iNaturalist Pipeline Overview

Created by Miles McCall

Table of Contents

[Contacts and Links 3](#_Toc15565802)

[Introduction 3](#_Toc15565803)

[Installation Instructions 3](#_Toc15565804)

[Venv – Installing & creating virtual environments 3](#_Toc15565805)

[Python 3.6 4](#_Toc15565806)

[Python 3.6 Libraries 4](#_Toc15565807)

[Openpyxl 4](#_Toc15565808)

[Requests 4](#_Toc15565809)

[Important Files & Folders 4](#_Toc15565810)

[Files 4](#_Toc15565811)

[Format\_Data.py 4](#_Toc15565812)

[Col\_functions.py 4](#_Toc15565813)

[Data/OR\_Cities.csv 4](#_Toc15565814)

[Data/usernames.csv 4](#_Toc15565815)

[Makefile 4](#_Toc15565816)

[Folders 5](#_Toc15565817)

[Data 5](#_Toc15565818)

[Elevations 5](#_Toc15565819)

[Results 5](#_Toc15565820)

[Running the Pipeline 5](#_Toc15565821)

[Usage Examples 5](#_Toc15565822)

# Contacts and Links

This program was created by Miles McCall, contact regarding any questions or concerns at:

* [mccallm@oregonstate.edu](mailto:mccallm@oregonstate.edu)
* [milescmccall@gmail.com](mailto:milescmccall@gmail.com)

The project can be found on GitHub at:

* <https://github.com/mccallm1/BeeLab.git>

# Introduction

The iNaturalist Pipeline is a program designed to ingest spreadsheet data queried from the iNaturalist website and produce cleaned, parsed, and normalized tables as output. It consists of several Python 3 files and a corresponding file structure to organize input and output data. The goal of this pipeline is to streamline the process of downloading iNaturalist data and performing a specific action to each column value from the query, such as converting date values.

# Installation Instructions

To install and run the pipeline the following software is required to be installed on the intended machine:

## Venv – [Installing & creating virtual environments](https://docs.python.org/3/library/venv.html)

I recommend creating a virtual environment to work in while running the pipeline. Install Venv compatible with python 3.6. This allows for better software version control and will help ensure the program functions properly as updates are released. The program is written in Python 3.6 and should be compatible with all versions of Python 3.

Install the following requirements within the virtual environment to avoid any conflict with local versions of programs currently installed on the operating system.

## Python 3.6

## Python 3.6 Libraries

### Openpyxl

Openpyxl is a python library that provides tools to read from and write to Microsoft Excel files.

### Requests

Requests allows the elevation function to make url requests to Google Elevation API.

# Important Files & Folders

## **Files**

### Format\_Data.py

Format\_data is the main python script run by the user. The user provides input and output parameters through the command line at run time, and results are printed to screen as the pipeline progresses.

### Col\_functions.py

Col\_functions is a secondary Python script that contains all functions called to process input data values. For each column in the input spreadsheets there is a corresponding function to generate a desired output. The program is structured in this manner to allow each column to be parsed and processed separately based on inputted criteria.

### Data/OR\_Cities.csv

The OR\_Cities CSV file contains a quick reference list of the majority of Oregon cities.

### Data/usernames.csv

The Usernames CSV file creates a map between an iNaturalist user’s full name, username, and id number.

### Makefile

The Makefile provides examples of use cases when running the pipeline.

## **Folders**

### Data

The data folder contains all input data to be processed by the pipeline. Within the folder there is a folder for each batch of inputs. These are usually named by the date of the query for an easy distinction, but folders can be named by any criteria.

### Elevations

The elevations folder saves the elevation results calculated for any batch of inputs in a corresponding folder. This minimizes the number of API calls to Google’s elevation service in the event of rerunning the script on the same batch of input multiple times.

### Results

The results folder creates a mirrored file structure to the input data folder. The results CSV files are created and stored based on the input file name and contain output data in the desired format.

# Running the Pipeline

For best results running the pipeline, start the script from the following location:

* BeeLab/iNaturalist/src/

To run the pipeline, execute the following command within the virtual environment:

* Python format\_data.py

Additionally, an input value must be provided as well:

* Python format\_data.py –input data/6\_12\_19/observation-123456.csv

Lastly, providing an output value will override the default of naming the results folder after the input folder:

* Python format\_data.py –input data/6\_12\_19/observation-123456.csv –output final\_res\_example