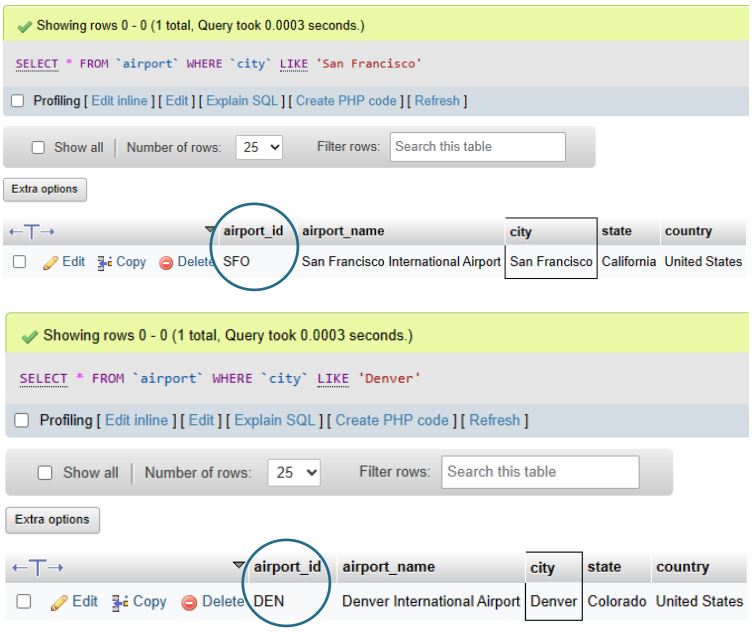


Then select the search tab along the top toolbar. From the search input screen under city type in San



Francisco and select ‘Go’ to search.

2. In the results, note the airport code for San Francisco. To search for the Denver airport code repeat steps 1 and 2 entering Denver in the city search.

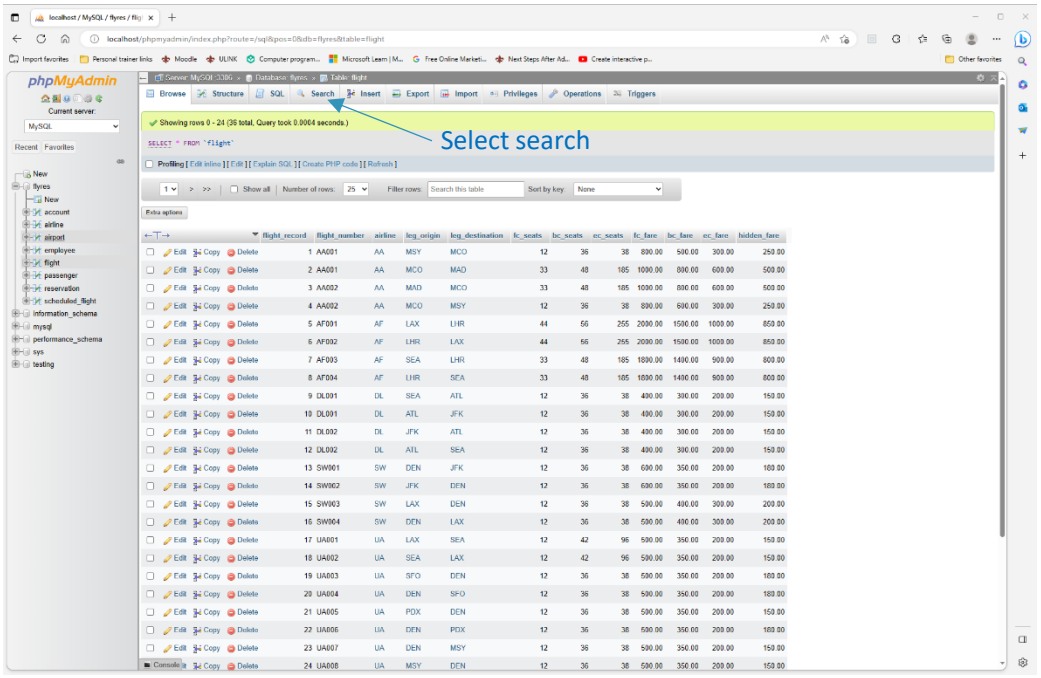


3. Now we know we are looking for flights with origin/destination of SFO and DEN.

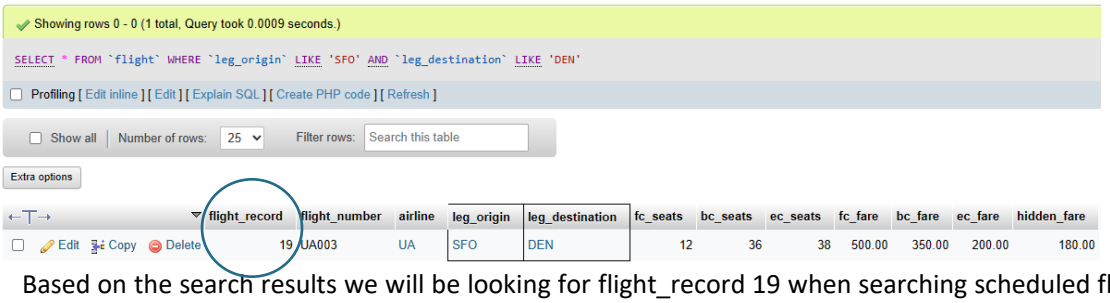
Flight Record

Our next step is figuring out which flights offer service between SFO and DEN. We are looking for a specific flight_record number for each leg of the trip.

1. Open the flight table by selecting flight from the left side panel.
2. Select the search tab along the top toolbar.



3. In the search input select SFO from the drop-down menu for leg_origin and DEN from the drop down menu for leg_destination. Then select 'Go' to perform the search.



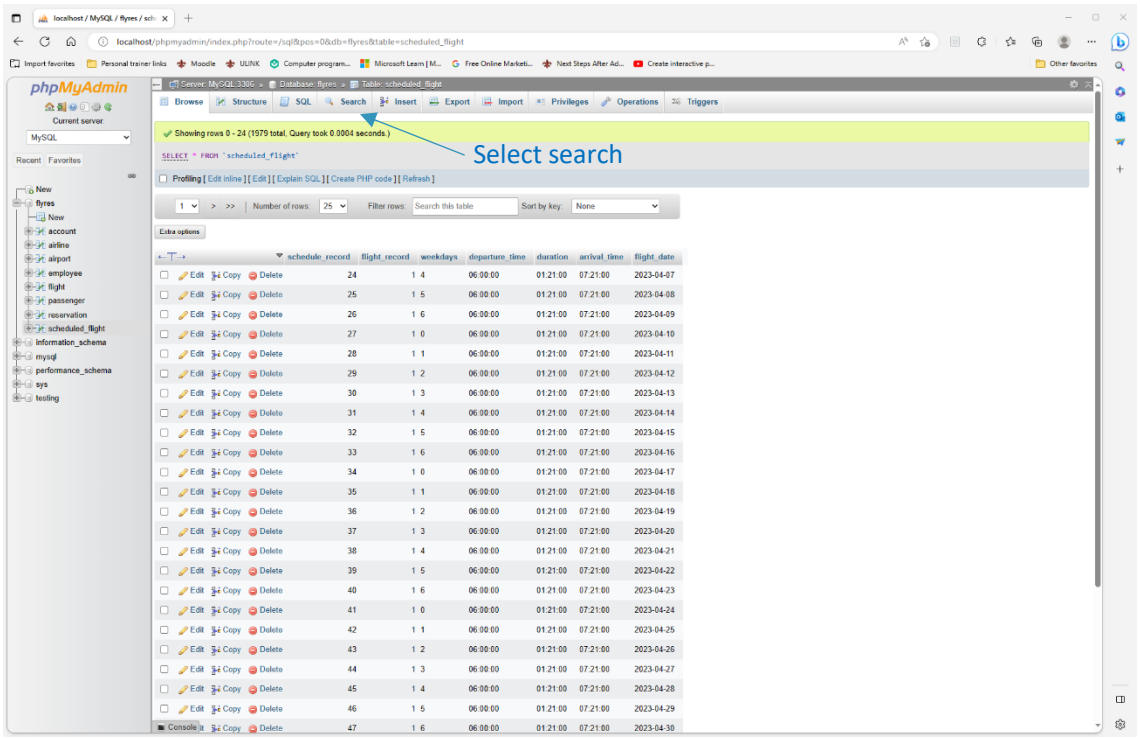
4. Based on the search results we will be looking for flight_record 19 when searching scheduled flights. This is for flight UA003 operating from SFO to DEN. We also note that the cost of an economy class ticket is $200. So, our associated booking fee for this flight would be $50.
5. Repeat steps 1-3 changing the leg_origin to DEN and leg_destination to SFO to search for the return flight for this roundtrip.

6. We note from the results that we are looking up flight_record 20 when searching for scheduled flights for the passenger's return trip. This is for flight UA004 from DEN to SFO. The economy class fare is $200 and will have a $50 booking fee.

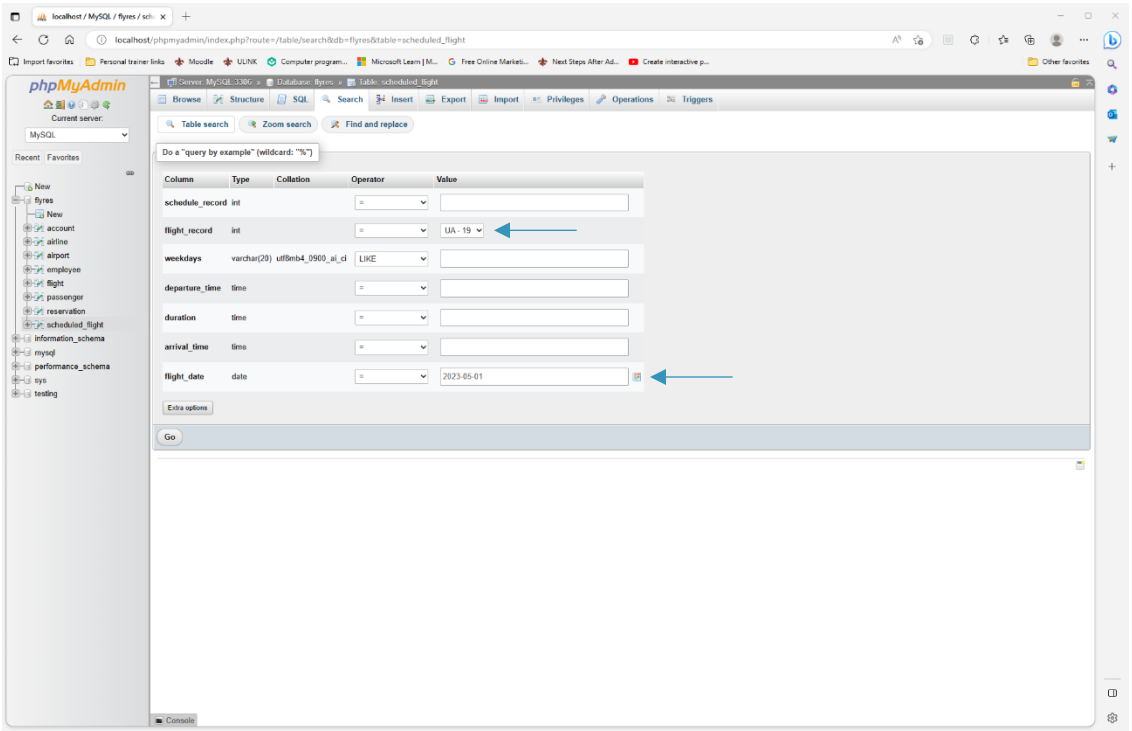
Searching for Scheduled Flights

Now that we know the flight_record number for each of the flights we can use this information to search for scheduled flights on the desired dates. So, we will be searching the scheduled flight table for flight_record 19 departing on 5/1/2023 and flight_record 20 departing on 5/7/2023.

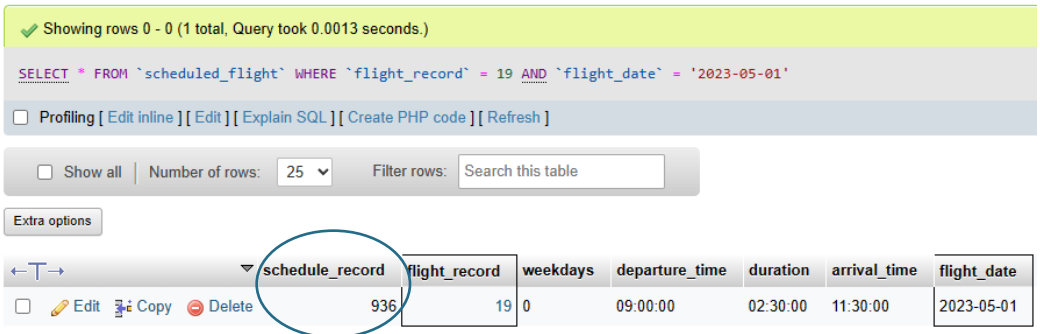
1. Open the scheduled flight table by selecting scheduled_flight from the left side panel of the flyres database. Then select the search tab along the top toolbar to search the scheduled flights.



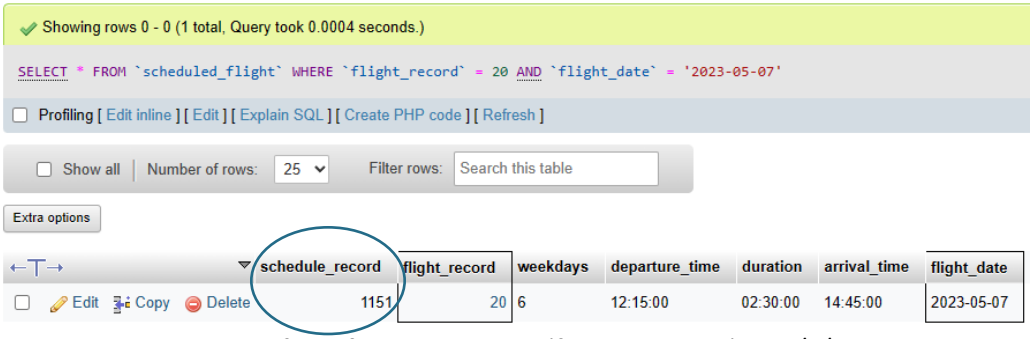
2. From the search input screen select flight_record 19 from the dropdown menu and then select 5/1/2023 from the calendar for flight_date. Then select ‘Go’ to search.



3. Results show that there is one flight from SFO to DEN (flight record 19) on 5/1/2023. The schedule_record is 936. We will use this information when making the reservation.



4. Repeat steps 1-3 for flight_record 20 to search for a return trip on 5/7/2023.



5. Results show there is one flight from DEN to SFO (flight record 20) on 5/7/2023. The schedule record is 1151. We will use this information to make the return flight reservation.

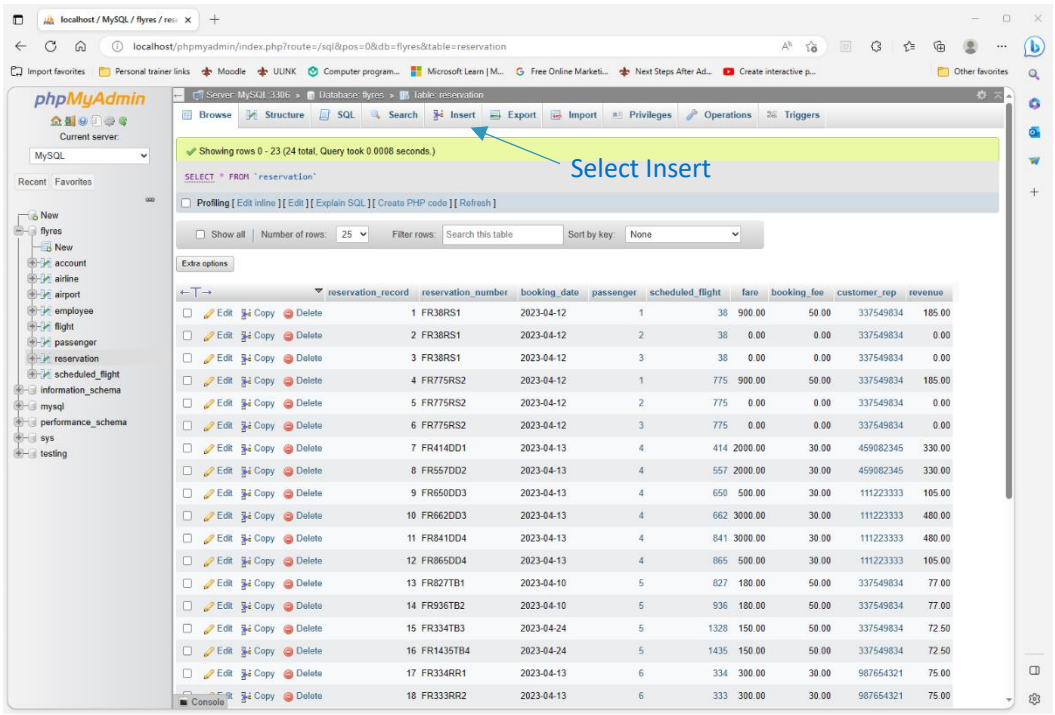
Reviewing all information

From the above searches we have compiled all of the information needed to make the necessary reservations. The passenger_id for Payton Wiscovitch is 7. He is traveling from SFO to DEN on schedule_record 936 on 5/1/2023 on an economy fare of $200 with a $50 booking fee for this flight. He will return on schedule_record 1151 from DEN to SFO on 5/7/2023 on an economy fare of $200 with a booking fee of $50.

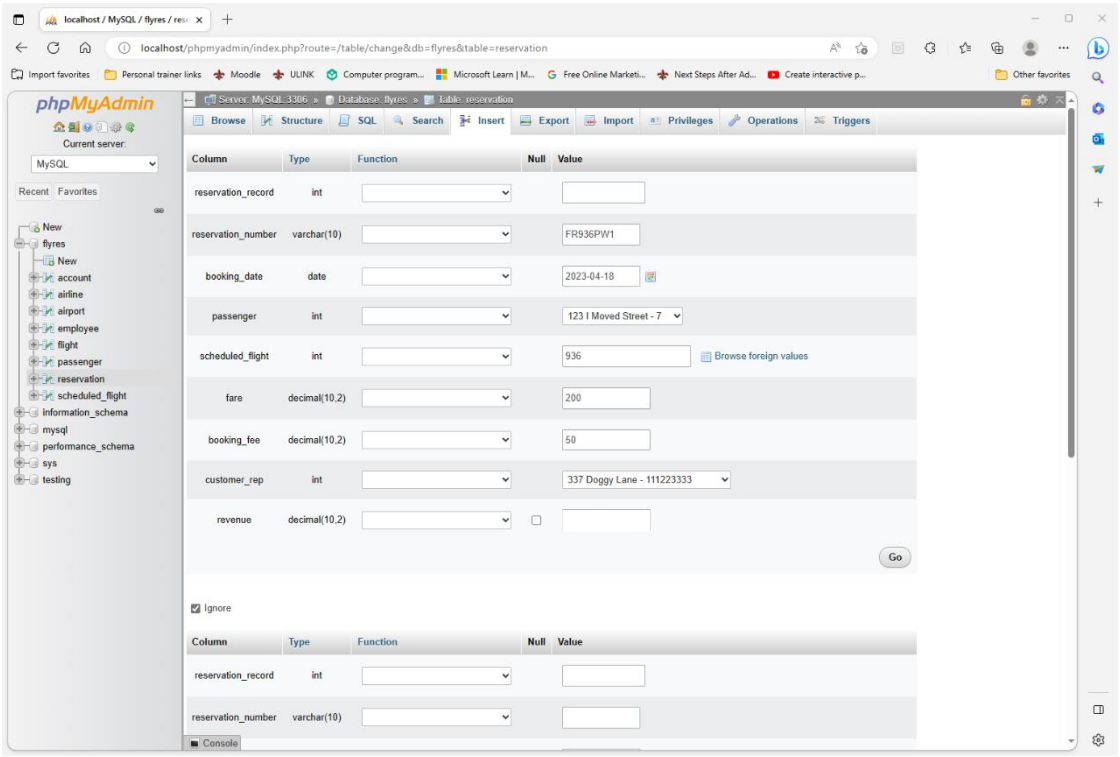
Creating the Reservations

We now have all our input information for creating the reservations for Payton Wiscovitch. We will now create a reservation by inputting this information into the reservation table.

1. Open the reservation table by selecting reservation from the left side panel under flyres.
2. Select the insert tab across the top toolbar of the reservation table.



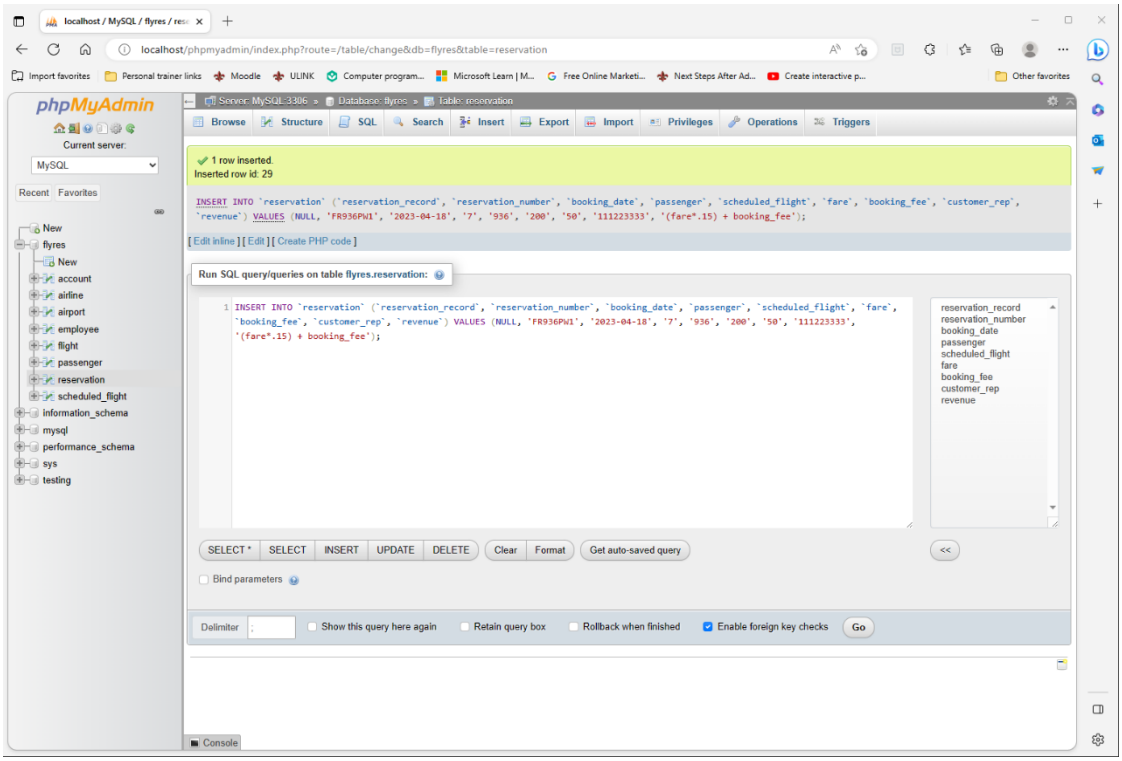
3. On the insert page, leave reservation record blank as this information is automatically generated.



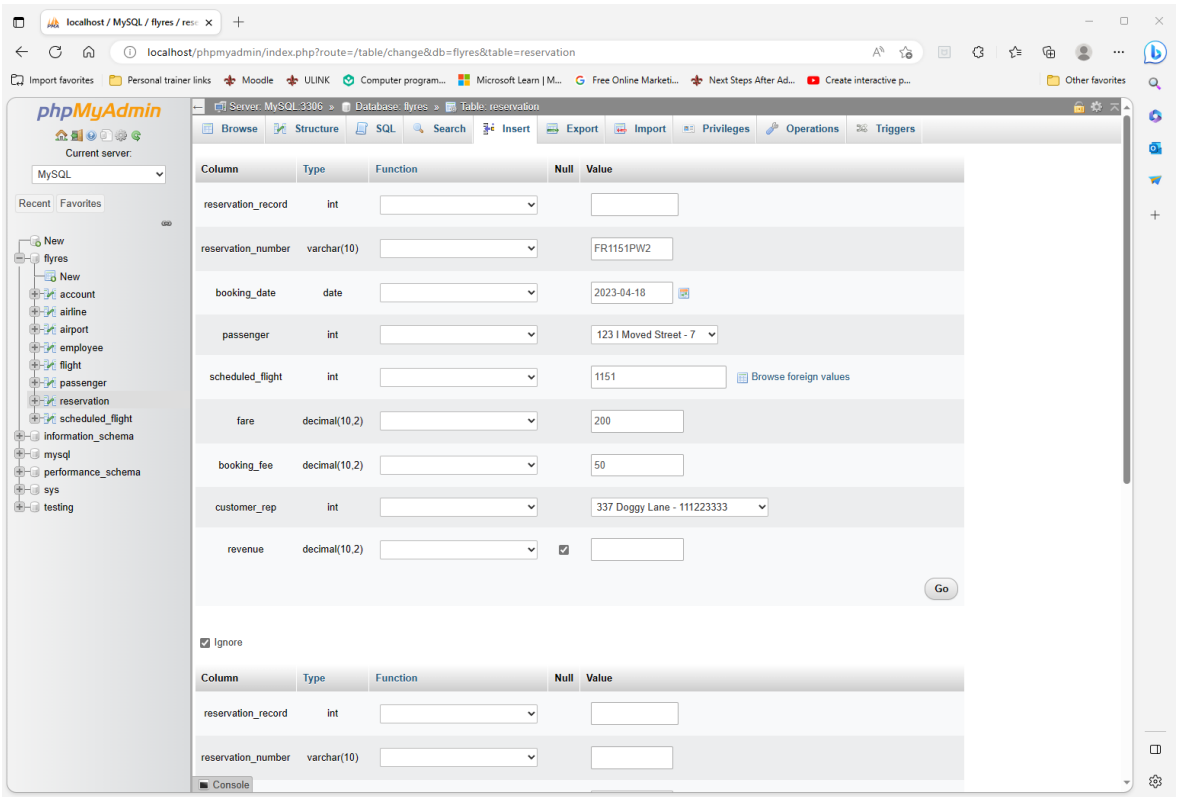
- Reservation number: FR[insert the schedule record number][passenger first and last initial][#of reservation being created in this session]. For this scenario, our first reservation will be for the flight from SFO to DEN on 5/1/23. Reservation number: FR936PW1.
- Select current date as booking date.
- Select passenger number 7 for Payton Wiscovitch from the drop down menu for passenger.

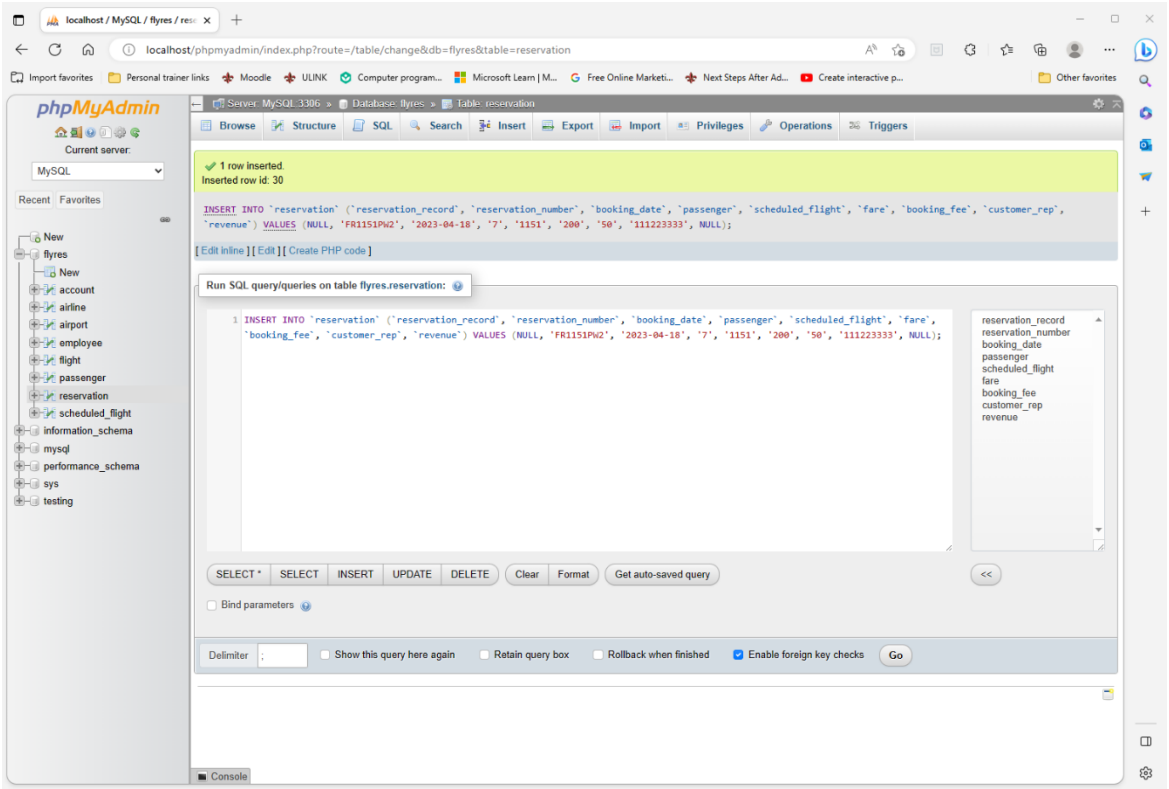
- For scheduled flight type in the flight record number for the first flight: 936.
- For the fare type in the economy class fare for this flight: 200.
- For the booking fee type in the booking fee for this flight: 50.
- Under customer_rep select your employee id.
- Revenue = leave blank this will be calculated later.

Select 'Go' to save.



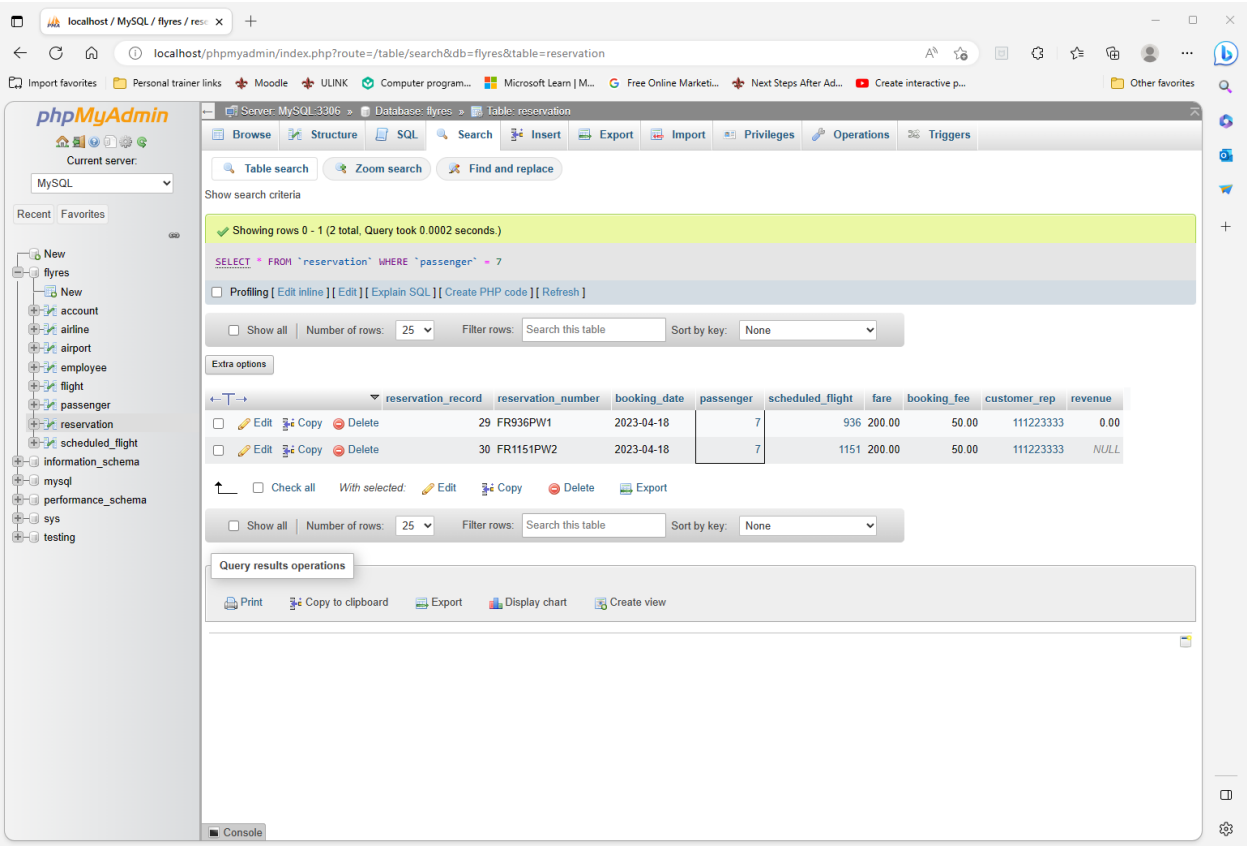
4. This is verification of a saved reservation. Now, repeat steps 1-3 to create the reservation for the return trip for flight record 1151 reservation number FR1151PW2.


localhost / MySQL / flyres / reservation
localhost/phpmyadmin/index.php?route=/table/change&db=flyres&table=reservation
phpMyAdmin
Current server:
MySQL
Recent Favorites
New
flyres
New
account
airline
airport
employee
flight
passenger
reservation
scheduled_flight
information_schema
mysql
performance_schema
sys
testing
Server: MySQL:3306 » Database: flyres » Table: reservation
Browse Structure SQL Search Insert Export Import Privileges Operations Triggers
Column Type Function Null Value
reservation_record int
reservation_number varchar(10) FR1151PW2
booking_date date 2023-04-18
passenger int 123 I Moved Street - 7
scheduled_flight int 1151 Browse foreign values
fare decimal(10,2) 200
booking_fee decimal(10,2) 50
customer_rep int 337 Doggy Lane - 111223333
revenue decimal(10,2)
Go
Ignore
Column Type Function Null Value
reservation_record int
reservation_number varchar(10)
Console


localhost / MySQL / flyres / reservation
localhost/phpmyadmin/index.php?route=/table/change&db=flyres&table=reservation
phpMyAdmin
Current server:
MySQL
Recent Favorites
New
flyres
New
account
airline
airport
employee
flight
passenger
reservation
scheduled_flight
information_schema
mysql
performance_schema
sys
testing
Server: MySQL:3306 » Database: flyres » Table: reservation
Browse Structure SQL Search Insert Export Import Privileges Operations Triggers
1 row inserted.
Inserted row id: 30
INSERT INTO `reservation` (`reservation_record`, `reservation_number`, `booking_date`, `passenger`, `scheduled_flight`, `fare`, `booking_fee`, `customer_rep`, `revenue`) VALUES (NULL, 'FR1151PW2', '2023-04-18', '7', '1151', '200', '50', '111223333', NULL);
[Edit inline] [Edit] [Create PHP code]
Run SQL query/queries on table flyres.reservation:
1 INSERT INTO `reservation` (`reservation_record`, `reservation_number`, `booking_date`, `passenger`, `scheduled_flight`, `fare`, `booking_fee`, `customer_rep`, `revenue`) VALUES (NULL, 'FR1151PW2', '2023-04-18', '7', '1151', '200', '50', '111223333', NULL);
reservation_record
reservation_number
booking_date
passenger
scheduled_flight
fare
booking_fee
customer_rep
revenue
SELECT * SELECT INSERT UPDATE DELETE Clear Format Get auto-saved query
Bind parameters
Delimiter ; Show this query here again Retain query box Rollback when finished Enable foreign key checks Go
Console

Viewing New Reservations

We can now view our new reservations by returning to the main reservation table and performing a search for passenger 7, Payton Wiscovitch. This will return all reservations for this passenger.



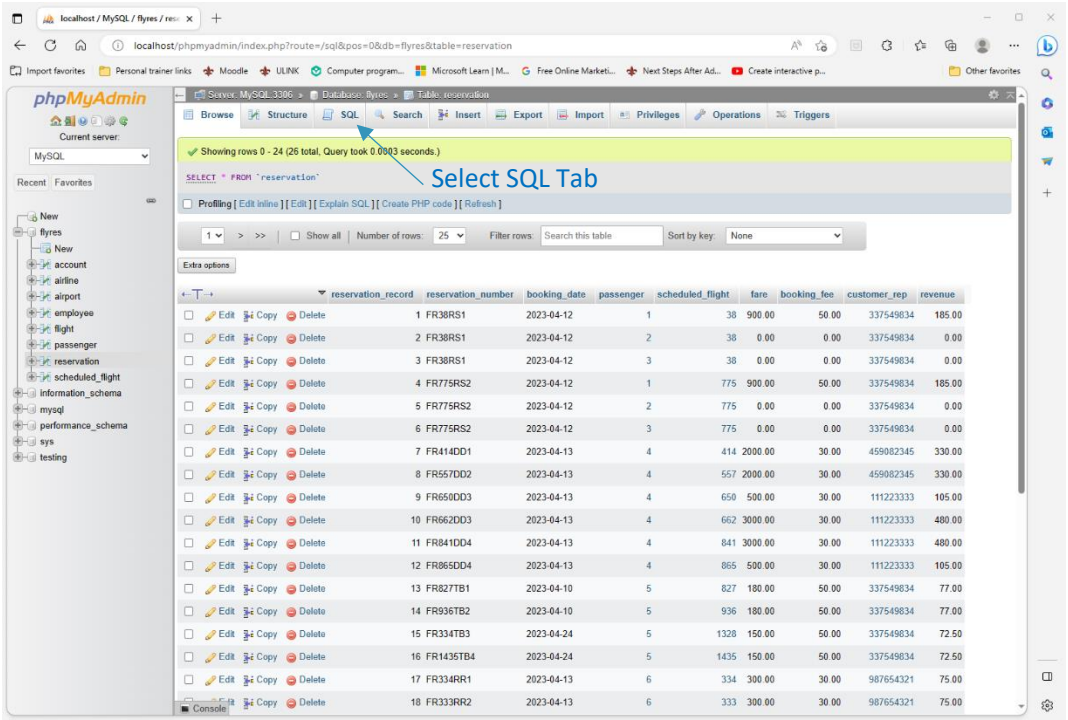
Calculating Revenue

In order to calculate revenue for new reservations, we need to update the revenue column in the reservation table. This is done by performing an UPDATE on the reservation table.

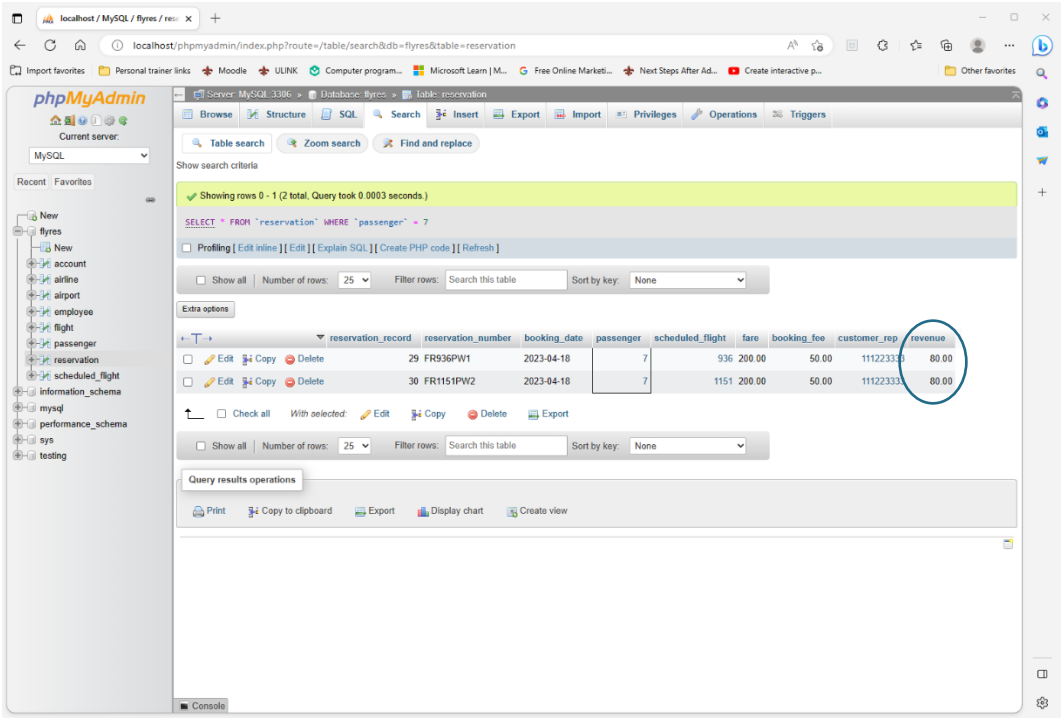
1. Open the reservation table by clicking reservation on the left side panel and then click the SQL tab across the top toolbar.
2. Copy and paste or type the following command:

   UPDATE reservation SET revenue = (fare*.15) + booking_fee;

3. Select ‘Go’ to execute the command.



4. If prompted, confirm that you would like to execute the UPDATE command. You can now go back to the reservation table and view the new reservations with the updated revenue generated by these transactions.

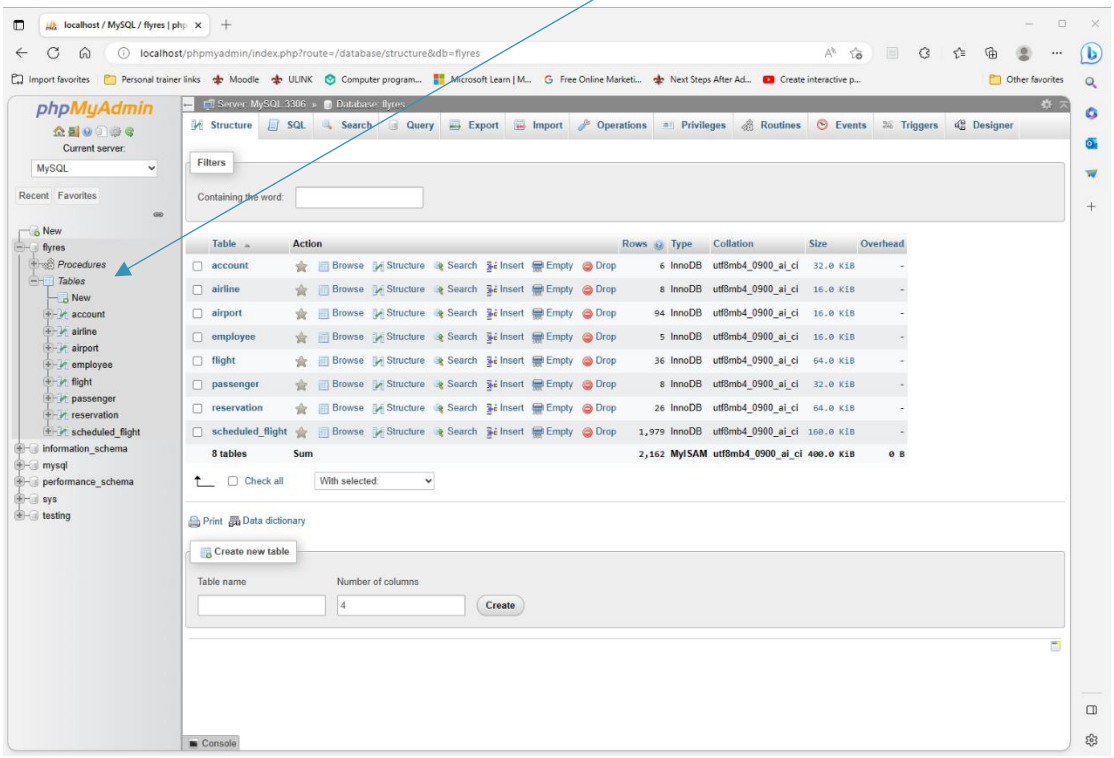


System Reports

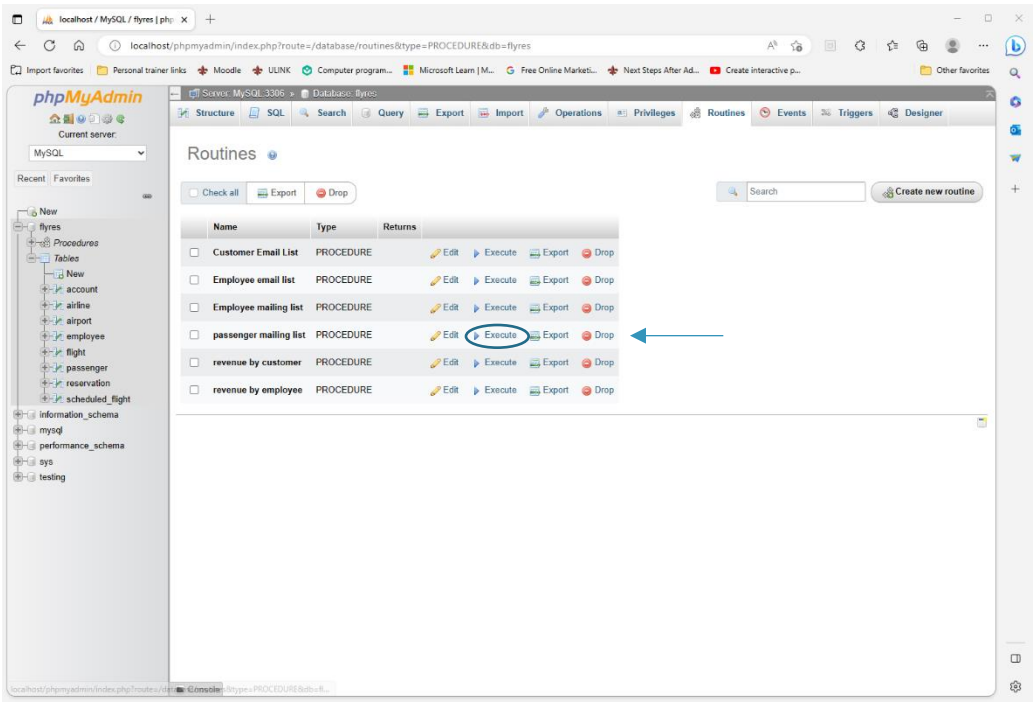
There are various reports that can be generated using the phpMyAdmin graphical user interface for FlyRes. This section will show how to create and view each of these reports. Some of these reports are generated using stored procedures. The list of stored procedures can be accessed from the left side panel of the FlyRes database. Simply click on Procedures listed under flyres to view this list. Then, select execute under the desired report. Some reports will require executing commands in the SQL command line. All instructions for reports are listed below.

Passenger Mailing List

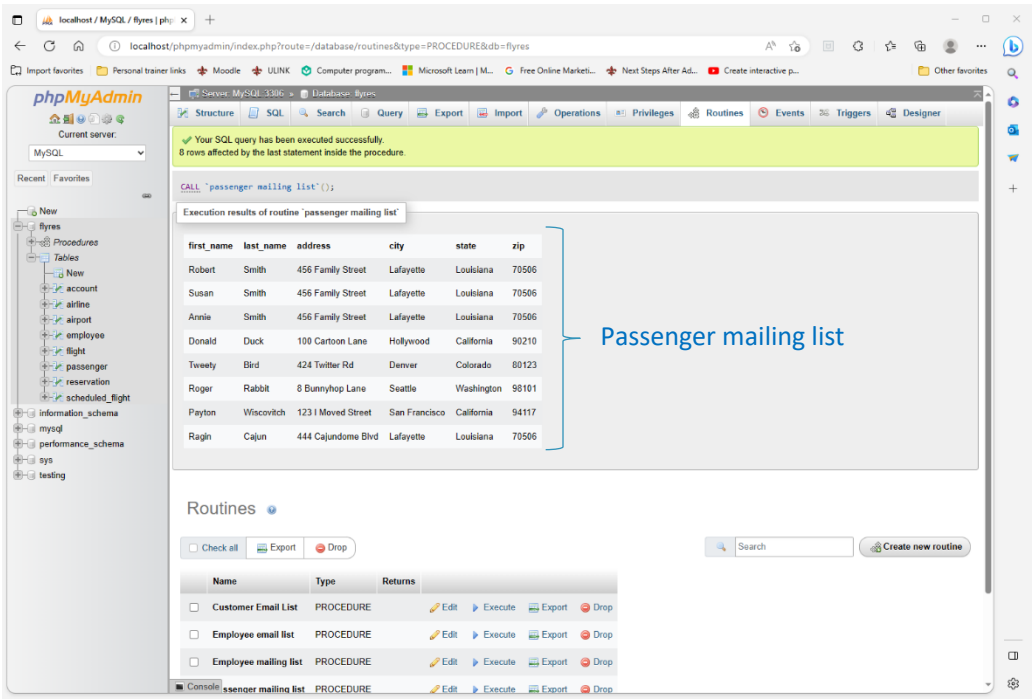
1. To generate a passenger mailing list, click on the Procedures link directly under flyres in the left side panel.



2. The stored procedures list will appear allowing you to select which report you would like to generate.
3. Locate the procedure named '**passenger mailing list**' and select **Execute**.

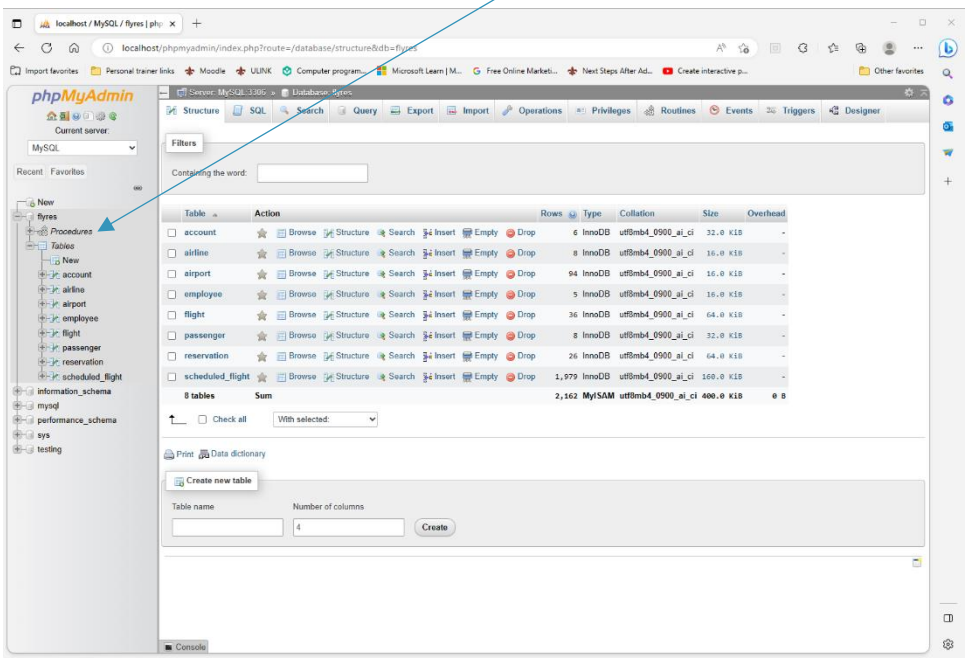


4. The report is generated.

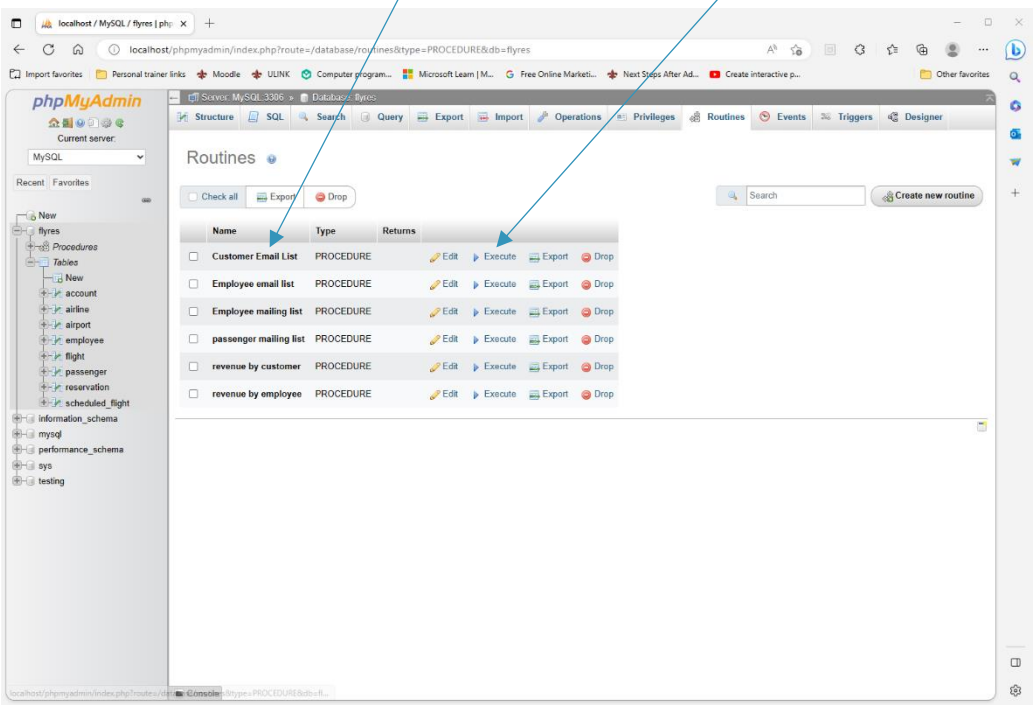


Customer Email List

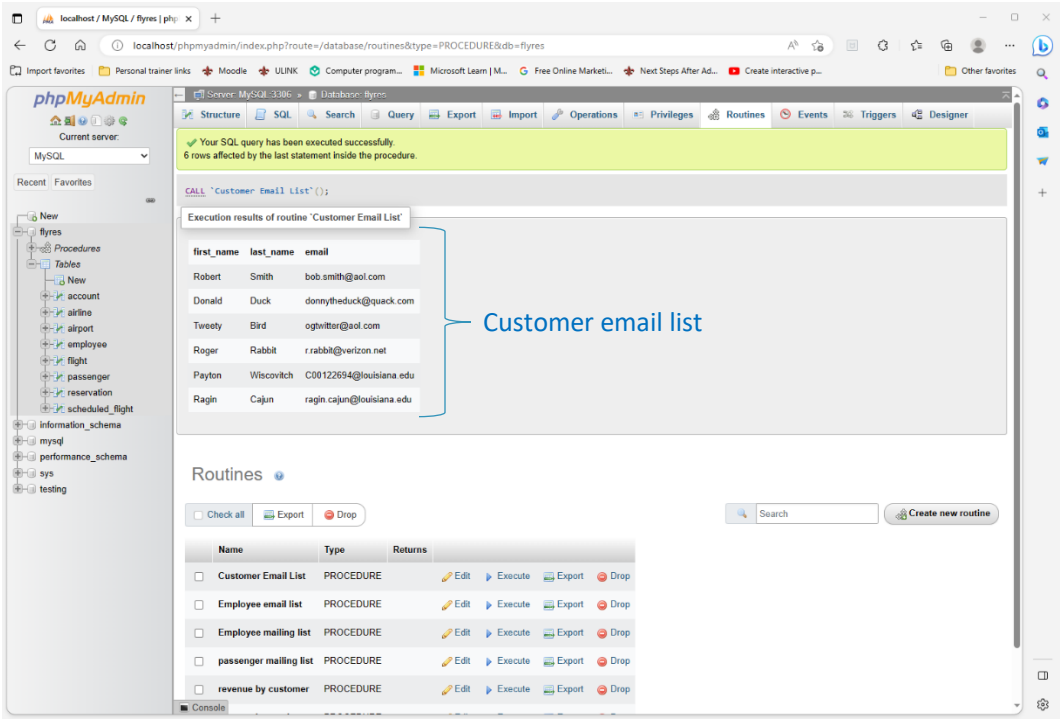
1. To generate a customer email list, click on Procedures under flyres from the left side panel.



2. The stored procedures list will appear allowing you to select which report you would like to generate.
3. Locate the procedure named '**Customer Email List**' and select **Execute.**

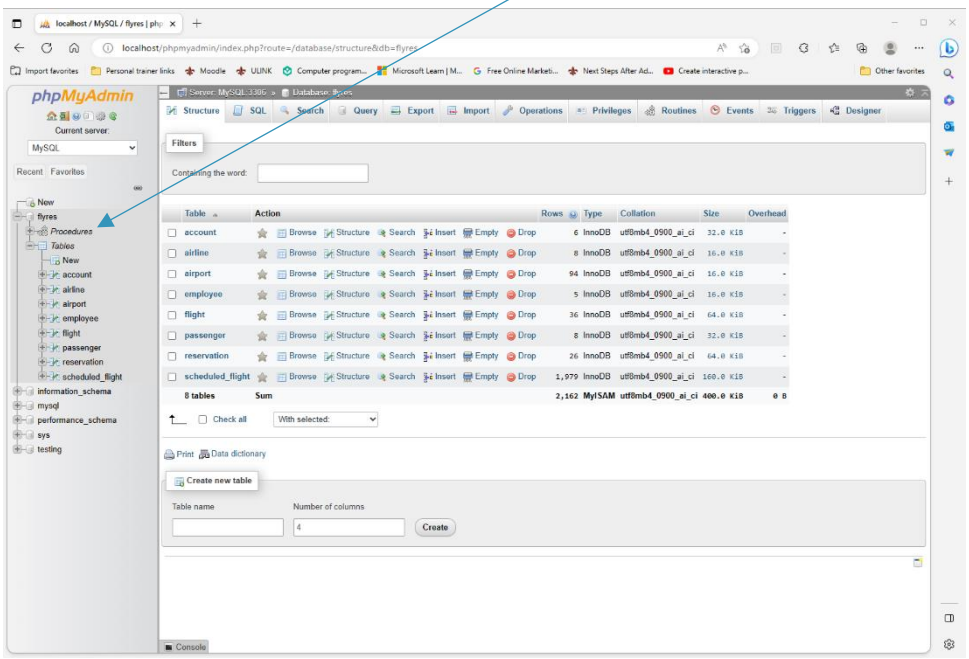


4. The customer email list is generated.

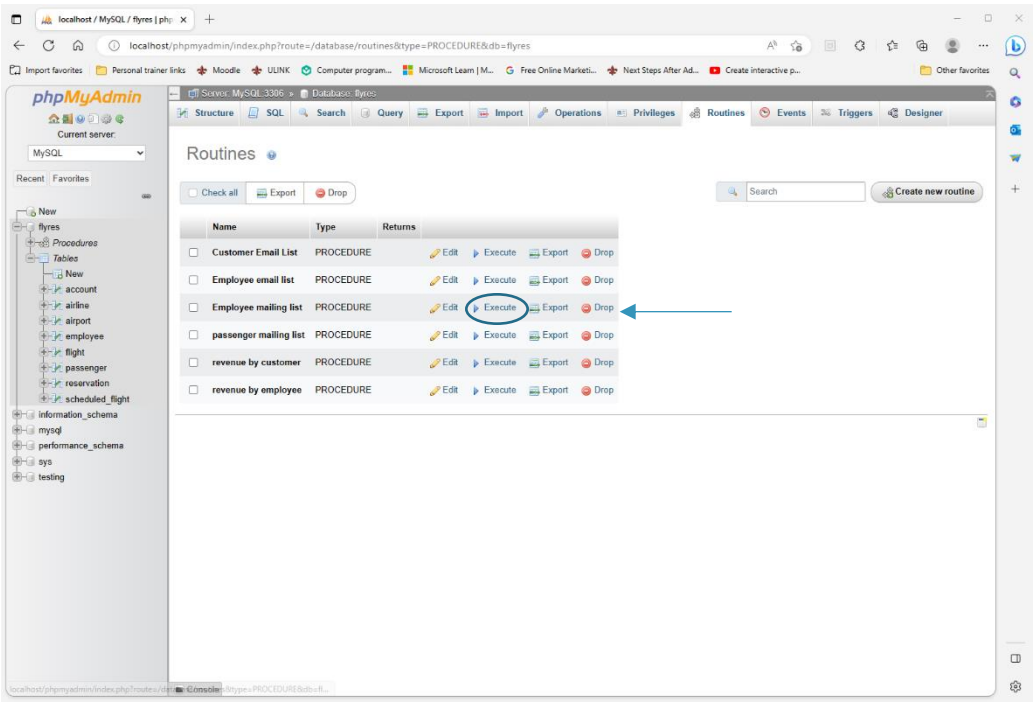


Employee Mailing List

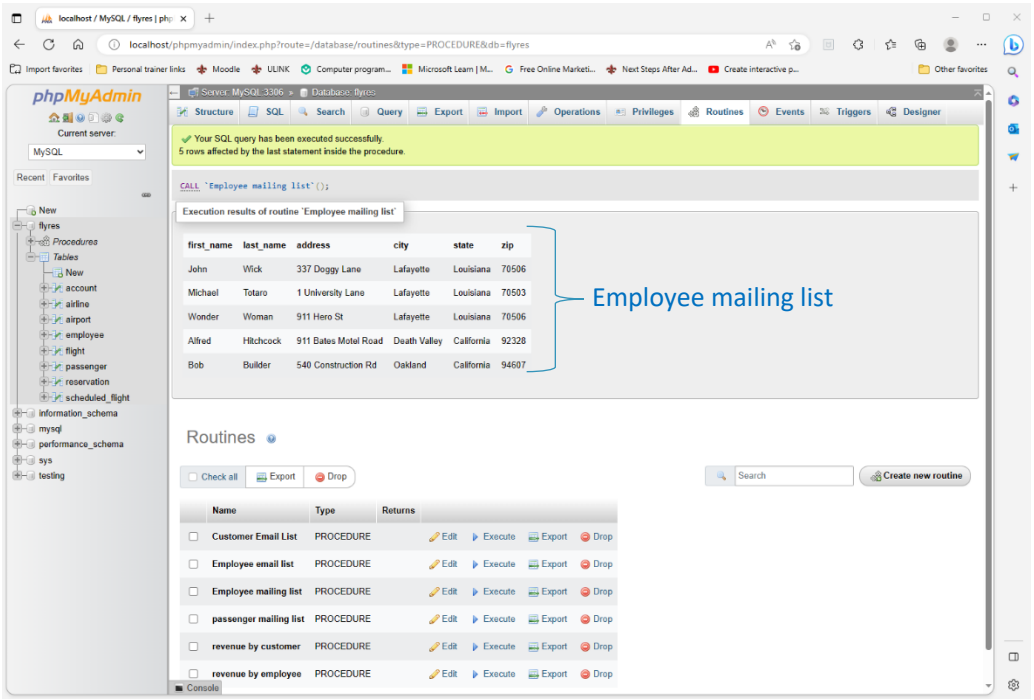
1. To generate an employee mailing list, click on Procedures under flyres from the left side panel.



2. The stored procedures list will appear allowing you to select which report you would like to generate.
3. Locate the procedure named '**Employee mailing list**' and select **Execute.**

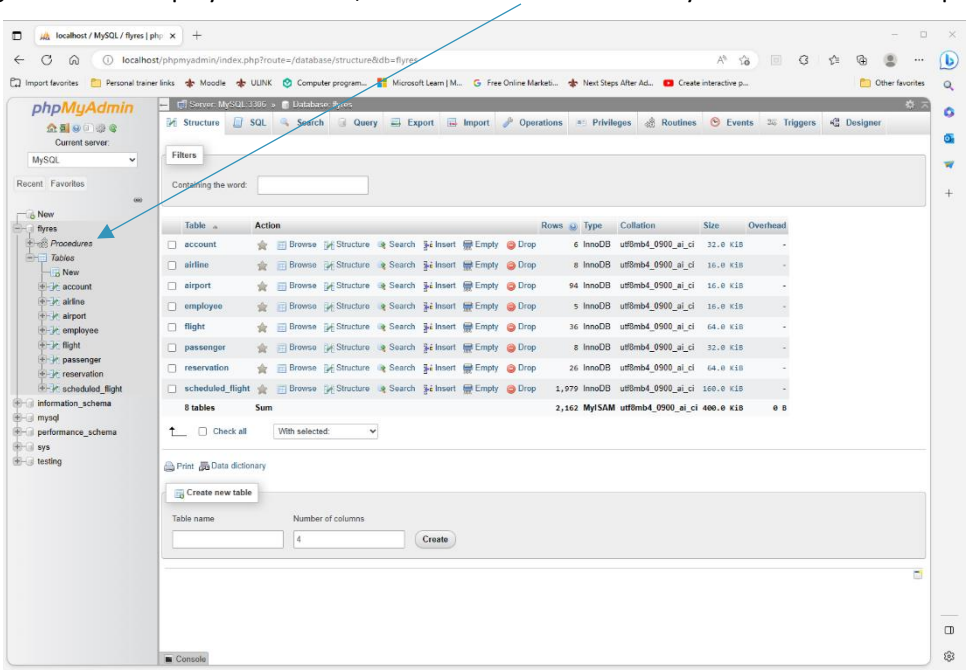


4. The employee mailing list is generated.

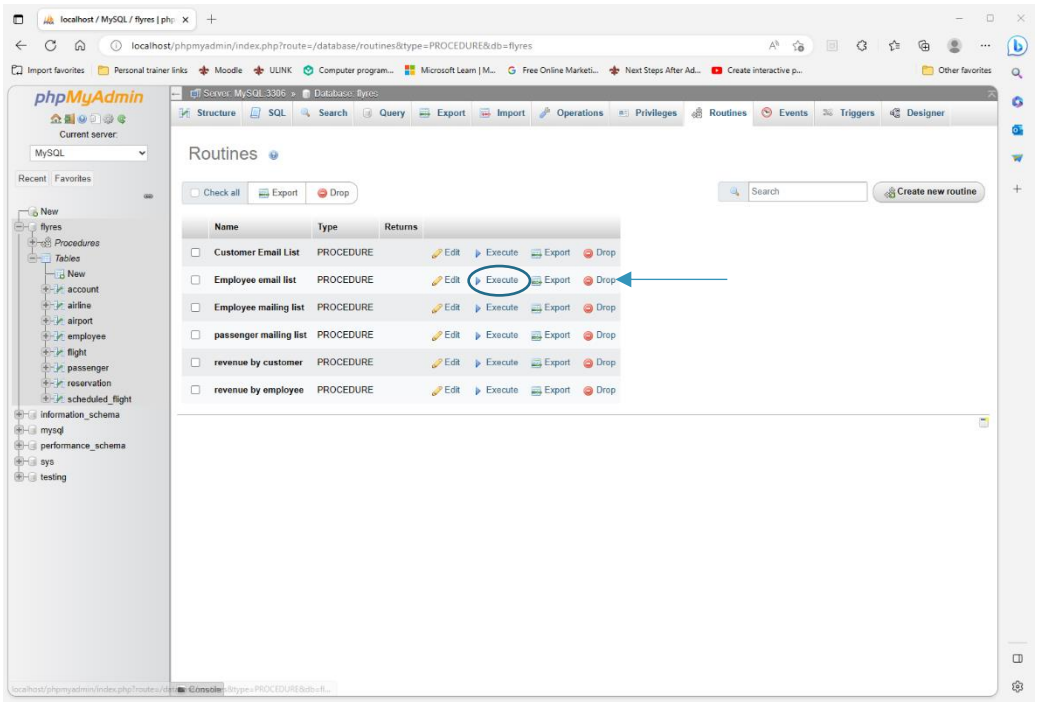


Employee Email List

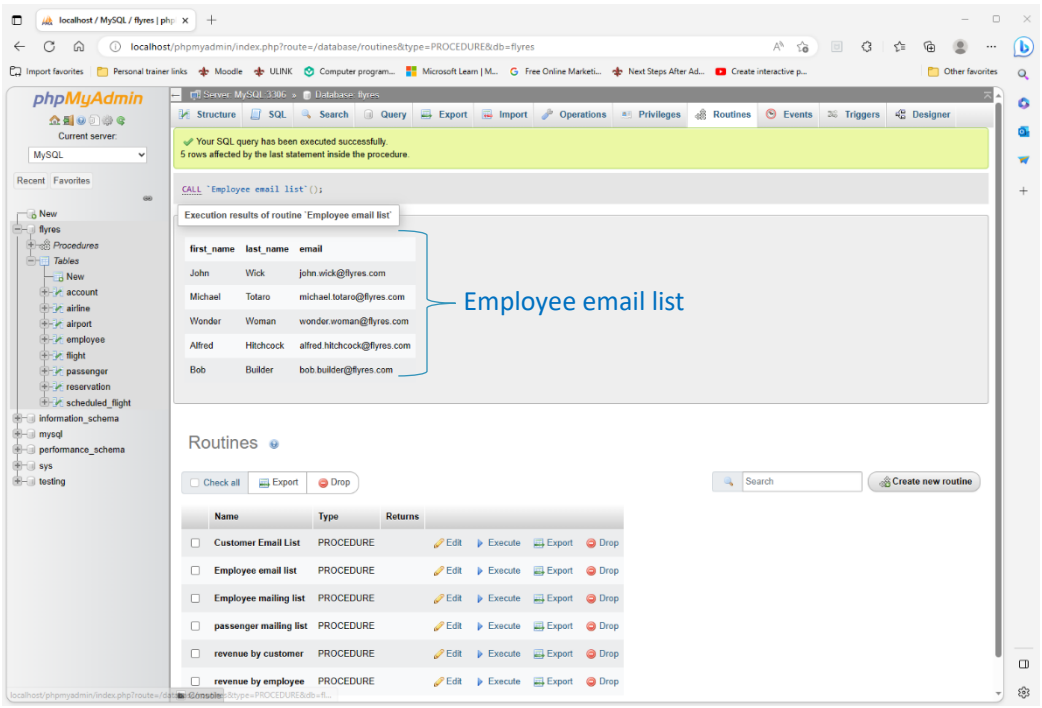
1. To generate an employee email list, click on Procedures under flyres from the left side panel.



2. The stored procedures list will appear allowing you to select which report you would like to generate.
3. Locate the procedure named '**Employee email list**' and select **Execute.**

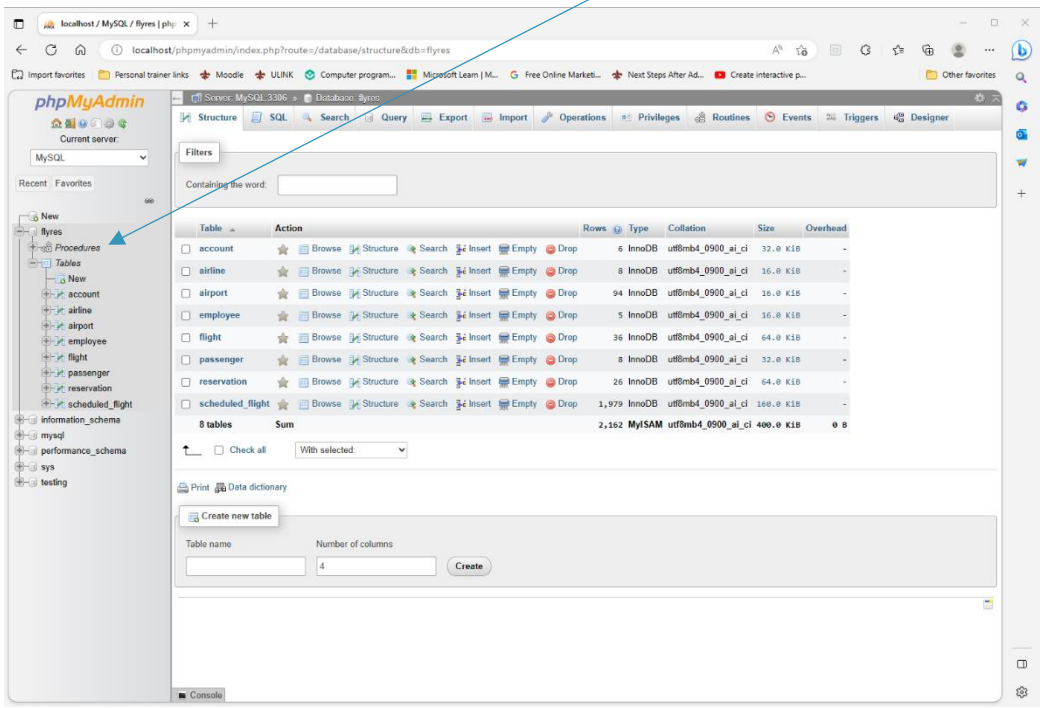


4. The employee email list is generated.



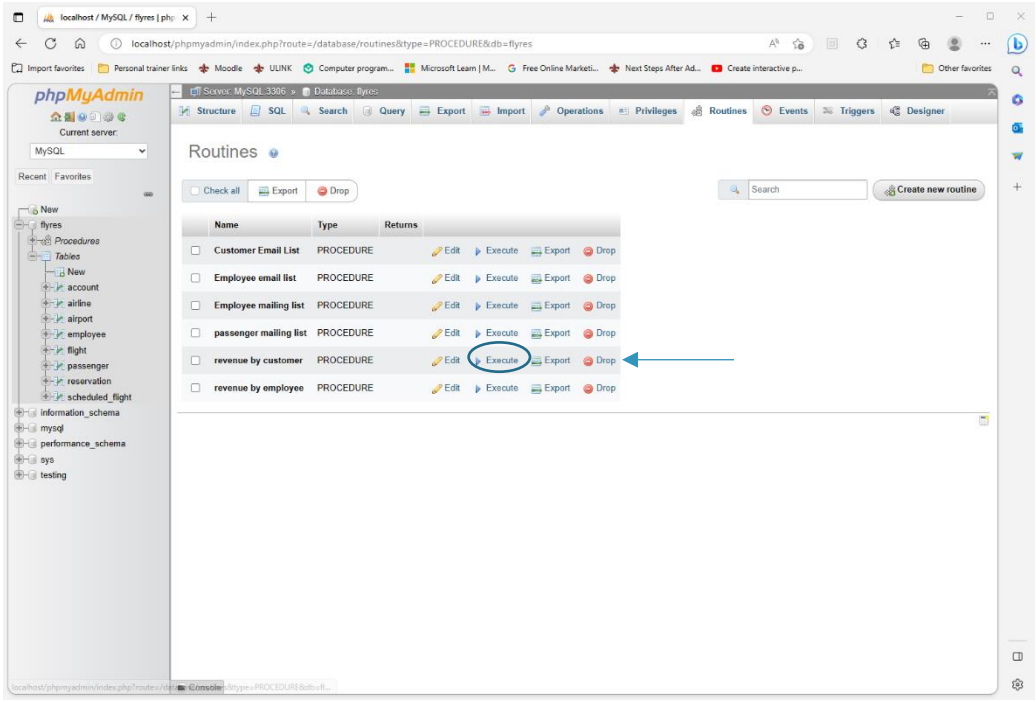
Revenue by Customer

1. To generate a revenue report by customer, click on Procedures under flyres on the left side panel.

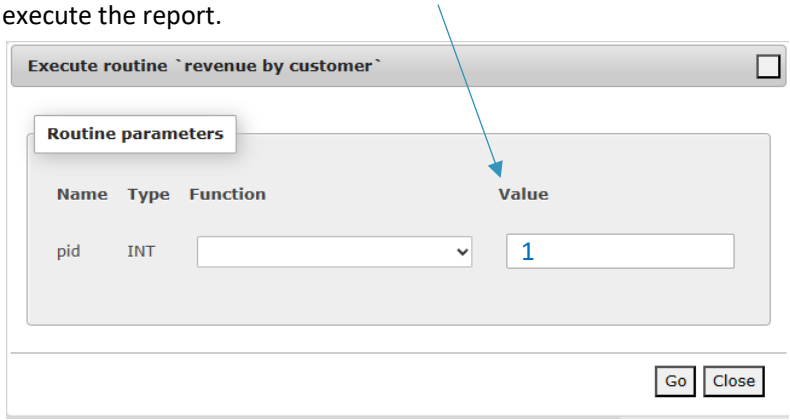


2. The stored procedures list will appear allowing you to select which report you would like to generate.

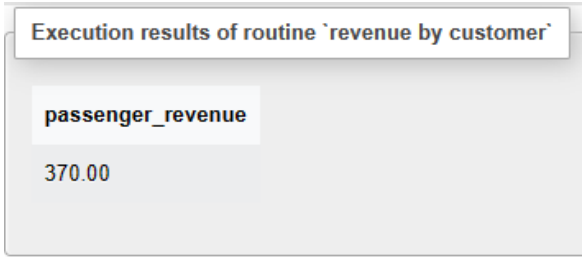
3. Locate the procedure named '**revenue by customer**' and select **Execute.**



4. You will be presented with a Routine parameters input and will need to provide the passenger_id for which you are generating the report. For our example, we will calculate the total revenue generated by customer/passenger 1, Robert Smith. Type the pid (passenger id) number into the value box. Then select 'Go' to execute the report.

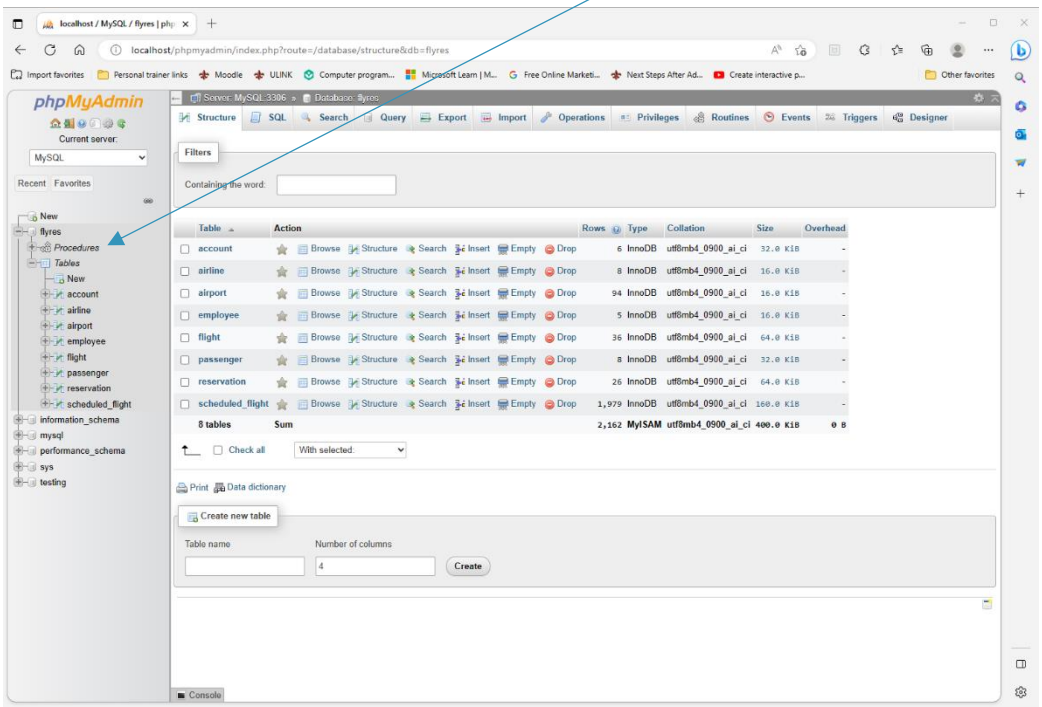


5. Total revenue generated for passenger 1 is calculated and returned.

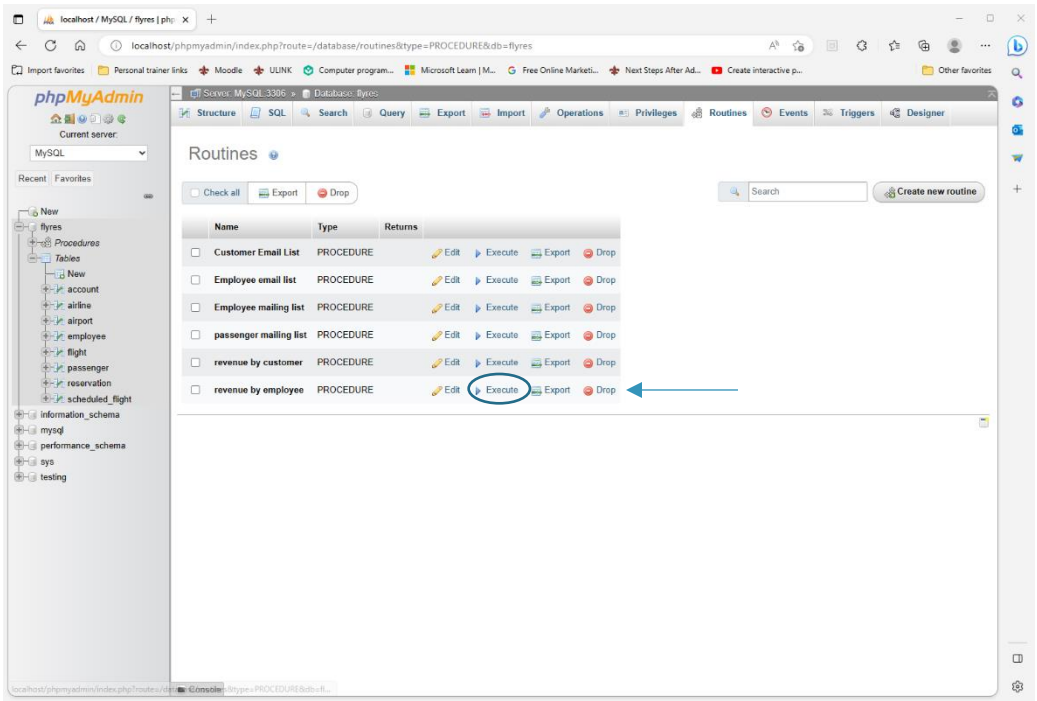


Revenue by Employee

1. To generate a revenue report by employee, click on Procedures under flyres on the left side panel.

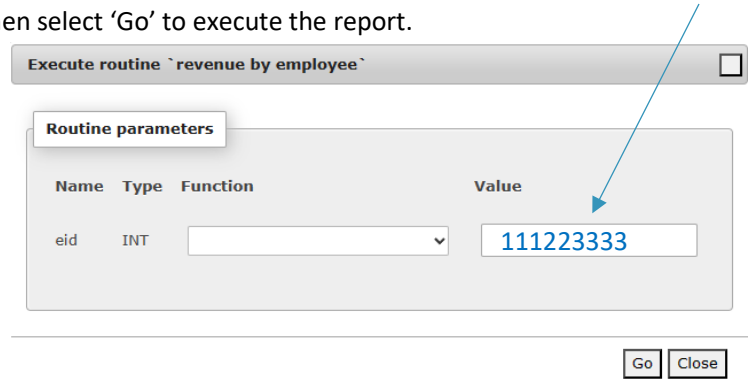


2. The stored procedures list will appear allowing you to select which report you would like to generate.
3. Locate the procedure named '**revenue by employee**' and select **Execute.**

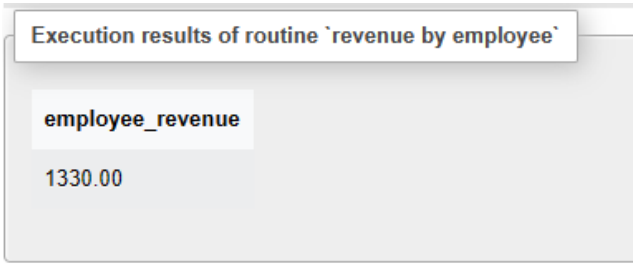


4. You will be presented with a Routine parameters input and will need to provide the employee_id for which you are generating the report. For our example, we will calculate the total revenue generated

by employee John Wick with employee_id 111223333. Type the eid (employee id) number into the value box. Then select 'Go' to execute the report.

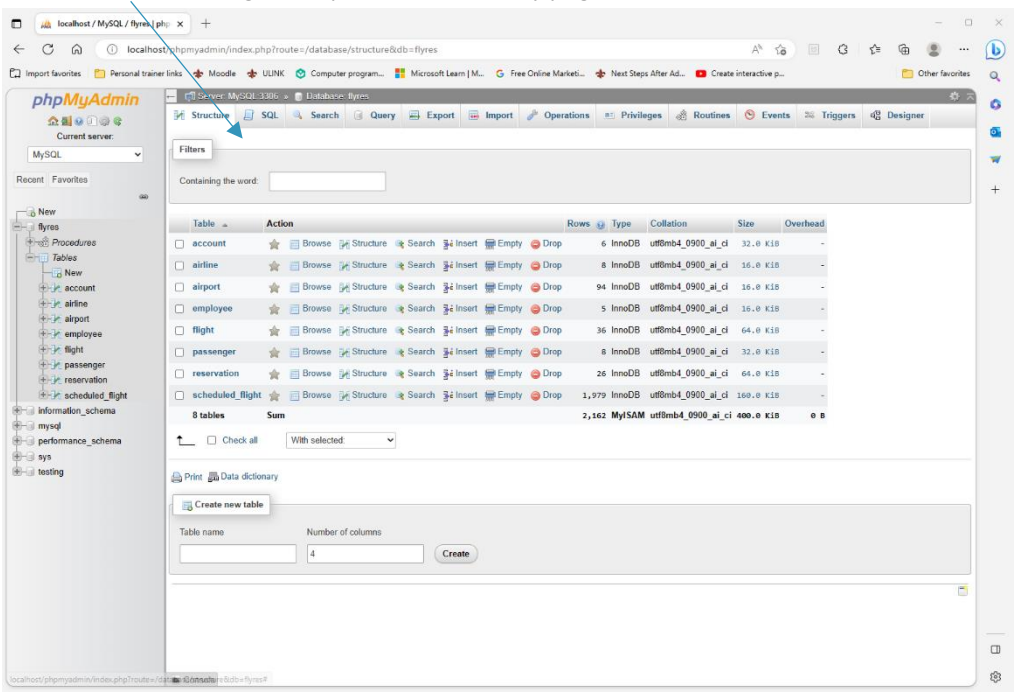


5. The total revenue generated by employee John Wick 111223333 is calculated.

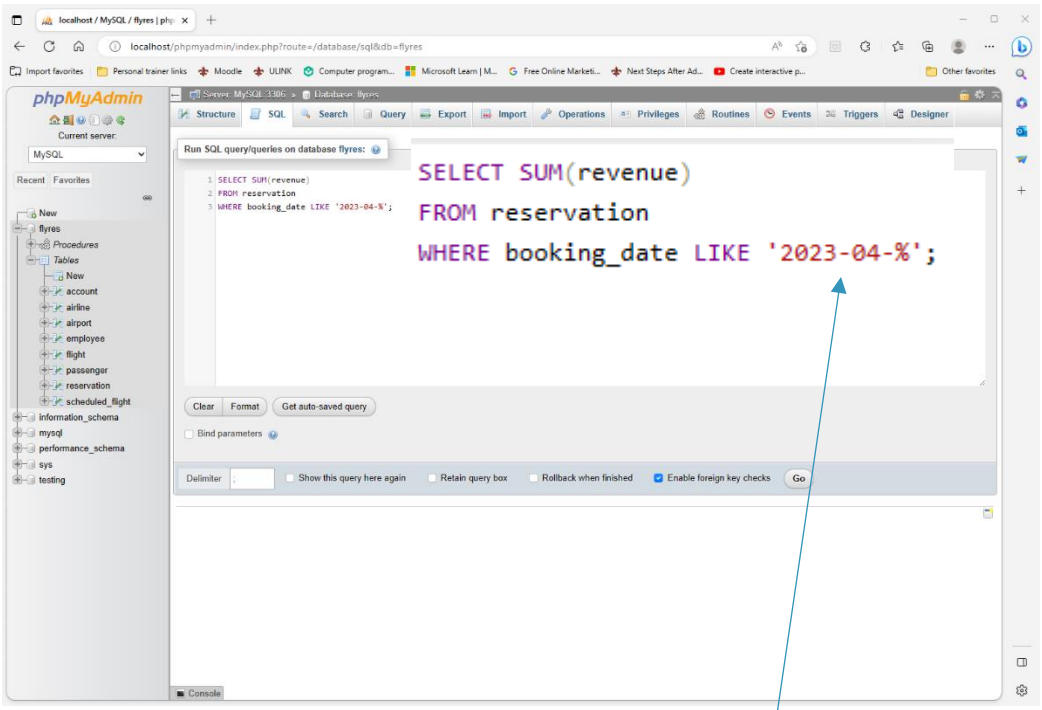


Revenue by Month

1. Select the SQL tab along the top toolbar from any page.



2. On the SQL query screen input the following:
   SELECT SUM(revenue) AS month_revenue
   FROM reservation
   WHERE booking_date LIKE [insert date parameter]
3. The date parameter needs to be entered in the following format:
   - 'YYYY-MM-%' example: '2023-04-%' will return the total revenue generated in April 2023.
   - The date must be enclosed in the quotation marks. The % symbol is used to represent all days in the month and year provided.



4. After inputting the command and appropriately providing the date parameter, click 'Go' to execute.
5. The total revenue for the month and year provided will be calculated.