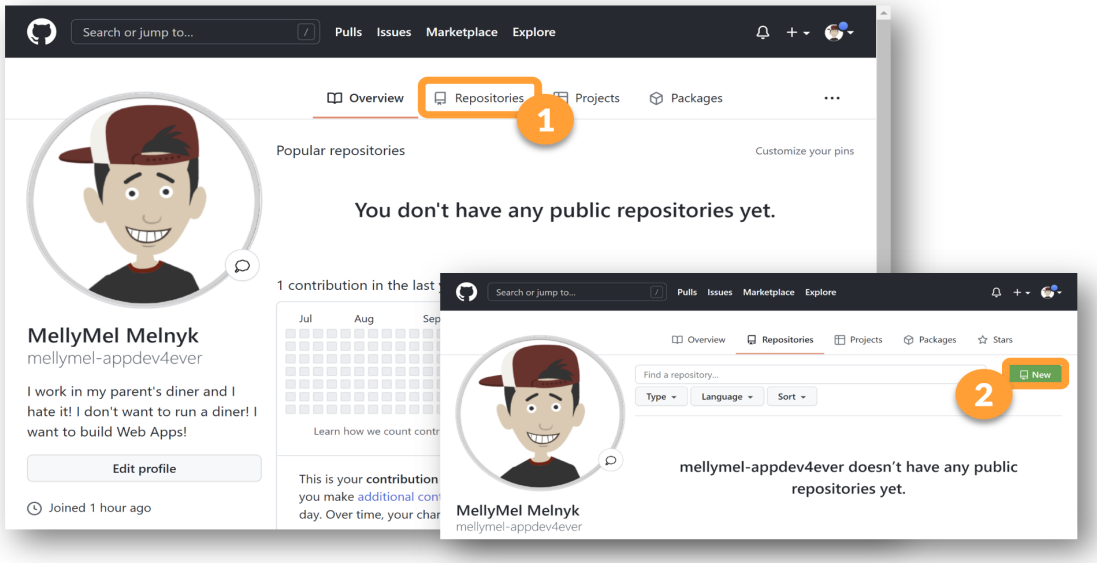
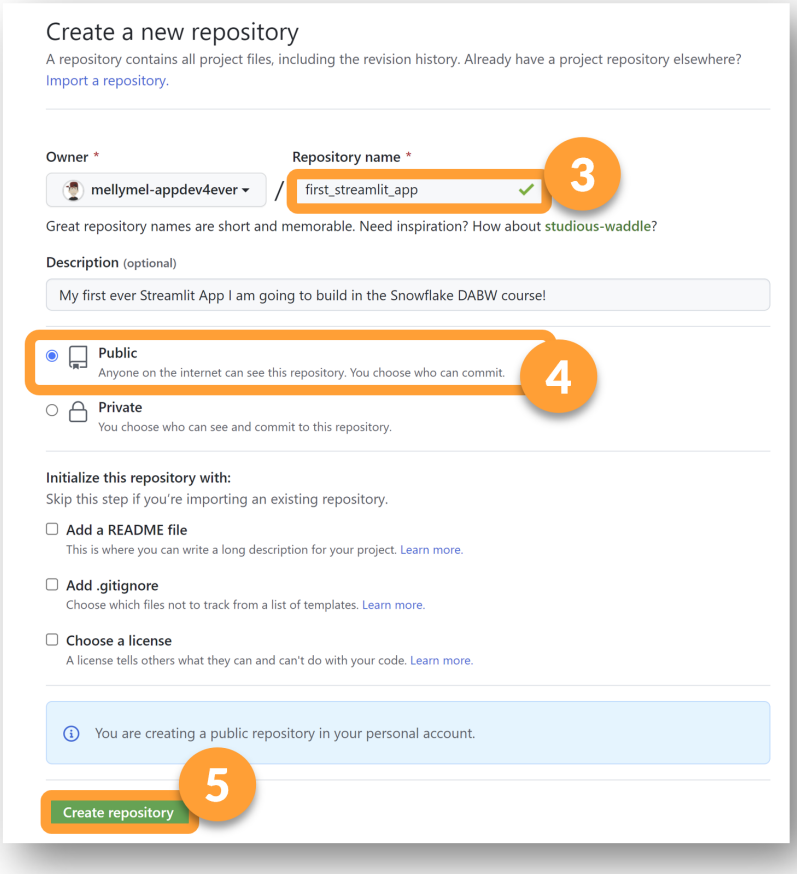
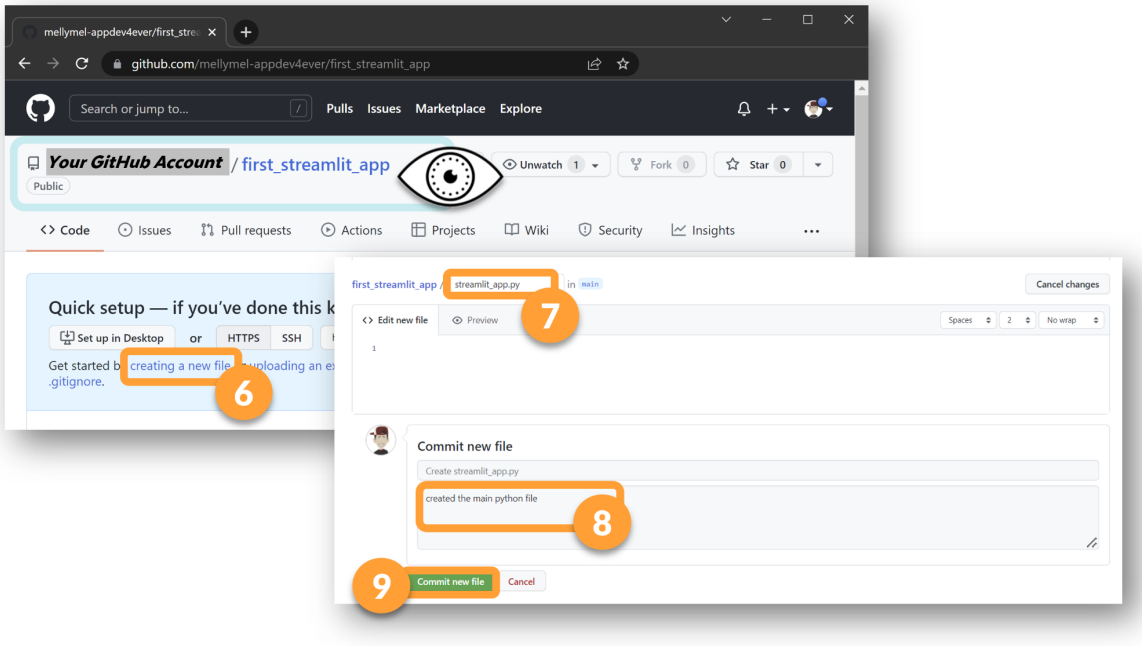
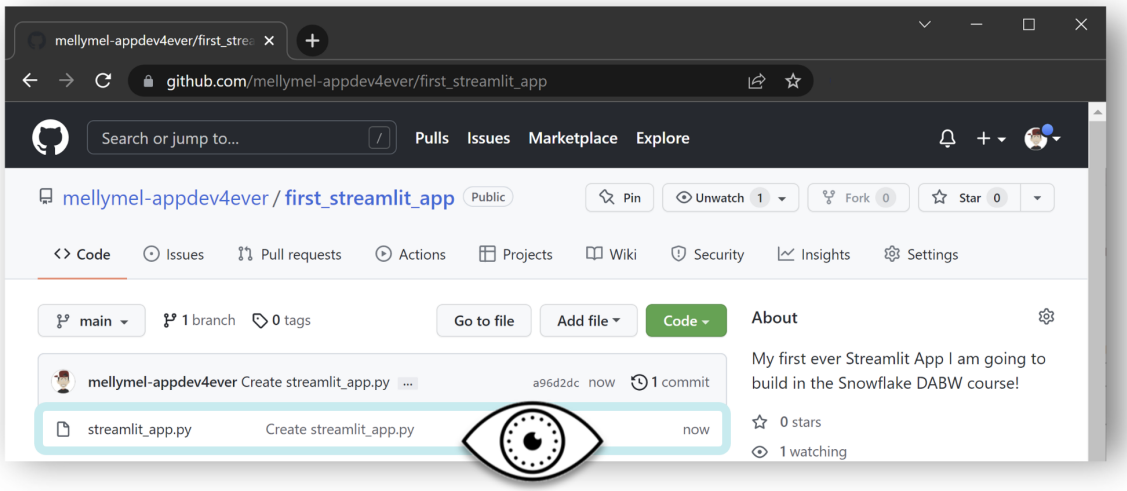
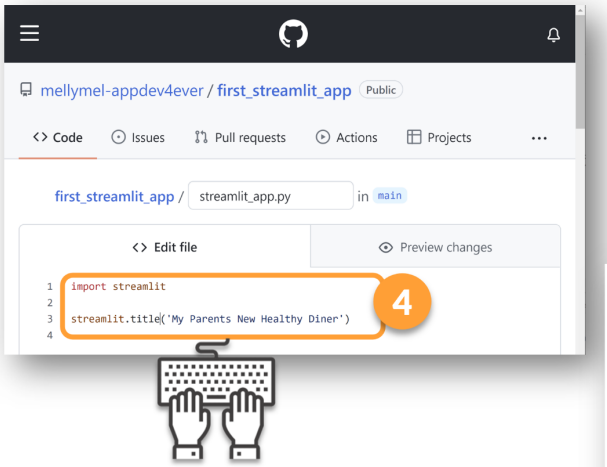
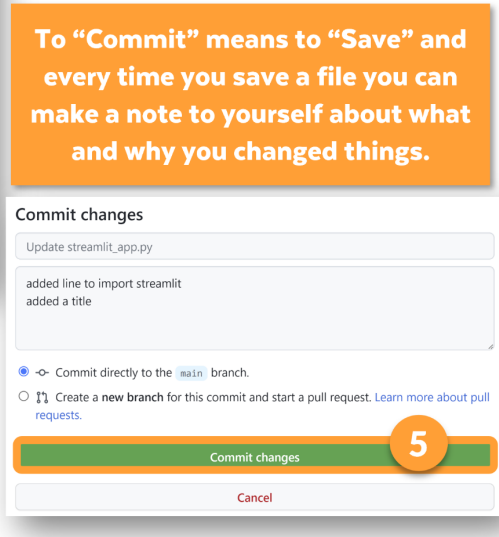
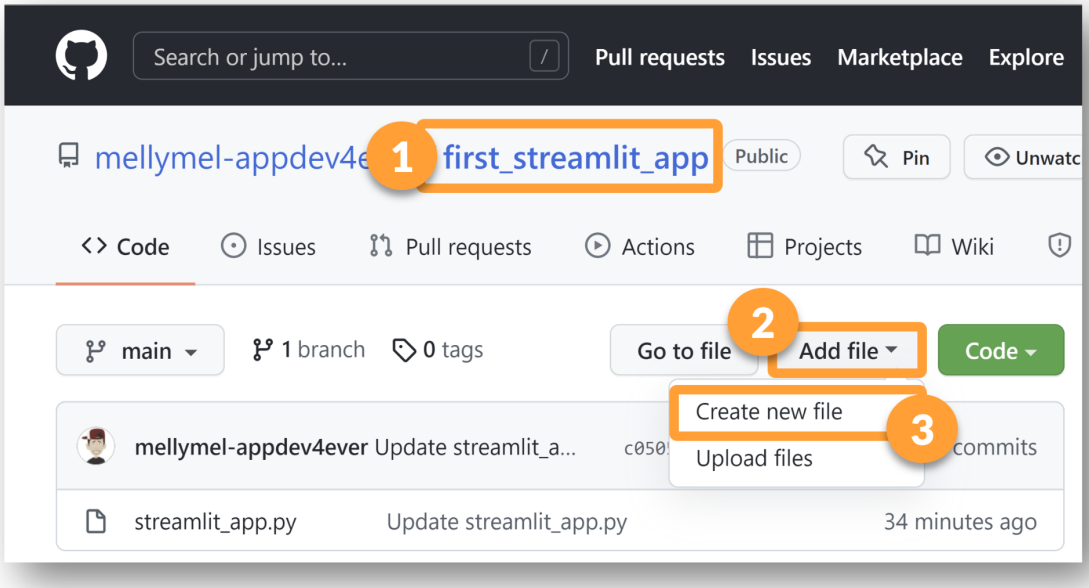
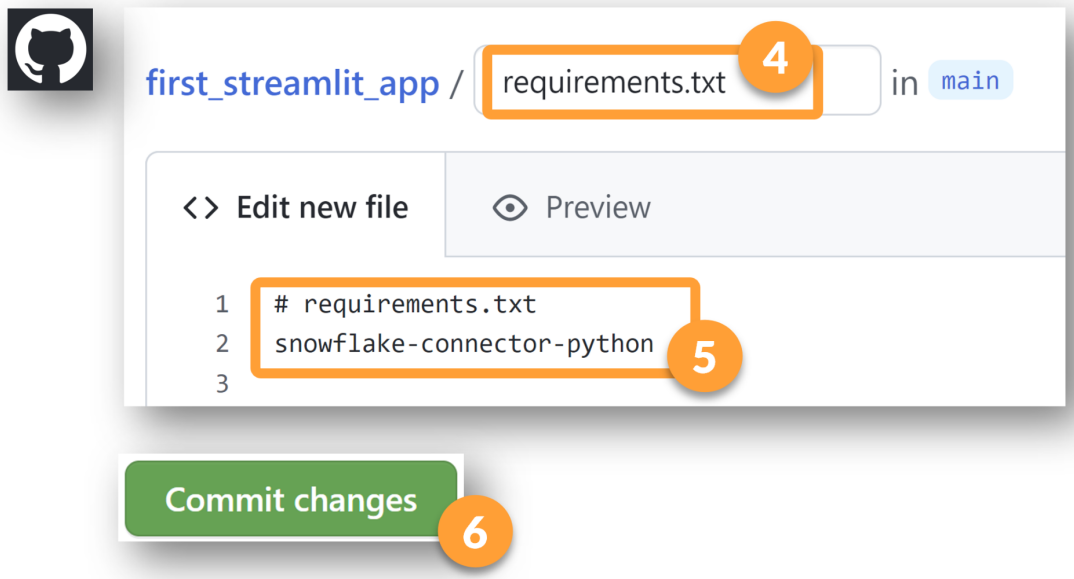
Zena's Streamlit App Cheat Sheet

This document is for learners who already did the DABW badge but can't remember all the steps for setting up a Streamlit app. Setting up your own version of Zena's App is **OPTIONAL.** Use this cheat sheet to get a new app up and running, then copy the final code from the last step of this document. This creates a copy of Zena's Athleisure Web Prototype for anyone who likes AppDev and wants a little more practice with Streamlit.

Sign in to your Github account. Click Repositories, New.	
Name the repo something that makes sense for the APP name. IT MUST BE PUBLIC. Maybe **zenas_web_Catalog**?	

Create a file that ends in .py All apps should probably have the same name for this file.	
Check to make sure it exists. Notice that the branch name is MAIN.	
Create App in Streamlit.	• Sign in to your existing Streamlit account. • Click the New App button • Select your new Repo from the drop list in the top field. • Leave default in place for the other two fields.

When app is built, start editing in one tab (Github) and viewing in another (Streamlit).	 
Click the little pencil icon on the gray bar otherwise it will just look like a gray bar.	
Add other python libraries (like Snowflake) by adding a requirements file.	

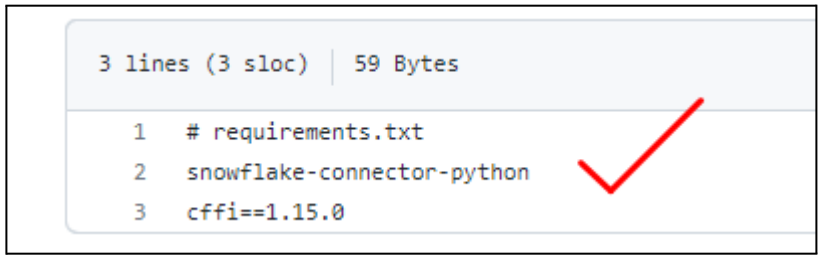


If you see errors like this one saying **VERSION MISMATCH**

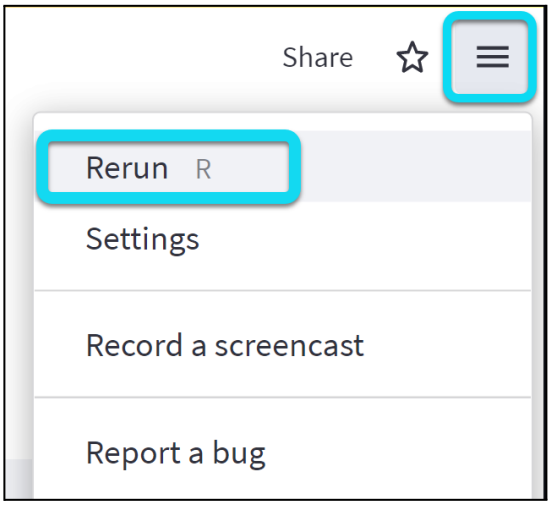
```
File "/home/appuser/.conda/lib/python3.7/site-packages/cffi/api.py", line 54,
    raise Exception("Version mismatch: this is the 'cffi' package version %s,
```

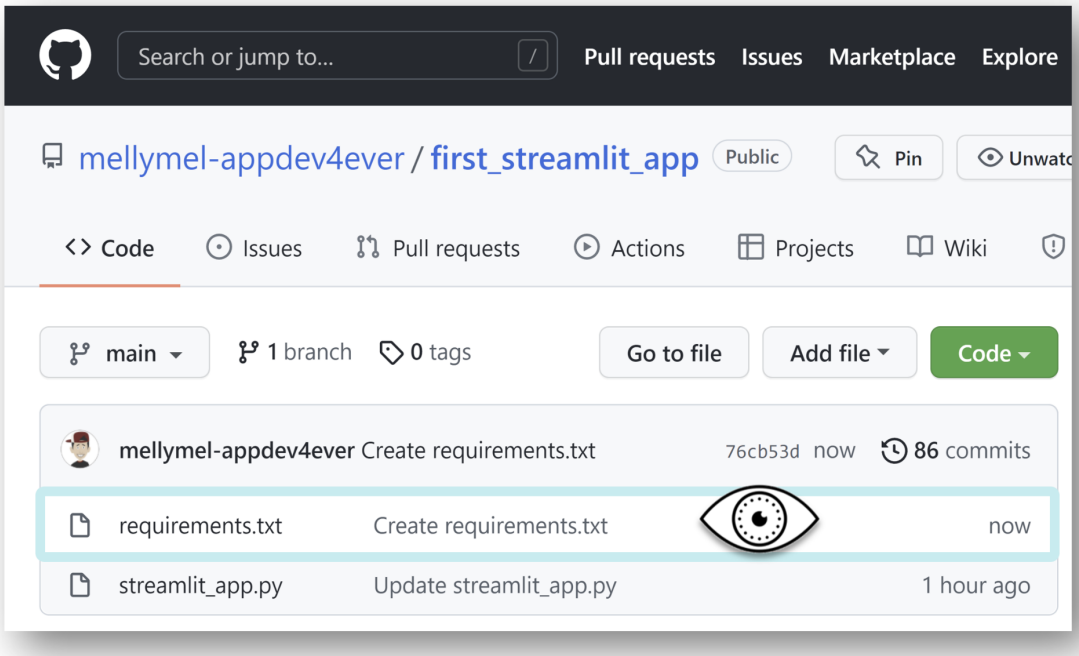
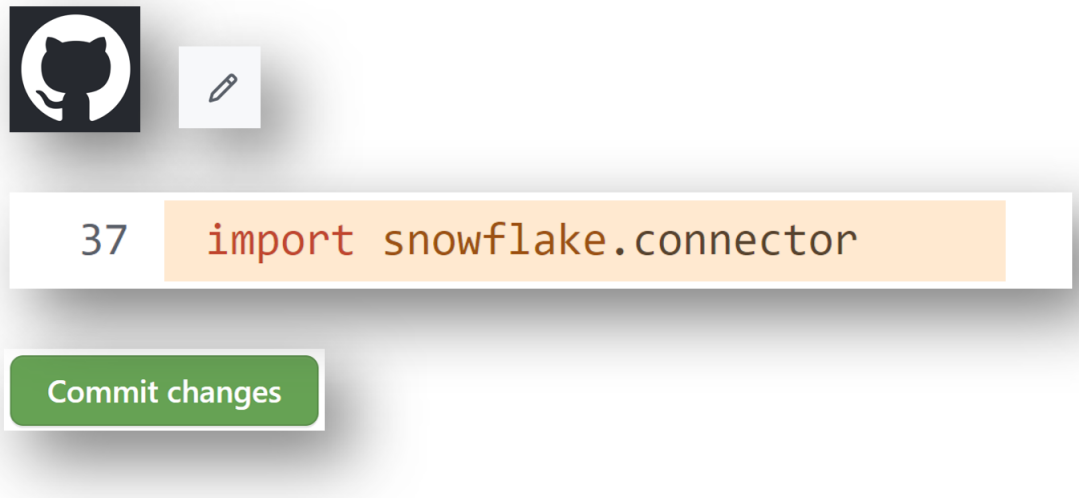
You will need to add lines to the requirements.txt file telling Streamlit exactly what versions of which packages you want it to use.

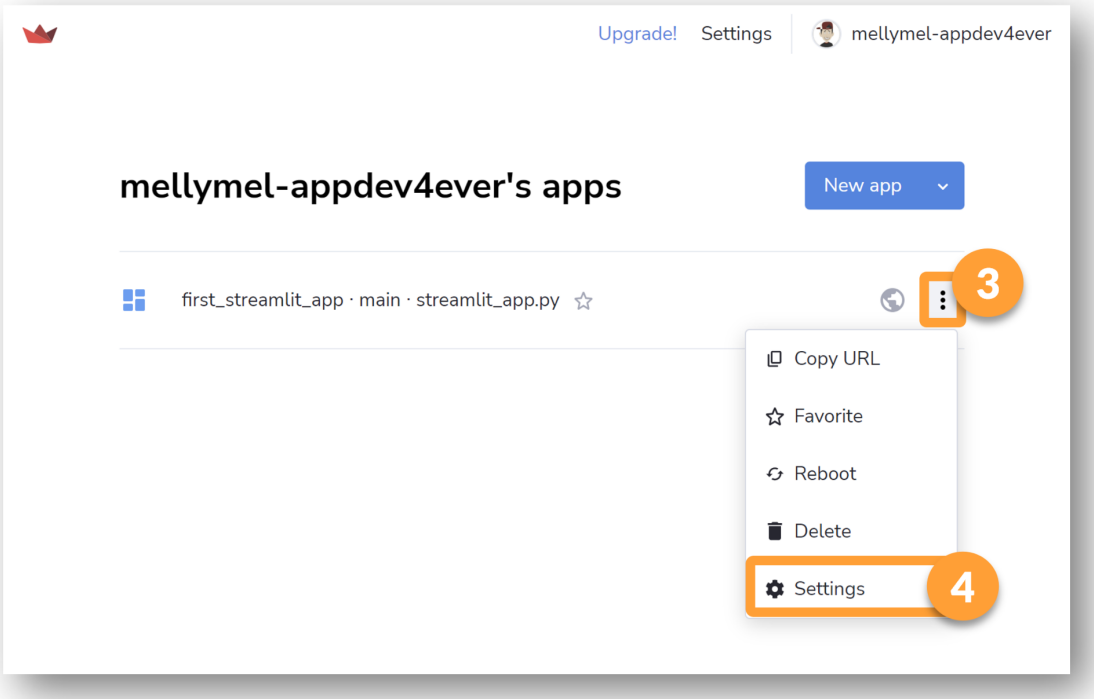
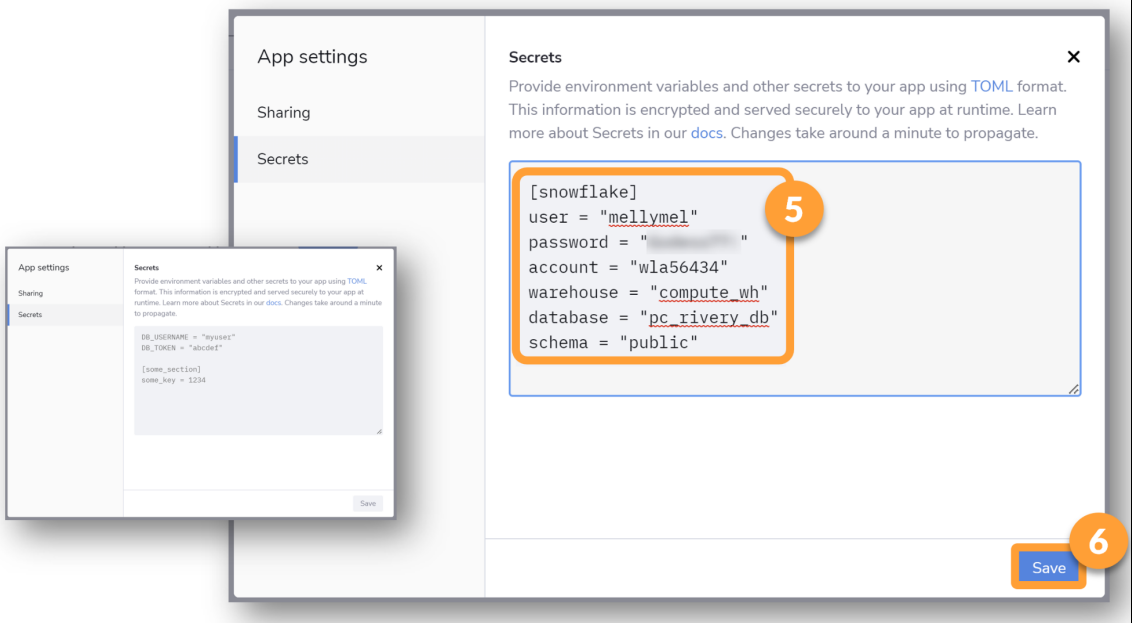
In this case it says there is a version issue with something called “the cffi package” so we add this to our requirements file and save it.



After you make a change like this, you have to restart the app because Streamlit needs to load a different package library. Use the menu in the upper right corner to RUN the app fresh.



You should now have two files in your project.	
Add an import statement to the py file to make sure it's not going to throw an error. If it does, you did something wrong in the last step.	

Add to the secrets file by returning to main menu using “Settings” If you don’t have the Settings option simply open a browser tab to: https://share.streamlit.io/	
	

Your secrets file will need to have YOUR TRIAL ACCOUNT info	`[snowflake]`
	`user = "xxx"`
	`password = "xxx"`
	`account = "acctlocator.region.cloud"`
	`warehouse = "xxx"`
	`database = "xxx"`
	`schema = "xxx"`

EXAMPLE SECRETS FILE VALUES **Remove region.cloud**	[snowflake] user = "zena" --put your username here password = "redacted!" account = "zoa49575.ca-central-2.aws"

if you are on us-west-2.aws **This region and cloud is the OG so it doesn't need to be stated**	warehouse = "compute_wh" database = "ZENAS_ATHLEISURE_DB" schema = "PRODUCTS"

Copy this code into the main py file and run to make sure requirements, secrets, and connection are all working together.
 	
Edit the secrets file until you get a successful connection.
	`my_cnx = snowflake.connector.connect(**streamlit.secrets["snowflake"])`
	
	`my_cur = my_cnx.cursor()`
	
	`my_cur.execute("SELECT CURRENT_USER(), CURRENT_ACCOUNT(),`
	`CURRENT_REGION()")`
	
	`my_data_row = my_cur.fetchone()`
	
	`streamlit.text("Hello from Snowflake:")`
	
	`streamlit.text(my_data_row)`

FINAL Code we used for Zena's Web Catalog Prototype

```
import streamlit
import snowflake.connector
import pandas

streamlit.title('Zena\'s Amazing Athleisure Catalog')

# connect to snowflake
my_cnx = snowflake.connector.connect(**streamlit.secrets["snowflake"])
my_cur = my_cnx.cursor()

# run a snowflake query and put it all in a var called my_catalog
my_cur.execute("select color_or_style from catalog_for_website")
my_catalog = my_cur.fetchall()

# put the dafta into a dataframe
df = pandas.DataFrame(my_catalog)

# temp write the dataframe to the page so I Can see what I am working with
# streamlit.write(df)

# put the first column into a list
color_list = df[0].values.tolist()
# print(color_list)

# Let's put a pick list here so they can pick the color
option = streamlit.selectbox('Pick a sweatsuit color or style:', list(color_list))

# We'll build the image caption now, since we can
product_caption = 'Our warm, comfortable, ' + option + ' sweatsuit!'

# use the option selected to go back and get all the info from the database
```

```
my_cur.execute("select direct_url, price, size_list, upsell_product_desc from catalog_for_website where
color_or_style = '" + option + "';")
df2 = my_cur.fetchone()

streamlit.image(
            df2[0],
            width=400,
            caption= product_caption
        )

streamlit.write('Price: ', df2[1])
streamlit.write('Sizes Available: ',df2[2])
streamlit.write(df2[3])
```

THIS IS AN OPTIONAL EXERCISE SO WE WILL NOT SUPPORT YOUR TROUBLESHOOTING. YOU MAY POST ON THE MESSAGE BOARDS ABOUT ISSUES BUT SNOWFLAKE STAFF WILL NOT BE ANSWERING YOUR INQUIRIES. YOU WILL NEED TO HOPE FOR THE BEST FROM YOUR PEERS OR SEARCH STREAMLIT DOCS OR POST ON STACK OVERFLOW.