Sample: GitHub Copilot Workshop Prompts

Step 1 - Create a new repo
Step 2 - Clone repo to machine OR Create a Codespace
Step 3 - Create a new branch
Step 4 - cd into workshop solution/server
Install copilot, copilot-chat, postman

Building the model

1. How do I create a notebook in vscode?
2. I want to load a dataset and load the first few lines.
3. I need to cleanse the loaded df above, how do I identify null values and replace those values?
4. How do I display null_values in numerical format?
5. Normalize data by removing outliers from depdelay and arrdelay?
6. View the first 50 rows in the clean data
7. Create a bar chart showing average departure delay by airline
8. Create a bar chart showing arrival delay that was > 15mins by airline
9. What is the distribution of the DepDeploy and ArrDelay cols
10. How do the carriers compare in terms of arrival delay and performance?
11. Which route has the highest average arrival delay?
12. Which route has the most late arrivals from origin airport to destination airport?
13. Which departure airport has the highest average delay?
14. What are the arrival delays for different days of the week?
15. Find the relationship between late departure and arrival days
16. Create a model to predict the likelihood of a flight being delayed based on the day of the week and the arrival airport, use a logistic regression model, split the date into training and testing sets, train the model, calculate the accuracy of the model
17. Calculate the accuracy of the model
18. Show the confusion matrix of the model

19. Example: show the odds of a flight being delayed to Chicago on a Monday
20. Example: Show the odds of a flight being delayed to Las Vegas on a Friday
21. Example: Make a prediction of the odds of a flight being delayed to Los Angeles on a Wednesday
22. Saving airports: Get unique column values for origin airport and id and export to CSV
23. Export model: export the model to import later into Flask

Making requests in Postman:

- Generate github token (echo $GITHUB_TOKEN)
- Copy token in terminal
- Add to Headers "value" in Postman with the key "X-Github-Token"
- Copy forwarded port and add url to postman to make request
    - Ensure you're on the correct route
- May need to install additional packages with pip if packages are nit found. Ask Copilot to help debug

Building the API

1. Create a basic server using flask
2. Inline Chat (CMD+I on MAC or CTRL+i on windows)
    a. Load the model from the pickle file
3. Create a route that returns a prediction of flight delays using the model
    a. The model accepts two params day of the week and airport id
    b. Then I add:
        i. "print(prediction[0])
        ii. Be sure to use the model.predict_probability
        iii. Make a request using postman
            1. Route: /predict?day_of_week=5&airport_id=10529
            2. (will get error, debug with copilot) >eg. Object of type ndarray is not JSON serializable

iv. Update the data type . . . to int for req
v. Convert predictions to list
vi. Make a postman request and should have the ML predictions now

vii. split the prediction into two variables
viii. the first value is the probability of the flight not being delayed
ix. the second value is the probability of the flight being delayed

x. Convert the prediction to a percentage and make 0 decimal places
xi. Return json
    1. Add interpretation to the json output

xii. Highlight route and ask chat "how can I improve the highlighted route?

4. Create a route that returns all the airports from the file
    a. Create a GET route that returns the list of airports
    b. the list of airports is stored in a file called origion_airports.csv
5. Do a Postman Get Request

6. Highlight route and Ask chat: is this formatted correctly?
    a. Accept suggestion and see if it runs
    b. Check code for accuracy

7. Remove the first line of the airports (may not need is using pandas)
8. Split the airports into a list of dictionaries
9. Sort alphabetically and go to route
10. Do a Postman GET request

11. Add error handling to the route - if not already there

12. Convert airport ids to ints
13. Go to route
14. Run Postman GET Request

15. How do I add error handling to the highlighted route?
16. Add error handling
17. Run Postman GET Request

18. Enable CORS; use flask_cors

BUILDING THE UI

1. How do I create a new react app with vite?
    a. copy command and update name to `client`
    b. cd in client
    c. npm i
    d. npm run dev

2. Prompt:
    a. create a basic UI with a drop down menu with days of the week and a second dropdown menu with airports pulled from an api and include a submit button. App is using vite

3. Can you center the content on the page in the body tag? (in the css)

4. I want to fetch the airport listing. The airport api endpoint is http://127.0.0.1:5000/airports and returns json. Update the code to reflect this

5. I'm getting a CORS error in the console. The ApI is built with Flask and uses the CORS package. How do I fix this error? Do I use the flask decorator?

6. create a function to fetch data from the route /predict that accepts 2 params - the day of the week as a number, and the airport ID. We

need to send the users selection from the dropdowns to the server, then receive the prediction response

7. update the handlesubmit function to send the users selection - the day of the week and the airportID to the route /predict to get and return the prediction of whether a flight will be delayed or not

8. ok before I move on, I need the days of the week to be returned as a number but present to the user as days, how can I do that?

9. remove the decimal from the percent (in the API)

10. update the styling of the app to use Material UI, make it look pretty and nice. Write custom CSS and add a custom theme if needed. Use material UI 5.

11. how to I add the Paper component so the container content is elevated

12. createMuiTheme is deprecated (if applicable)

13. How do I Implement client-side caching?