**Course Project**

**DeVry University**

**College of Engineering and Information Sciences**

**Course Number: CEIS110**

# Module 3: Downloading Weather Data

# Objectives

* To practice writing and executing Python programs using Anaconda Spyder IDE
* To learn how to use download data from the cloud using an API
* To learn how to create a relational database
* To learn how to save data into a database table using SQL and Python

# Parts List

Equipment:

* Internet-connected PC running the Anaconda Spyder IDE

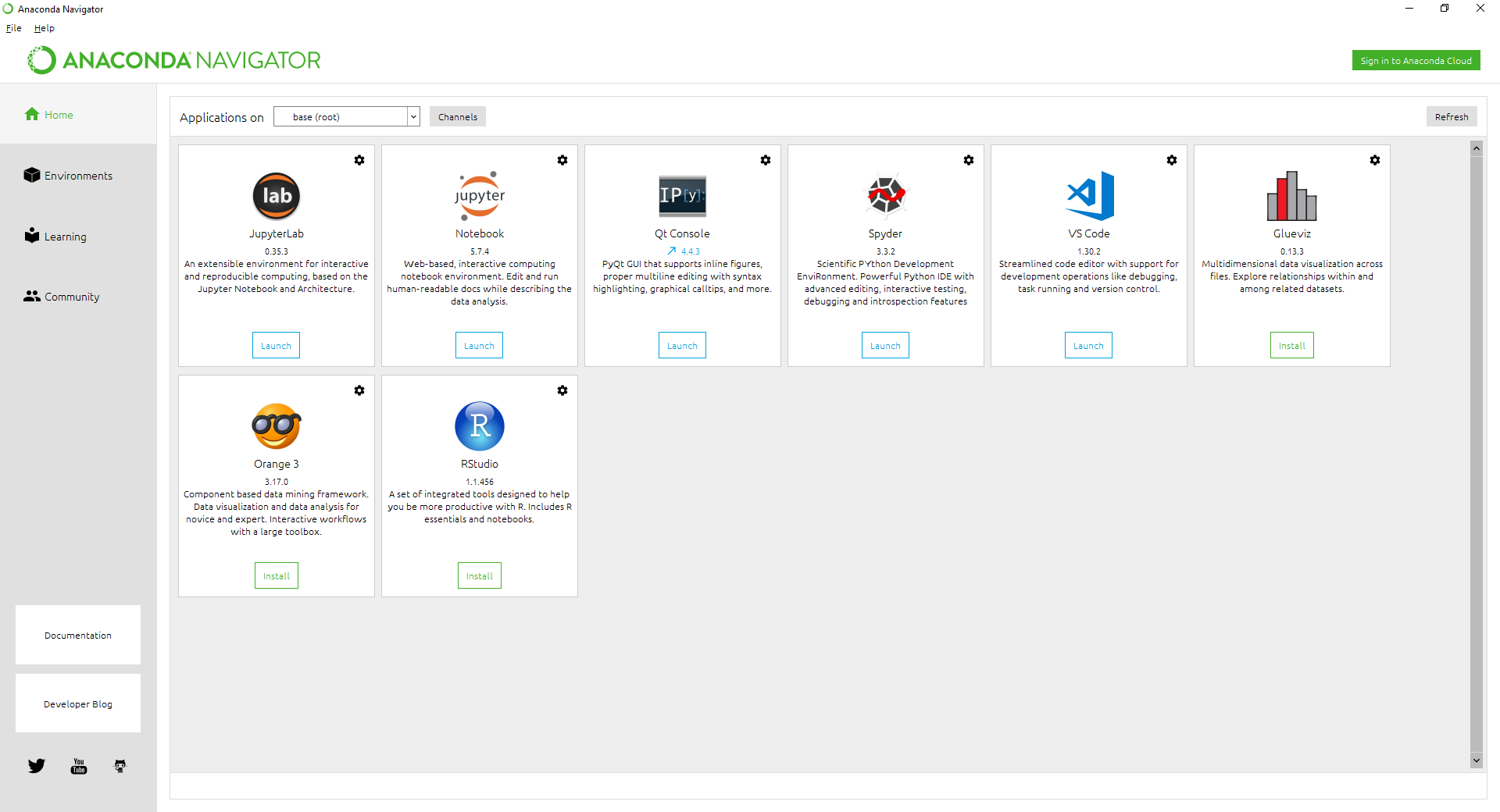
# Introduction

**Downloading Weather Data**

The US National Oceanic and Atmospheric Administration (NOAA) provides free access to nationwide weather observations via a cloud-based Application Programming Interface (API). You will create and run a Python program to download a data set of recent weather observations for your location. Your Python program will create a database on your computer’s hard drive and store the weather data in a table for later analysis.

# Steps

1. Open Anaconda Navigator and click Launch on the Spyder icon:



1. Copy the following code into the window. You will need to modify a few things:
2. Change the name and date to your own
3. Change the zip code to your own local 5-digit zip code
4. Save your program as BuildWeatherDb.py. **NOTE:** You should create a CEIS110 class folder, if you have not already done so, and save this and all your other Python files for this project into that folder. Your weather database will be created in the same folder where your Python code files are saved. All Python programs must be in the same folder as the database in order to access the data.
5. Please pay attention to the create table command in the following code. In the createTableCmd you will be creating an observations table with the fields: timestamp, windSpeed, temperature, relativeHumidity, windDirection, barometricPressure, visibility, and textDescription. The data types of each field are listed next to the field names below. The insert command enters the data into the table.

***#Purpose: Build weather database from NOAA data***

***#Name: Your name***

***#Date: the date***

***# See https://pypi.org/project/noaa-sdk/ for details on noaa\_sdk package used***

***from noaa\_sdk import noaa***

***import sqlite3***

***import datetime***

***# parameters for retrieving NOAA weather data***

***zipCode = "90808" # change to your postal code***

***country = "US"***

***#date-time format is yyyy-mm-ddThh:mm:ssZ, times are Zulu time (GMT)***

***#gets the most recent 14 days of data***

***today = datetime.datetime.now()***

***past = today - datetime.timedelta(days=14)***

***startDate = past.strftime("%Y-%m-%dT00:00:00Z")***

***endDate = today.strftime("%Y-%m-%dT23:59:59Z")***

***#create connection - this creates database if not exist***

***print("Preparing database...")***

***dbFile = "weather.db"***

***conn = sqlite3.connect(dbFile)***

***#create cursor to execute SQL commands***

***cur = conn.cursor()***

***#drop previous version of table if any so we start fresh each time***

***dropTableCmd = "DROP TABLE IF EXISTS observations;"***

***cur.execute(dropTableCmd)***

***#create new table to store observations***

***createTableCmd = """ CREATE TABLE IF NOT EXISTS observations (***

***timestamp TEXT NOT NULL PRIMARY KEY,***

***windSpeed REAL,***

***temperature REAL,***

***relativeHumidity REAL,***

***windDirection INTEGER,***

***barometricPressure INTEGER,***

***visibility INTEGER,***

***textDescription TEXT***

***) ; """***

***cur.execute(createTableCmd)***

***print("Database prepared")***

***# Get hourly weather observations from NOAA Weather Service API***

***print("Getting weather data...")***

***n = noaa.NOAA()***

***observations = n.get\_observations(zipCode,country,startDate,endDate)***

***#populate table with weather observations***

***print("Inserting rows...")***

***insertCmd = """ INSERT INTO observations***

***(timestamp, windSpeed, temperature, relativeHumidity,***

***windDirection, barometricPressure, visibility, textDescription)***

***VALUES***

***(?, ?, ?, ?, ?, ?, ?, ?) """***

***count = 0***

***for obs in observations:***

***insertValues = (obs["timestamp"],***

***obs["windSpeed"]["value"],***

***obs["temperature"]["value"],***

***obs["relativeHumidity"]["value"],***

***obs["windDirection"]["value"],***

***obs["barometricPressure"]["value"],***

***obs["visibility"]["value"],***

***obs["textDescription"])***

***cur.execute(insertCmd, insertValues)***

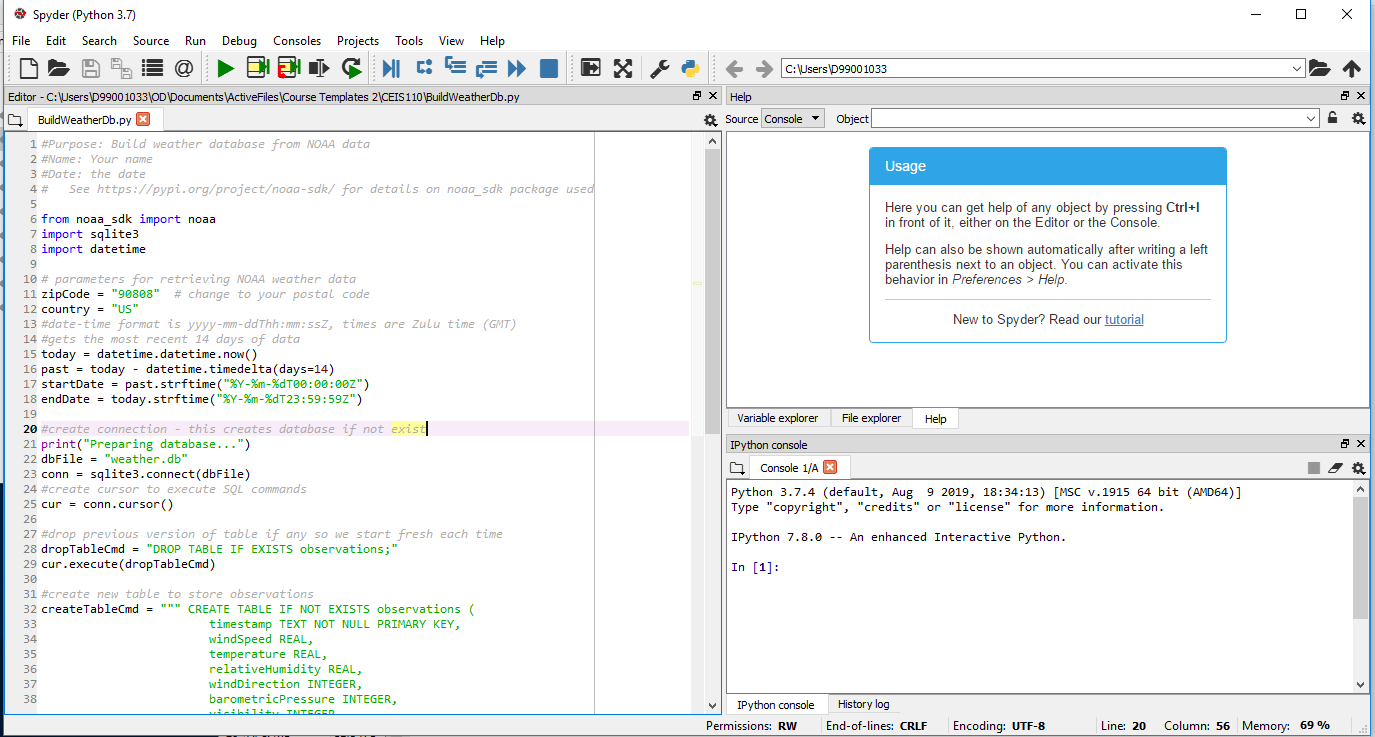
***count += 1***

***if count > 0:***

***cur.execute("COMMIT;")***

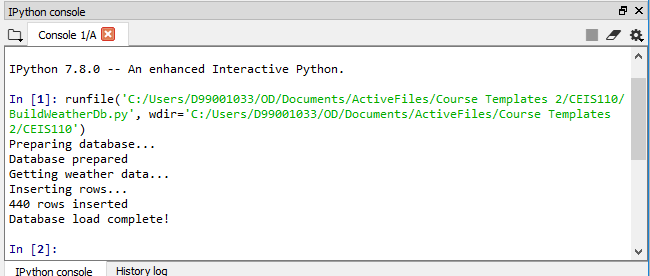
***print(count, "rows inserted")***

***print("Database load complete!")***



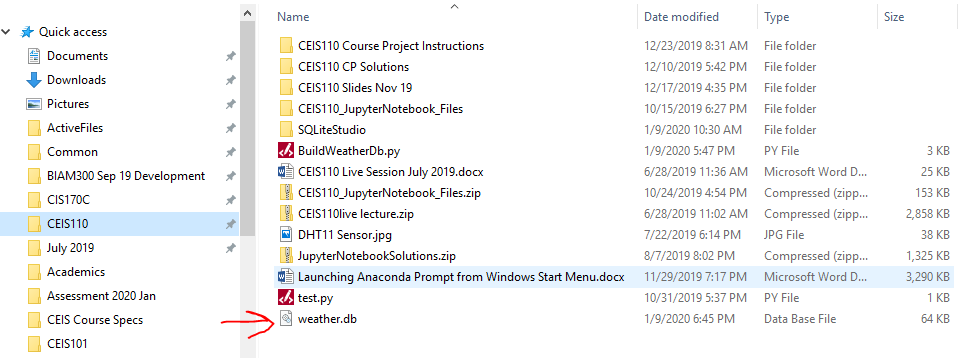
1. After saving, run your Python code by clicking the Run button (green triangle in the toolbar) or going to Run > Run (F5) in the menus.

The program may take a few minutes to run. When it completes successfully, you should see messages similar to the following in the Python console at the lower right of the Spyder window:



Your number of rows inserted may be different from what is shown above, but it should be a few hundred rows. Capture a screenshot of this output to document that your program ran successfully.

1. Check the contents of your CEIS110 folder (or wherever your BuildWeatherDb.py program file was saved) to verify that a database file named weather.db was created in this folder. Take a screenshot showing the listing for this file. (You will not be able to open the file and view its contents, just take a screenshot of Windows Explorer to show that the file is there.)



# Deliverables Module 3

* Complete the Course Project Template presentation
* Include a screenshot from Spyder showing your Python code with your name, the date, and your zip code.
* Include a screenshot of the Python console messages showing your program ran successfully.
* Include a screenshot from Windows Explorer showing the database file.