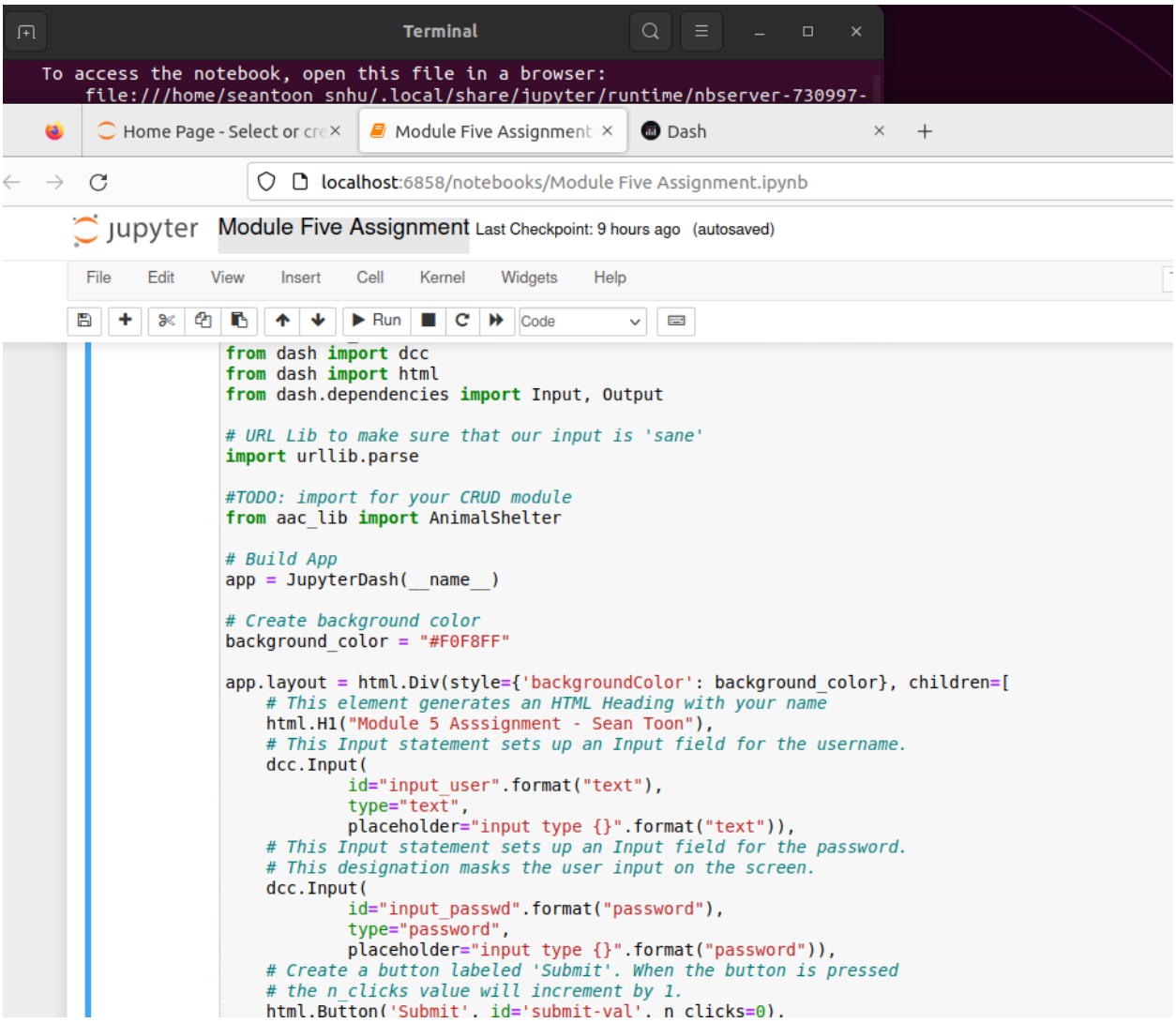
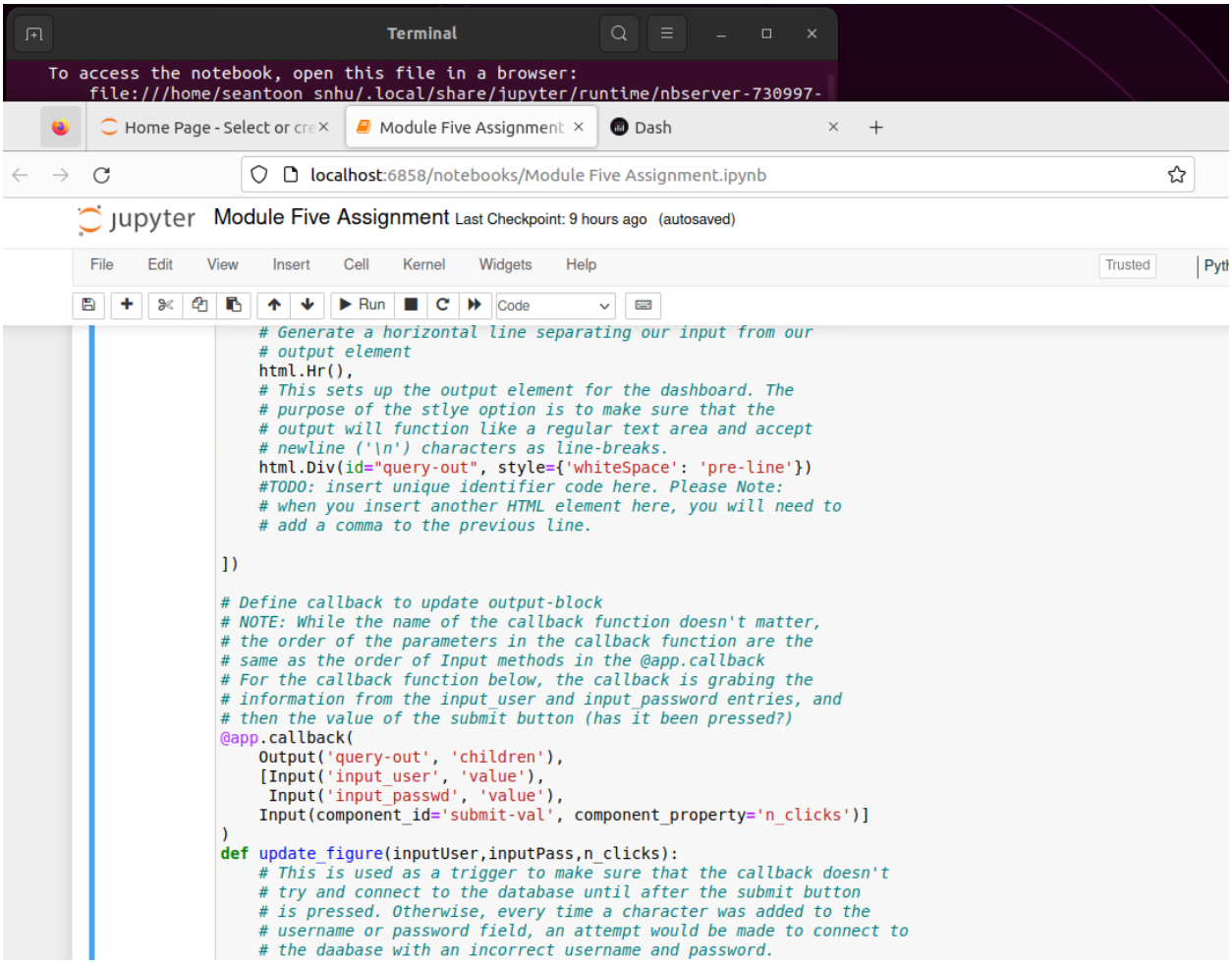
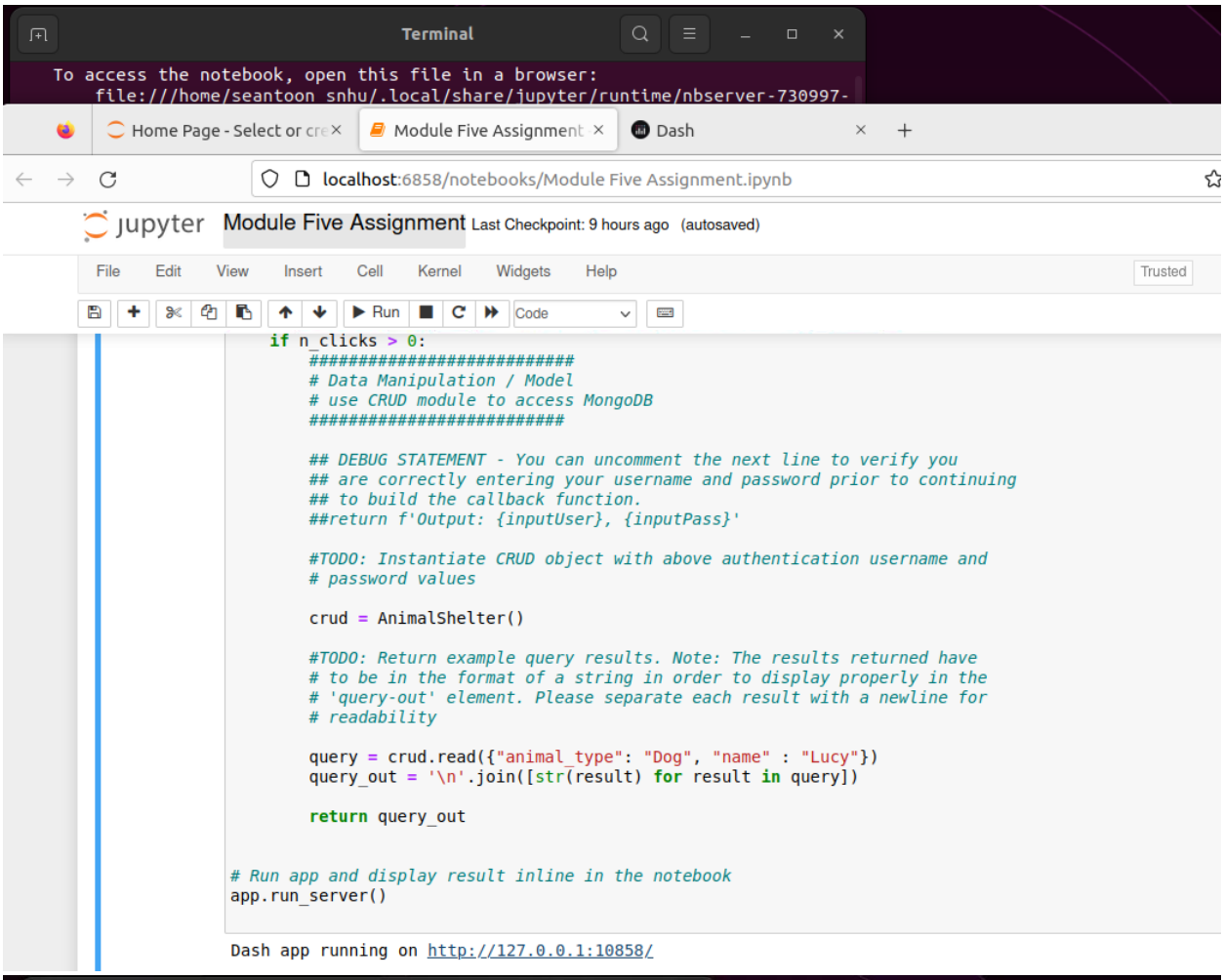

Terminal
To access the notebook, open this file in a browser:
file:///home/seantoon_snhu/.local/share/jupyter/runtime/nbserver-730997-
Home Page - Select or cre
Module Five Assignment
Dash
localhost:6858/notebooks/Module Five Assignment.ipynb
jupyter
Module Five Assignment
Last Checkpoint: 9 hours ago (autosaved)
File Edit View Insert Cell Kernel Widgets Help
Run
Code
from dash import dcc
from dash import html
from dash.dependencies import Input, Output
# URL Lib to make sure that our input is 'sane'
import urllib.parse
#TODO: import for your CRUD module
from aac_lib import AnimalShelter
# Build App
app = JupyterDash(__name__)
# Create background color
background_color = "#F0F8FF"
app.layout = html.Div(style={'backgroundColor': background_color}, children=[
# This element generates an HTML Heading with your name
html.H1("Module 5 Asssignment - Sean Toon"),
This Input statement sets up an Input field for the username.
dcc.Input(
id="input_user".format("text"),
type="text",
placeholder="input type {}".format("text")),
# This Input statement sets up an Input field for the password.
# This designation masks the user input on the screen.
dcc.Input(
id="input_passwd".format("password"),
type="password",
placeholder="input type {}".format("password")),
# Create a button labeled 'Submit'. When the button is pressed
# the n_clicks value will increment by 1.
html.Button('Submit', id='submit-val', n_clicks=0),

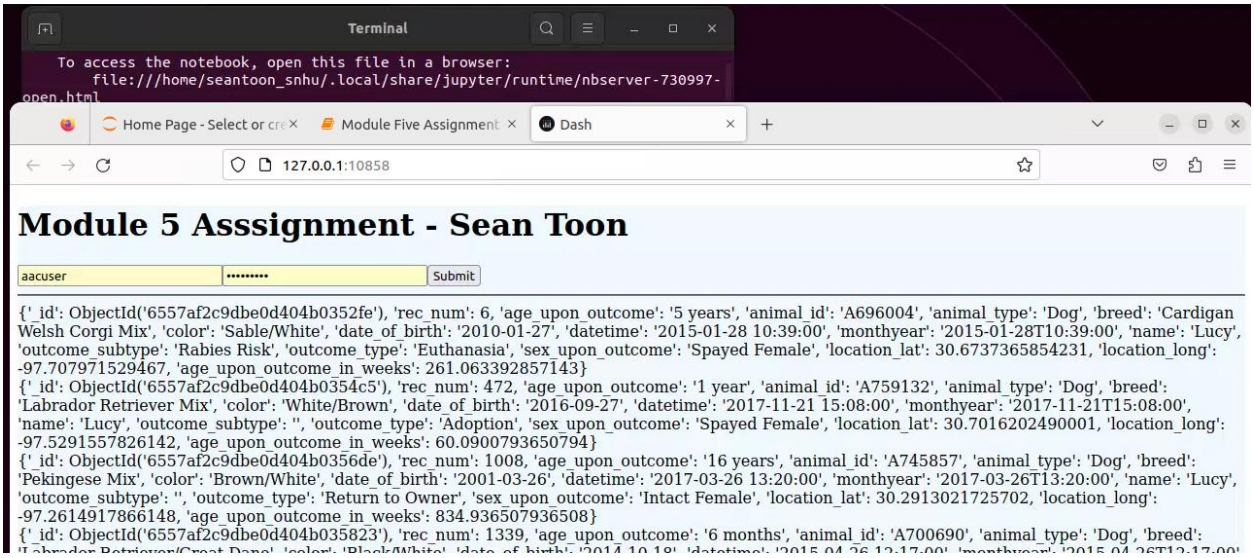



```
    # Generate a horizontal line separating our input from our
    # output element
    html.Hr(),
    # This sets up the output element for the dashboard. The
    # purpose of the stlye option is to make sure that the
    # output will function like a regular text area and accept
    # newline ('\n') characters as line-breaks.
    html.Div(id="query-out", style={'whiteSpace': 'pre-line'})
    #TODO: insert unique identifier code here. Please Note:
    # when you insert another HTML element here, you will need to
    # add a comma to the previous line.

])

# Define callback to update output-block
# NOTE: While the name of the callback function doesn't matter,
# the order of the parameters in the callback function are the
# same as the order of Input methods in the @app.callback
# For the callback function below, the callback is grabing the
# information from the input_user and input_password entries, and
# then the value of the submit button (has it been pressed?)
@app.callback(
    Output('query-out', 'children'),
    [Input('input_user', 'value'),
     Input('input_passwd', 'value'),
    Input(component_id='submit-val', component_property='n_clicks')]
)
def update_figure(inputUser,inputPass,n_clicks):
    # This is used as a trigger to make sure that the callback doesn't
    # try and connect to the database until after the submit button
    # is pressed. Otherwise, every time a character was added to the
    # username or password field, an attempt would be made to connect to
    # the daabase with an incorrect username and password.
```


Terminal
To access the notebook, open this file in a browser:
file:///home/seantoon_snhu/.local/share/jupyter/runtime/nbserver-730997-
Home Page - Select or cre
Module Five Assignment
Dash
localhost:6858/notebooks/Module Five Assignment.ipynb
jupyter
Module Five Assignment
Last Checkpoint: 9 hours ago
(autosaved)
File
Edit
View
Insert
Cell
Kernel
Widgets
Help
Trusted
Run
Code
if n_clicks > 0:
##########################
Data Manipulation / Model
use CRUD module to access MongoDB
##########################
DEBUG STATEMENT - You can uncomment the next line to verify you
are correctly entering your username and password prior to continuing
to build the callback function.
##return f'Output: {inputUser}, {inputPass}'
#TODO: Instantiate CRUD object with above authentication username and
# password values
crud = AnimalShelter()
#TODO: Return example query results. Note: The results returned have
# to be in the format of a string in order to display properly in the
# 'query-out' element. Please separate each result with a newline for
# readability
query = crud.read({"animal_type": "Dog", "name" : "Lucy"})
query_out = '\n'.join([str(result) for result in query])
return query_out
# Run app and display result inline in the notebook
app.run_server()
Dash app running on http://127.0.0.1:10858/


Terminal
To access the notebook, open this file in a browser:
file:///home/seantoon_snhu/.local/share/jupyter/runtime/nbserver-730997-
open.html
Home Page - Select or cre
Module Five Assignment
Dash
127.0.0.1:10858
Module 5 Asssignment - Sean Toon
aacuser
Submit
{'_id': ObjectId('6557af2c9dbe0d404b0352fe'), 'rec_num': 6, 'age_upon_outcome': '5 years', 'animal_id': 'A696004', 'animal_type': 'Dog', 'breed': 'Cardigan Welsh Corgi Mix', 'color': 'Sable/White', 'date_of_birth': '2010-01-27', 'datetime': '2015-01-28 10:39:00', 'monthyear': '2015-01-28T10:39:00', 'name': 'Lucy', 'outcome_subtype': 'Rabies Risk', 'outcome_type': 'Euthanasia', 'sex_upon_outcome': 'Spayed Female', 'location_lat': 30.6737365854231, 'location_long': -97.707971529467, 'age_upon_outcome_in_weeks': 261.063392857143}
{'_id': ObjectId('6557af2c9dbe0d404b0354c5'), 'rec_num': 472, 'age_upon_outcome': '1 year', 'animal_id': 'A759132', 'animal_type': 'Dog', 'breed': 'Labrador Retriever Mix', 'color': 'White/Brown', 'date_of_birth': '2016-09-27', 'datetime': '2017-11-21 15:08:00', 'monthyear': '2017-11-21T15:08:00', 'name': 'Lucy', 'outcome_subtype': '', 'outcome_type': 'Adoption', 'sex_upon_outcome': 'Spayed Female', 'location_lat': 30.7016202490001, 'location_long': -97.5291557826142, 'age_upon_outcome_in_weeks': 60.0900793650794}
{'_id': ObjectId('6557af2c9dbe0d404b0356de'), 'rec_num': 1008, 'age_upon_outcome': '16 years', 'animal_id': 'A745857', 'animal_type': 'Dog', 'breed': 'Pekingese Mix', 'color': 'Brown/White', 'date_of_birth': '2001-03-26', 'datetime': '2017-03-26 13:20:00', 'monthyear': '2017-03-26T13:20:00', 'name': 'Lucy', 'outcome_subtype': '', 'outcome_type': 'Return to Owner', 'sex_upon_outcome': 'Intact Female', 'location_lat': 30.2913021725702, 'location_long': -97.2614917866148, 'age_upon_outcome_in_weeks': 834.936507936508}
{'_id': ObjectId('6557af2c9dbe0d404b035823'), 'rec_num': 1339, 'age_upon_outcome': '6 months', 'animal_id': 'A700690', 'animal_type': 'Dog', 'breed':