Utilising Cygwin, Python & Octave to obtain daily & hourly climate data for multiple months, years, and stations from climate.weather.gc.ca
• **Why?**
  • MSc students’ requests
  • (my current research)

• Environment Canada historical climate / weather data via:
  • web
  • Cygwin
  • Python
  • Octave

• raster GIS data
student wanted data for Mt.Fromme, BC
5 stations found with name containing "North Vancouver", with data available between 1840 and 2017. Stations are listed in alphabetical order. Confirm the Data Interval and the date for one of the stations listed and click "GO" to display the historical data.

<table>
<thead>
<tr>
<th>Station</th>
<th>Prov.</th>
<th>Data Interval</th>
<th>Year</th>
<th>Month</th>
<th>Day</th>
</tr>
</thead>
<tbody>
<tr>
<td>NORTH VANCOUVER BLUE RIDGE</td>
<td>BC</td>
<td>Daily</td>
<td>1989</td>
<td>Sep</td>
<td>30</td>
</tr>
<tr>
<td>NORTH VANCOUVER CITY</td>
<td>BC</td>
<td>Daily</td>
<td>1961</td>
<td>Dec</td>
<td>31</td>
</tr>
<tr>
<td>NORTH VANCOUVER DOLLARTON</td>
<td>BC</td>
<td>Daily</td>
<td>1991</td>
<td>Jan</td>
<td>31</td>
</tr>
<tr>
<td>NORTH VANCOUVER HOLYROOD</td>
<td>BC</td>
<td>Daily</td>
<td>1968</td>
<td>Dec</td>
<td>31</td>
</tr>
<tr>
<td>NORTH VANCOUVER NORGATE</td>
<td>BC</td>
<td>Daily</td>
<td>1963</td>
<td>Apr</td>
<td>30</td>
</tr>
</tbody>
</table>
Mt. Fromme restoration
results for Grouse

1 station found with name containing "Grouse", with data available between 1840 and 2017. Stations are listed in alphabetical order. Confirm the Data Interval and the date for one of the stations listed and click "GO" to display the historical data.

<table>
<thead>
<tr>
<th>Station</th>
<th>Prov.</th>
<th>Data Interval</th>
<th>Year</th>
<th>Month</th>
<th>Day</th>
</tr>
</thead>
<tbody>
<tr>
<td>N VANC GROUSE MTN RESORT</td>
<td>BC</td>
<td>Daily</td>
<td>2017</td>
<td>Nov</td>
<td>11</td>
</tr>
</tbody>
</table>

[Go button]
EnvironmentCanada weather stations as of 2012 in this dataset (not all active)
### Mt. Fromme Restoration

<table>
<thead>
<tr>
<th>Station Name</th>
<th>N VANC GROUSE MTN RESORT</th>
</tr>
</thead>
<tbody>
<tr>
<td>Province</td>
<td>BRITISH COLUMBIA</td>
</tr>
<tr>
<td>Latitude</td>
<td>49.38</td>
</tr>
<tr>
<td>Longitude</td>
<td>123.08</td>
</tr>
<tr>
<td>Elevation</td>
<td>11278</td>
</tr>
<tr>
<td>Climate Identifier</td>
<td>1105658</td>
</tr>
</tbody>
</table>

---

**CanadianWeatherStations - Feature Attributes**

<table>
<thead>
<tr>
<th>ID</th>
<th>842</th>
</tr>
</thead>
<tbody>
<tr>
<td>Station_Name</td>
<td>N VANC GROUSE MTN RESORT</td>
</tr>
<tr>
<td>Province</td>
<td>British Columbia</td>
</tr>
<tr>
<td>Latitude</td>
<td>49°22'05.000&quot;N</td>
</tr>
<tr>
<td>Longitude</td>
<td>123°04'50.000&quot;W</td>
</tr>
<tr>
<td>Elevation</td>
<td>11278</td>
</tr>
<tr>
<td>Climate_ID</td>
<td>1105658</td>
</tr>
</tbody>
</table>
Nurdles (pellets used to make #2-HDPE, #4-LDPE & #5-PP plastics) on Victoria beaches & waterfronts
Nurdles were observed at red sites on map;  
10 Litre samples of beach surface debris, plastics, wood, etc, were collected at 5 sites and nurdles measured in milliLitres
High tide and high winds
MonNov13, 1pm

Willows Beach
Historical Data

To determine data availability for a custom location and date, please complete and submit one of the following searches:

- Search by Station Name
- Search by Province
- Search by Proximity

Name: Victoria Int
- contains
- begins with
- with data available between: 1840 to 2017
- with data on: 2017 November 6

Display 25 results per page.

Search Reset
### Station Results - Historical Data

2 stations found with name containing "Victoria Int", with data available between 1840 and 2017. Stations are listed in alphabetical order. Confirm the Data Interval and the date for one of the stations listed and click "GO" to display the historical data.

<table>
<thead>
<tr>
<th>Station</th>
<th>Prov</th>
<th>Data Interval</th>
<th>Year</th>
<th>Month</th>
<th>Day</th>
</tr>
</thead>
<tbody>
<tr>
<td>VICTORIA INT'L A</td>
<td>BC</td>
<td>Hourly</td>
<td>1953</td>
<td>Jan</td>
<td>1</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>1954</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>1955</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>1956</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>1957</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>1958</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>1959</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>1960</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>1961</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>1962</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>1963</td>
<td></td>
<td></td>
</tr>
<tr>
<td>VICTORIA INT'L A</td>
<td>BC</td>
<td>Hourly</td>
<td>2013</td>
<td>Jul</td>
<td>2</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>2014</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>2015</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>2016</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>2017</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

[Historical Data]
Google maps to obtain Lat / Long

50°22'18.8"N 120°01'08.5"W
50.371894, -120.019035
via the web...

<table>
<thead>
<tr>
<th>Station</th>
<th>Prov.</th>
<th>Data Interval</th>
<th>Year</th>
<th>Month</th>
<th>Day</th>
</tr>
</thead>
<tbody>
<tr>
<td>VICTORIA INT’L A</td>
<td>BC</td>
<td>Hourly</td>
<td>2010</td>
<td>Jan</td>
<td>1</td>
</tr>
</tbody>
</table>

If **hourly** needed, **only by 1 year and 1 month and 1 day** at a time...

<table>
<thead>
<tr>
<th>Station</th>
<th>Prov.</th>
<th>Data Interval</th>
<th>Year</th>
<th>Month</th>
<th>Day</th>
</tr>
</thead>
<tbody>
<tr>
<td>VICTORIA INT’L A</td>
<td>BC</td>
<td>Daily</td>
<td>2013</td>
<td>Jul</td>
<td>10</td>
</tr>
</tbody>
</table>

If **daily** needed, **only by 1 year and 1 month at a time**

<table>
<thead>
<tr>
<th>Station</th>
<th>Prov.</th>
<th>Data Interval</th>
<th>Year</th>
<th>Month</th>
<th>Day</th>
</tr>
</thead>
<tbody>
<tr>
<td>VICTORIA INT’L A</td>
<td>BC</td>
<td>Monthly</td>
<td>2013</td>
<td>Jul</td>
<td>1</td>
</tr>
</tbody>
</table>

If **monthly** needed, **only by 1 year at a time**
1 station found with name containing "grouse", with data available between 1840 and 2017. Stations are listed in alphabetical order. Confirm the Data Interval and the date for one of the stations listed and click "GO" to display the historical data.

<table>
<thead>
<tr>
<th>Station</th>
<th>Prov.</th>
<th>Data Interval</th>
<th>Year</th>
<th>Month</th>
<th>Day</th>
</tr>
</thead>
<tbody>
<tr>
<td>N VANC GROUSE MTN RESORT</td>
<td>BC</td>
<td>Daily</td>
<td>2013</td>
<td>Dec</td>
<td>31</td>
</tr>
</tbody>
</table>

**Daily Data Report for December 2013**

N VANC GROUSE MTN RESORT  
BRITISH COLUMBIA

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Latitude</td>
<td>49°22'52.002&quot; N</td>
</tr>
<tr>
<td>Longitude</td>
<td>123°04'42.004&quot; W</td>
</tr>
<tr>
<td>Elevation</td>
<td>1,103.00 m</td>
</tr>
<tr>
<td>Climate ID</td>
<td>1105658</td>
</tr>
<tr>
<td>WMO ID</td>
<td></td>
</tr>
<tr>
<td>TC ID</td>
<td></td>
</tr>
</tbody>
</table>

**Related Data**  
1981-2010 Climate Normals

**Additional Search Options**  
Nearby Stations with Data  
Historical Data Search

**Download Data**  
Daily Data (2013)  
- CSV  
- XML  
Get More Data
For Daily (data) .csv output provides entire month...  
...but not multiple months or multiple years

<table>
<thead>
<tr>
<th>Date/Time</th>
<th>Year</th>
<th>Month</th>
<th>Day</th>
<th>Time</th>
<th>Temp (°C)</th>
<th>Dew Point Temp (°C)</th>
<th>Rel Hum (%)</th>
<th>Wind Dir (10s deg)</th>
<th>Wind Spd (km/h)</th>
<th>Visibility (km)</th>
<th>Stn Press (kPa)</th>
<th>Hmdx</th>
<th>Wind Chill</th>
<th>Weather</th>
</tr>
</thead>
<tbody>
<tr>
<td>2010-01-01 1:00</td>
<td>2010</td>
<td>1</td>
<td>1</td>
<td>1:00</td>
<td>4.4</td>
<td>4.1</td>
<td>98</td>
<td>28</td>
<td>7</td>
<td>8</td>
<td>100.61</td>
<td>Fog</td>
<td></td>
<td></td>
</tr>
<tr>
<td>2010-01-01 2:00</td>
<td>2010</td>
<td>1</td>
<td>1</td>
<td>2:00</td>
<td>4.5</td>
<td>4.4</td>
<td>99</td>
<td>32</td>
<td>4</td>
<td>9.7</td>
<td>100.48</td>
<td>Fog</td>
<td></td>
<td></td>
</tr>
<tr>
<td>2010-01-01 3:00</td>
<td>2010</td>
<td>1</td>
<td>1</td>
<td>3:00</td>
<td>4.9</td>
<td>4.8</td>
<td>99</td>
<td>9</td>
<td>0</td>
<td>8</td>
<td>100.53</td>
<td>Fog</td>
<td></td>
<td></td>
</tr>
<tr>
<td>2010-01-01 4:00</td>
<td>2010</td>
<td>1</td>
<td>1</td>
<td>4:00</td>
<td>4.9</td>
<td>4.8</td>
<td>99</td>
<td>30</td>
<td>4</td>
<td>8</td>
<td>100.29</td>
<td>Fog</td>
<td></td>
<td></td>
</tr>
<tr>
<td>2010-01-01 5:00</td>
<td>2010</td>
<td>1</td>
<td>1</td>
<td>5:00</td>
<td>5.4</td>
<td>5.2</td>
<td>99</td>
<td>30</td>
<td>7</td>
<td>6.4</td>
<td>100.24</td>
<td>Rain,Fog</td>
<td></td>
<td></td>
</tr>
<tr>
<td>2010-01-01 6:00</td>
<td>2010</td>
<td>1</td>
<td>1</td>
<td>6:00</td>
<td>5.6</td>
<td>5.5</td>
<td>99</td>
<td>27</td>
<td>4</td>
<td>4.8</td>
<td>100.18</td>
<td>Rain,Fog</td>
<td></td>
<td></td>
</tr>
<tr>
<td>2010-01-01 7:00</td>
<td>2010</td>
<td>1</td>
<td>1</td>
<td>7:00</td>
<td>5.5</td>
<td>5.3</td>
<td>99</td>
<td>0</td>
<td>12.9</td>
<td>100.14</td>
<td>Rain,Fog</td>
<td>Rain,Fog</td>
<td></td>
<td></td>
</tr>
<tr>
<td>2010-01-01 8:00</td>
<td>2010</td>
<td>1</td>
<td>1</td>
<td>8:00</td>
<td>5.5</td>
<td>5.5</td>
<td>100</td>
<td>0</td>
<td>16.1</td>
<td>100.12</td>
<td>Rain,Fog</td>
<td>Rain,Fog</td>
<td></td>
<td></td>
</tr>
<tr>
<td>2010-01-01 9:00</td>
<td>2010</td>
<td>1</td>
<td>1</td>
<td>9:00</td>
<td>6.2</td>
<td>6.1</td>
<td>99</td>
<td>12</td>
<td>7</td>
<td>16.1</td>
<td>100.09</td>
<td>Rain,Fog</td>
<td></td>
<td></td>
</tr>
<tr>
<td>2010-01-01 10:00</td>
<td>2010</td>
<td>1</td>
<td>1</td>
<td>10:00</td>
<td>7.1</td>
<td>6.9</td>
<td>99</td>
<td>14</td>
<td>9</td>
<td>19.3</td>
<td>100.06</td>
<td>Rain,Fog</td>
<td></td>
<td></td>
</tr>
<tr>
<td>2010-01-01 11:00</td>
<td>2010</td>
<td>1</td>
<td>1</td>
<td>11:00</td>
<td>8.3</td>
<td>7.7</td>
<td>96</td>
<td>16</td>
<td>11</td>
<td>12.9</td>
<td>100.12</td>
<td>Rain,Fog</td>
<td></td>
<td></td>
</tr>
<tr>
<td>2010-01-01 12:00</td>
<td>2010</td>
<td>1</td>
<td>1</td>
<td>12:00</td>
<td>9.3</td>
<td>8.2</td>
<td>93</td>
<td>13</td>
<td>13</td>
<td>24.1</td>
<td>100.02</td>
<td>Rain,Fog</td>
<td></td>
<td></td>
</tr>
<tr>
<td>2010-01-01 13:00</td>
<td>2010</td>
<td>1</td>
<td>1</td>
<td>13:00</td>
<td>9.5</td>
<td>8.2</td>
<td>92</td>
<td>14</td>
<td>15</td>
<td>12.9</td>
<td>100.07</td>
<td>Rain,Fog</td>
<td></td>
<td></td>
</tr>
<tr>
<td>2010-01-01 14:00</td>
<td>2010</td>
<td>1</td>
<td>1</td>
<td>14:00</td>
<td>9.2</td>
<td>8.1</td>
<td>93</td>
<td>17</td>
<td>20</td>
<td>16.1</td>
<td>100.1</td>
<td>Rain,Fog</td>
<td></td>
<td></td>
</tr>
<tr>
<td>2010-01-01 15:00</td>
<td>2010</td>
<td>1</td>
<td>1</td>
<td>15:00</td>
<td>8.8</td>
<td>7.9</td>
<td>94</td>
<td>17</td>
<td>20</td>
<td>4.8</td>
<td>100.19</td>
<td>Moderate Rain,Fog</td>
<td></td>
<td></td>
</tr>
<tr>
<td>2010-01-01 16:00</td>
<td>2010</td>
<td>1</td>
<td>1</td>
<td>16:00</td>
<td>8.2</td>
<td>7.7</td>
<td>97</td>
<td>14</td>
<td>15</td>
<td>3.2</td>
<td>100.24</td>
<td>Rain,Fog</td>
<td></td>
<td></td>
</tr>
<tr>
<td>2010-01-01 17:00</td>
<td>2010</td>
<td>1</td>
<td>1</td>
<td>17:00</td>
<td>8.6</td>
<td>7.7</td>
<td>94</td>
<td>18</td>
<td>20</td>
<td>6.4</td>
<td>100.3</td>
<td>Rain,Fog</td>
<td></td>
<td></td>
</tr>
<tr>
<td>2010-01-01 18:00</td>
<td>2010</td>
<td>1</td>
<td>1</td>
<td>18:00</td>
<td>9.1</td>
<td>7.2</td>
<td>88</td>
<td>21</td>
<td>17</td>
<td>6.4</td>
<td>100.43</td>
<td>Rain,Fog</td>
<td></td>
<td></td>
</tr>
<tr>
<td>2010-01-01 19:00</td>
<td>2010</td>
<td>1</td>
<td>1</td>
<td>19:00</td>
<td>8.7</td>
<td>7.7</td>
<td>93</td>
<td>14</td>
<td>9</td>
<td>4.8</td>
<td>100.57</td>
<td>Rain,Fog</td>
<td></td>
<td></td>
</tr>
<tr>
<td>2010-01-01 20:00</td>
<td>2010</td>
<td>1</td>
<td>1</td>
<td>20:00</td>
<td>8.8</td>
<td>7.5</td>
<td>92</td>
<td>23</td>
<td>11</td>
<td>16.1</td>
<td>100.71</td>
<td>Rain,Fog</td>
<td></td>
<td></td>
</tr>
<tr>
<td>2010-01-01 21:00</td>
<td>2010</td>
<td>1</td>
<td>1</td>
<td>21:00</td>
<td>7.9</td>
<td>7.4</td>
<td>97</td>
<td>24</td>
<td>4</td>
<td>19.3</td>
<td>100.84</td>
<td>Rain,Fog</td>
<td></td>
<td></td>
</tr>
<tr>
<td>2010-01-01 22:00</td>
<td>2010</td>
<td>1</td>
<td>1</td>
<td>22:00</td>
<td>7.9</td>
<td>6.9</td>
<td>93</td>
<td>20</td>
<td>15</td>
<td>24.1</td>
<td>100.94</td>
<td>Rain,Fog</td>
<td></td>
<td></td>
</tr>
<tr>
<td>2010-01-01 23:00</td>
<td>2010</td>
<td>1</td>
<td>1</td>
<td>23:00</td>
<td>9.1</td>
<td>6.7</td>
<td>85</td>
<td>23</td>
<td>15</td>
<td>32.2</td>
<td>101.03</td>
<td>Rain,Fog</td>
<td></td>
<td></td>
</tr>
<tr>
<td>2010-01-02 0:00</td>
<td>2010</td>
<td>1</td>
<td>1</td>
<td>0:00</td>
<td>9.1</td>
<td>6.7</td>
<td>85</td>
<td>24</td>
<td>13</td>
<td>24.1</td>
<td>101.13</td>
<td>Rain,Fog</td>
<td></td>
<td></td>
</tr>
<tr>
<td>2010-01-02 1:00</td>
<td>2010</td>
<td>1</td>
<td>1</td>
<td>1:00</td>
<td>9</td>
<td>6.2</td>
<td>83</td>
<td>24</td>
<td>20</td>
<td>24.1</td>
<td>101.23</td>
<td>Rain,Fog</td>
<td></td>
<td></td>
</tr>
<tr>
<td>2010-01-02 2:00</td>
<td>2010</td>
<td>1</td>
<td>1</td>
<td>2:00</td>
<td>8.9</td>
<td>5.7</td>
<td>80</td>
<td>23</td>
<td>22</td>
<td>24.1</td>
<td>101.35</td>
<td>Mostly Cloudy</td>
<td></td>
<td></td>
</tr>
<tr>
<td>2010-01-02 3:00</td>
<td>2010</td>
<td>1</td>
<td>1</td>
<td>3:00</td>
<td>8.8</td>
<td>5.6</td>
<td>80</td>
<td>22</td>
<td>22</td>
<td>32.2</td>
<td>101.43</td>
<td>Mostly Cloudy</td>
<td></td>
<td></td>
</tr>
<tr>
<td>2010-01-02 4:00</td>
<td>2010</td>
<td>1</td>
<td>1</td>
<td>4:00</td>
<td>8.8</td>
<td>5.3</td>
<td>79</td>
<td>24</td>
<td>7</td>
<td>32.2</td>
<td>101.51</td>
<td>Cloudy</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
for Daily or Hourly (data)
b/c .csv does not provide...
...multiple months or multiple years

Hourly Data Report for January 01, 2010

All times are specified in Local Standard Time (LST). Add 1 hour to adjust for Daylight Saving Time where and when it is observed.

VICTORIA INTL A
BRITISH COLUMBIA

Latitude: 48°38'50.010" N  Longitude: 123°25'33.000" W  Elevation: 19.50 m

Climate ID: 1018620  WMO ID: 71799  TC ID: YYJ

Related Data
- Almanac Averages & Extremes (January 01)
- Daily Data (January 2010)
- Monthly Data (2010)
- 1981-2010 Climate Normals

Additional Search Options
- Nearby Stations with Data
- Historical Data Search

Download Data
- Hourly Data (January 2010)
  - CSV  ○  XML
  - Download Data
  - Get More Data
Index of ftp://client_climate@ftp.tor.ec.gc.ca/Pub/Get_More_Data_Plus_de_donnees/

<table>
<thead>
<tr>
<th>Name</th>
<th>Size</th>
<th>Last Modified</th>
</tr>
</thead>
<tbody>
<tr>
<td>Lisezmoi.txt</td>
<td>3 KB</td>
<td>2016-05-11 12:00:00 AM</td>
</tr>
<tr>
<td>Readme.txt</td>
<td>3 KB</td>
<td>2016-12-20 12:00:00 AM</td>
</tr>
<tr>
<td>Répertoire des stations FR.csv</td>
<td>1302 KB</td>
<td>2017-11-05 8:32:00 PM</td>
</tr>
<tr>
<td>Station Inventory EN.csv</td>
<td>1293 KB</td>
<td>2017-11-05 8:32:00 PM</td>
</tr>
<tr>
<td>Suggestions_on_installing_Cygwin_and_running_the_command_line_to_download_dat...</td>
<td>107 KB</td>
<td>2017-10-05 12:41:00 PM</td>
</tr>
</tbody>
</table>
EC instructions to install **Cygwin**

**Cygwin**: Unix-like environment and command-line interface for Windows...

Version: 2016-10-02

Suggestions on installing Cygwin, and running command lines to download data:

1. **Installing Cygwin.**
   Further to downloading the compatible bit-version of Cygwin in your computer (32/64 bit), one of the reasons why the command line may not generate csv files with data is because there may a little installation missteps when setting up Cygwin. The following are some directions that you may follow in order to set the program properly and solve issues:
   - When prompt to the window “Available Download Sites” select the option “cygwin.mirrors.hoobly.com”
I got mixed up confusion
Man, it’s a-killing me...

Well, my head’s full of questions
My temperature’s rising fast
Well, I’m looking for some answers
But I don’t know who to ask...

(Mixed up confusion, 1962, with apologies to Nobel Laureate Bob Dylan)

So while waiting for answers from EC,
one person from EC suggested...
import sys
import os.path
import string
import os
import urllib2
import csv

# Title: Bulk CBOC Download for daily data only from Climate data online
# Author: Unique Lapinskas
# Last modified: 14 11 2016
# Comments: This program downloads bulk data from CBO It requires the user to change the following:
# HOMEEDIR: make sure you put a folder that exists on your computer
# stn_input.txt: this file must be in the HOMEEDIR folder and must contain the stnID in a column. See the attached example
# The output file is called output.csv and will be located in your HOMEEDIR.

# SET DIRECTORIES
# ATTENTION - ATTENTION - MAKE SURE TO PUT THE "/" AT THE END OF THE FOLDER NAME
# MAKE SURE TO FOLLOW THIS FORMAT: '/climate/tools/wget/
# Cask BahnLive, so make sure that is exactly correct also
HOMEEDIR= 'C:\bramblers\Data_Sources\Python\Updated_Scripts_July_2016\Giselle'/

HOMEEDIR= 'C:\Data\EnvrnCan\CanPython/'

#step 1 - open up the station list and put the IDs in a list
staname = HOMEEDIR + "stn_input.txt"
stnlist = []
with open(staname, 'r') as lines:
    for line in lines:
        stnID = line.strip()
        for intYr in range(2015,2016+1):
            for intMon in range(1,12+1):
                #build the query
                strQry = 'http://climate.weather.gc.ca/climate_data/bulk_data_e.html?format=csv&stationID=' + stnID + '&Year=' + str(intYr) + '&Month=' + str(intMon) + '&timeframe=List&disp=Downloads&Data'
                #print strQry
                print 'Querying station ' + stnID + ' for year ' + str(intYr) + ' and month ' + str(intMon)
                try:
                    response = urllib2.urlopen(strQry)
                    with open(HOMEEDIR + "output.csv", 'a') as output:
                        output.write(response.read())
                except Exception, e:
                    print 'Failure getting data for ' + stnID + ' for year ' + str(intYr)
• widely used programming language for general-purpose
• created by Guido van Rossum and first released in 1991
Python 2.7.11 must be 2.7.11
## Files

<table>
<thead>
<tr>
<th>Version</th>
<th>Operating System</th>
<th>Description</th>
<th>MD5 Sum</th>
<th>File Size</th>
<th>GPG</th>
</tr>
</thead>
<tbody>
<tr>
<td>Gzipped source tarball</td>
<td>Source release</td>
<td></td>
<td>6b6076ec9e93f05dd63e47eb9c15728b</td>
<td>16856409</td>
<td>SIG</td>
</tr>
<tr>
<td>XZ compressed source tarball</td>
<td>Source release</td>
<td></td>
<td>1dbcc848b4cd8399a8199f0000f9823c</td>
<td>12227476</td>
<td>SIG</td>
</tr>
<tr>
<td>Mac OS X 32-bit i386/PPC installer</td>
<td>Mac OS X</td>
<td>for Mac OS X 10.5 and later</td>
<td>8d563a63b261fc3868c101471442b601</td>
<td>24018001</td>
<td>SIG</td>
</tr>
<tr>
<td>Mac OS X 64-bit/32-bit installer</td>
<td>Mac OS X</td>
<td>for Mac OS X 10.6 and later</td>
<td>cacd8b6a05c5a5c0f0e19f684a0c7f10</td>
<td>22162527</td>
<td>SIG</td>
</tr>
<tr>
<td>Windows debug information files</td>
<td>Windows</td>
<td></td>
<td>b5ebe6703d69ee971d648d20d6ee55</td>
<td>24359078</td>
<td>SIG</td>
</tr>
<tr>
<td>Windows debug information files for 64-bit binaries</td>
<td>Windows</td>
<td></td>
<td>34b3e9342b7a9d58e020c6108e72e6</td>
<td>25104550</td>
<td>SIG</td>
</tr>
<tr>
<td>Windows help file</td>
<td>Windows</td>
<td></td>
<td>0d804ef1da197c8381b0789c2d5cc98</td>
<td>6171837</td>
<td>SIG</td>
</tr>
<tr>
<td>Windows x86-64 MSI installer</td>
<td>Windows</td>
<td>for AMD64/EM64T/x64</td>
<td>25acc4a2662d4b02682e00df3f3446d</td>
<td>19550208</td>
<td>SIG</td>
</tr>
<tr>
<td>Windows x86 MSI installer</td>
<td>Windows</td>
<td></td>
<td>241bf8e097ab4e1047d9bb4f59602095</td>
<td>18636800</td>
<td>SIG</td>
</tr>
</tbody>
</table>
Python 2.7.11

right-click

Open
Edit with IDLE

CDO_Daily_Data_Bulk_Download.py
CDO_Hourly_Data_Bulk_Download.py

Edit with IDLE (ArcGIS Pro)
Run with ArcGIS Pro
7-Zip
CRC SHA
Edit with Notepad++

Scan for Viruses...

Send to
Cut
Copy
Create shortcut
Delete
Rename
Properties
import sys
import os.path
import string
import os
import urllib2
import csv

#==================================
#
# Title:       Bulk CDO download for daily
# Author:      Monique Lapalme
# Last modified:  le 11 avril 2016
# Comments:     This program downloads bulk data from CDO Daily data
#                to
#                HOMEDIR: make sure you put this file in the correct
#                directory
#                stn_input.txt: this file contains the
#                station data
#                The output file is called output.csv and will be
#                saved in the current
#                directory
#
#==================================
#
# Set directories
# ATTENTION - ATTENTION - MAKE SURE TO PUT THE 'E:'
# MAKE SURE TO FOLLOW THIS FORMAT: 'E:/climate/tools'
# Case SENSITIVE, so make sure that is exactly correct
#
# HOMEDIR= 'C:/bramwel/Gen/Python/Update'

HOMEDIR= 'C:/Data/EnvOnCan/ECanPython/'
import sys
import os.path
import stringing
import csv
import os
import urllib2
import csv

# Title: Bulk CDO download for daily data only from Climate data online
# Author: Monique Lapalme
# Last modified: 14 11 avril 2016
# Comments: This program downloads bulk data from CDO It requires the user to change the following:
# HOMEDIR: make sure you put a folder that exists on your computer
# stn_input.txt: this file must be in the HOMEDIR folder and must contain the stnID in a column. See the attached example
# The output file is called output.csv and will be located in your HOMEDIR

# Set directories
# ATTENTION: ATTENTION: MAKE SURE TO PUT THE ‘/’ AT THE END OF THE FOLDER NAME
# MAKE SURE TO FOLLOW THIS FORMAT: ‘/climate/tools/wget/’
# os.seteuid(oseuid, osenid), so make sure that is exactly correct also
# HOMEDIR: ‘C:s\brasmello\Data_Sources\Python\Updated_Scripts_July_2016\Giselle/’
# HOMEDIR: ‘C:\\Data\\Environ\\CanPython’

# Step 1 - open up the station list and put the IDs in a list
# station = HOMEDIR + "stn_input.txt"
stnlist = []
with open(stnlist, 'r') as lines:
    for line in lines:
        stnID = line.strip()
        for intYr in range(2015, 2016+1):
            # Build the query
            strYr = 'http://climate.weather.gov.cn/climate_data/bulk_data_s.html?format=csv&stnID=' + strYr + '&Year=' + str(intYr) + '&Month=' + str(intMonth) + '&idframe=issubmit=Download+Data
            try:
                #print 'Querying station', stnID, 'for year', strYr, 'and month', str(intMonth)
                response = urllib2.urlopen(strYr)
                if open(HOMEDIR + "output8.csv", 'a') as output:
                    output.write(response.read())
            except Exception, e:
                print 'Failure getting data for', stnID, 'for year', strYr
```python
import sys
import os.path
import string
import os
import urllib2
import csv

# # Title: Bulk CDO downloaded for daily data only from Climate data online
# # Author: Monique Lapelée
# # Last modified: 16 11 avril 2016
# # Comments: This program downloads bulk data from CDO IC requires the user to change the following:
# # HOMEDIR: make sure you put a folder that exists on your computer
# # stn_input.txt: this file must be in the HOME_DIR folder and must contain the stnID in a column. See the attached example
# The output file is called output.cov and will be located in your HOMEDIR
#
#
# #---------------------------------------------------------------
# # Set Directories
# # ATTENTION - ATTENTION - MAKE SURE TO PUT THE '/' AT THE END OF THE FOLDER NAME
# # MAKE SURE TO FOLLOW THIS FORMAT: '/climate/tools/wget/
# # Case SENSITIVE, so make sure that is exactly correct also
# # HOMEDIR= '/C:/braxwell/Data_Sources/Python/Updated_Scripts_July_2016/Giselle/
# HOMEDIR= 'C:\Data\EnvCan\ScanPython/'
#
# #---------------------------------------------------------------

# Step 1: open up the station file and put the IDs in a list
stnFile = HOMEDIR + "stn_input.txt"
with open(stnFile, 'r') as lines:
    stnID = [line.strip() for line in lines]
for intYr in range(2015, 2016+1):
    for intMnt in range(1,12+1):
        for i in stnID:
            urlQuery = http://climate.weather.gc.ca/climate_data/bulk_data_e.html?format=csv&stationID=" + stnID + "," + str(intYr) + "Yr=" + str(intYr) + ",Month=" + str(intMnt) + ",timeframe=1
            print 'Querying station ' + stnID + ' for year ' + str(intYr) + ' and month ' + str(intMnt)
            try:
                response = urllib2.urlopen(urlQuery)
                with open(HOMEDIR + "output.cov", 'a') as output:
                    output.write(response.read())
            except Exception, e:
                print 'Failure getting data for ' + stnID + ' for year ' + str(intYr)
```

31
Python 2.7.11

#step 1 - open up the station list and put the IDs in a list
#
stnFile = HOMEDIR + "stn_input.txt"
stnList =[]
Python 2.7.11

```python
# Step 1 - open up the station list and put the IDs in a list
staFile = BURLDIR + "sta_input.txt"
staList = []
with open(staFile, 'r') as lines:
    for line in lines:
        staID = line.strip()
        # Add the station ID to the list
        staList.append(staID)

for intYr in range(2013,2016+1):
    for intMnt in range(1,12+1):
        # Build the query
        strQry = 'http://climate.weather.gov/climate_data/bulk_data_e.html?format=csv&stationID="' + staID + '&Year="' + str(intYr) + '&Month="' + str(intMnt) + '&timeframe=1&submit=Download+Data"
        print strQry
        try:
            response = urllib2.urlopen(strQry)
            with open(BURLDIR + 'outputX.csv', 'w') as output:
                output.write(response.read())
        except Exception, e:
            print "Voila, C'est fini" + staID + ' for year ' + str(intYr)
        retrieve data from
        year range 2013-2016; the +1 is necessary
        month range 1-12; the +1 is necessary
```
import sys
import os, path
import os
import csv
import urllib2

# Set directories
# ATTENTION - ATTENTION - MAKE SURE TO FOLLOW THIS FORMAT: 'E:/'
# Case Sensitive, so make sure that is a capital E
# If running this script from the command line, make sure to change the path -
# HOME DIR to 'C:\bramellig\Data_Sources\Python'
HOME DIR = 'C:\Data\Environ\ECanPython'

#step 1 - open up the station list and
station file = HOME DIR + "\stn_input.txt"
station list = []
with open(station file, 'r') as lines:
    for line in lines:
        station ID = line.strip()
        for int yr in range(2015, 2014 + 1):
            for int mnt in range(1, 13 + 1):
                #build the query
                str qry = "http://climate.weather.gc.ca/climate_data/bulk_data_e.html?format=csv&stationID='" + station ID + "&Yr:
                print str qry
                try:
                    response = urllib2.urlopen(str qry)
                    with open(HOME DIR + "\output\csv", 'a') as output:
                        output.write(response.read())
                except Exception, e:
                    print 'Failure getting data for ' + station ID + ' for year ' + str(int yr)
Python 2.7.11

output

<table>
<thead>
<tr>
<th>Name</th>
<th>Date modified</th>
<th>Type</th>
</tr>
</thead>
<tbody>
<tr>
<td>CDO_Daily_Data_Bulk.Download.py</td>
<td>2017-10-08 12:27</td>
<td>Python File</td>
</tr>
<tr>
<td>CDO.Hourly_Data_Bulk.Download.py</td>
<td>2017-11-17 8:56 AM</td>
<td>Python File</td>
</tr>
<tr>
<td>stn_input.txt</td>
<td>2017-10-08 12:10</td>
<td>Text Document</td>
</tr>
<tr>
<td>outputH.csv</td>
<td>2017-11-17 9:33 AM</td>
<td>Microsoft Excel</td>
</tr>
</tbody>
</table>
Python 2.7.11

If want to sort, etc need to strip out (extraneous) station info repeated between each month / year
Python 2.7.11: How long will it be supported?

If Python 2.7 converted to Python 3.6 (via 2to3.py) can (historical) climate data still be extracted from EC?
email to EC: Government of Canada Gouvernement du Canada

IF I convert the script to Python 3.6, will the EC server(s) "understand" Python 3.6?

(Or is historical weather data only set-up for Python 2.7?)

Reply:
...if EC is still using the version which uses the urllib2 library...
(If yes)
then no, a version of Python of 3.1 or higher will not work with this script

Reply#2:
EC does not support this way of database extraction...
for Daily (data)  
b/c .csv does not provide...  
...multiple months or multiple years

Hourly Data Report for January 01, 2010

All times are specified in Local Standard Time (LST). Add 1 hour to adjust for Daylight Saving Time where and when it is observed.

VICTORIA INTL A  
BRITISH COLUMBIA

<table>
<thead>
<tr>
<th>Latitude:</th>
<th>48°38'50.010&quot; N</th>
<th>Longitude:</th>
<th>123°25'33.000&quot; W</th>
<th>Elevation:</th>
<th>19.50 m</th>
</tr>
</thead>
<tbody>
<tr>
<td>Climate ID:</td>
<td>1018620</td>
<td>WMO ID:</td>
<td>71799</td>
<td>TC ID:</td>
<td>YYJ</td>
</tr>
</tbody>
</table>

Related Data

- Almanac Averages & Extremes (January 01)
- Daily Data (January 2010)
- Monthly Data (2010)
- 1981-2010 Climate Normals

Additional Search Options

- Nearby Stations with Data
- Historical Data Search

Download Data

- Hourly Data (January 2010)
  - CSV
  - XML

Get More Data
### Cygwin and script

**Index of ftp://client_climate@ftp.tor.ec.gc.ca/Pub/Get_More_Data_Plus_de_donnees/**

<table>
<thead>
<tr>
<th>Name</th>
<th>Size</th>
<th>Last Modified</th>
</tr>
</thead>
<tbody>
<tr>
<td>Lisezmoi.txt</td>
<td>3 KB</td>
<td>2016-05-11 12:00:00 AM</td>
</tr>
<tr>
<td>Readme.txt</td>
<td>3 KB</td>
<td>2016-12-20 12:00:00 AM</td>
</tr>
<tr>
<td>Répertoire des stations FR.csv</td>
<td>1302 KB</td>
<td>2017-11-05 8:32:00 PM</td>
</tr>
<tr>
<td>Station Inventory EN.csv</td>
<td>1293 KB</td>
<td>2017-11-05 8:32:00 PM</td>
</tr>
<tr>
<td><strong>Suggestions_on_installing_Cygwin_and_running_the_command_line_to_download_data_</strong></td>
<td>107 KB</td>
<td>2017-10-05 12:41:00 PM</td>
</tr>
</tbody>
</table>
EC instructions to install Cygwin

**Cygwin**: Unix-like environment and command-line interface for Windows

**Version**: 2016-10-02

Suggestions on installing Cygwin, and running command lines to download data:

1. **Installing Cygwin**
   Further to downloading the compatible bit-version of Cygwin in your computer (32/64 bit), one of the reasons why the command line may not generate csv files with data is because there may a little installation missteps when setting up Cygwin. The following are some directions that you may follow in order to set the program properly and solve issues:

   - When prompt to the window “Available Download Sites” select the option “cygwin.mirrors.huoble.com”
Cygwin and script

Index of ftp://client_climate@ftp.tor.ec.gc.ca/Pub/Get_More_Data_Plus_de_donnees/

Up to higher level directory

<table>
<thead>
<tr>
<th>Name</th>
<th>Size</th>
<th>Last Modified</th>
</tr>
</thead>
<tbody>
<tr>
<td>Lisezmoi.txt</td>
<td>3 KB</td>
<td>2016-05-11 12:00:00 AM</td>
</tr>
<tr>
<td>Readme.txt</td>
<td>3 KB</td>
<td>2016-12-20 12:00:00 AM</td>
</tr>
<tr>
<td>Répertoire des stations FR.csv</td>
<td>1302 KB</td>
<td>2017-11-05 8:32:00 PM</td>
</tr>
<tr>
<td>Station Inventory EN.csv</td>
<td>1293 KB</td>
<td>2017-11-05 8:32:00 PM</td>
</tr>
<tr>
<td>Suggestions_on_installing_Cygwin_and_running_the_command_line_to_download_dat...</td>
<td>107 KB</td>
<td>2017-10-05 12:41:00 PM</td>
</tr>
</tbody>
</table>

this includes the command line script!
EC instructions to install Cygwin
Cygwin: Unix-like environment and command-line interface for Windows

Readme.txt

URL based procedure to automatically download data in bulk from Climate Website
(http://www.climate.weather.gc.ca)
Version: 2016-05-10

-----------------------------------
ENVIRONMENT AND CLIMATE CHANGE CANADA

Command line:

for year in `seq 1998 2008`;do for month in `seq 1 12`;do wget --content-disposition "http://climate.weather.gc.ca/climate_data/bulk_data_e.html?format=csv&stationID=27226&Year=${year}&Month=${month}&Day=14&timeframe=1&submit=Download+Data" ; done;done

NOTE: spaces are important! Copy from EC as 1(one) continuous line

for year in `seq 2006 2008`;do for month in `seq 1 12`;do wget --content-disposition "http://climate.weather.gc.ca/climate_data/bulk_data_e.html?format=csv&stationID=27226&Year=${year}&Month=${month}&Day=14&timeframe=1&submit=Download+Data" ; done;done

WHERE:
• year = change values in command line (`seq 1998 2008`)
• month = change values in command line (`seq 1 12`)
• format= [csv|xml]: the format output
• timeframe = 1: for hourly data
• timeframe = 2: for daily data
• timeframe = 3 for monthly data
• Day: the value of the "day" variable is not used and can be an arbitrary value
• For another station, change the value of the variable stationID
• For the data in XML format, change the value of the variable format to xml in the URL.
Cygwin

separate files for each month...

...bit of a pain?....
...combine / merge multiple .csv (or .xls)  

**NOTE:** the following is for Excel 2016
...combine / merge multiple .csv (or .xls)
...combine / merge multiple .csv (or .xls)
combined / merged files after Excel 2016 (obtained via Cygwin)

<table>
<thead>
<tr>
<th>Date/Time</th>
<th>Year</th>
<th>Month</th>
<th>Day</th>
<th>Time</th>
<th>Temp (°C)</th>
<th>Dew Point Temp (°C)</th>
<th>Rel Hum (%)</th>
<th>Wind Dir (10s deg)</th>
<th>Wind Spd (km/h)</th>
<th>Visibility (km)</th>
<th>Stn Press (kPa)</th>
<th>Hmdx</th>
<th>Wind Chill</th>
<th>Weather</th>
</tr>
</thead>
<tbody>
<tr>
<td>2010-01-01 0:00</td>
<td>2010</td>
<td>1</td>
<td>1</td>
<td>0:00</td>
<td>4.3</td>
<td>4</td>
<td>98</td>
<td>25</td>
<td>6</td>
<td>8</td>
<td>100.64</td>
<td>Fog</td>
<td></td>
<td></td>
</tr>
<tr>
<td>2010-01-01 1:00</td>
<td>2010</td>
<td>1</td>
<td>1</td>
<td>1:00</td>
<td>4.4</td>
<td>4.1</td>
<td>98</td>
<td>28</td>
<td>7</td>
<td>8</td>
<td>100.61</td>
<td>Fog</td>
<td></td>
<td></td>
</tr>
<tr>
<td>2010-01-01 2:00</td>
<td>2010</td>
<td>1</td>
<td>1</td>
<td>2:00</td>
<td>4.5</td>
<td>4.4</td>
<td>99</td>
<td>32</td>
<td>4</td>
<td>9.7</td>
<td>100.48</td>
<td>Fog</td>
<td></td>
<td></td>
</tr>
<tr>
<td>2010-01-01 3:00</td>
<td>2010</td>
<td>1</td>
<td>1</td>
<td>3:00</td>
<td>4.9</td>
<td>4.8</td>
<td>99</td>
<td>0</td>
<td>8</td>
<td>100.53</td>
<td>Fog</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>2010-01-01 4:00</td>
<td>2010</td>
<td>1</td>
<td>1</td>
<td>4:00</td>
<td>4.9</td>
<td>4.8</td>
<td>99</td>
<td>30</td>
<td>4</td>
<td>8</td>
<td>100.29</td>
<td>Fog</td>
<td></td>
<td></td>
</tr>
<tr>
<td>2010-01-01 5:00</td>
<td>2010</td>
<td>1</td>
<td>1</td>
<td>5:00</td>
<td>5.4</td>
<td>5.2</td>
<td>99</td>
<td>30</td>
<td>7</td>
<td>6.4</td>
<td>100.24</td>
<td>Rain,Fog</td>
<td></td>
<td></td>
</tr>
<tr>
<td>2010-01-01 6:00</td>
<td>2010</td>
<td>1</td>
<td>1</td>
<td>6:00</td>
<td>5.6</td>
<td>5.5</td>
<td>99</td>
<td>27</td>
<td>4</td>
<td>4.8</td>
<td>100.18</td>
<td>Rain,Fog</td>
<td></td>
<td></td>
</tr>
<tr>
<td>2010-01-01 7:00</td>
<td>2010</td>
<td>1</td>
<td>1</td>
<td>7:00</td>
<td>5.5</td>
<td>5.3</td>
<td>99</td>
<td>0</td>
<td>12.9</td>
<td>100.14</td>
<td>Rain</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>2010-01-01 8:00</td>
<td>2010</td>
<td>1</td>
<td>1</td>
<td>8:00</td>
<td>5.5</td>
<td>5.5</td>
<td>100</td>
<td>0</td>
<td>15.1</td>
<td>100.12</td>
<td>Rain</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>2010-01-01 9:00</td>
<td>2010</td>
<td>1</td>
<td>1</td>
<td>9:00</td>
<td>6.2</td>
<td>6.1</td>
<td>99</td>
<td>12</td>
<td>7</td>
<td>15.1</td>
<td>100.09</td>
<td>Rain</td>
<td></td>
<td></td>
</tr>
<tr>
<td>2010-01-01 10:00</td>
<td>2010</td>
<td>1</td>
<td>1</td>
<td>10:00</td>
<td>7.1</td>
<td>6.9</td>
<td>99</td>
<td>14</td>
<td>9</td>
<td>19.3</td>
<td>100.06</td>
<td>Rain</td>
<td></td>
<td></td>
</tr>
<tr>
<td>2010-01-01 11:00</td>
<td>2010</td>
<td>1</td>
<td>1</td>
<td>11:00</td>
<td>8.3</td>
<td>7.7</td>
<td>96</td>
<td>16</td>
<td>11</td>
<td>12.9</td>
<td>100.12</td>
<td>Rain</td>
<td></td>
<td></td>
</tr>
<tr>
<td>2010-01-01 12:00</td>
<td>2010</td>
<td>1</td>
<td>1</td>
<td>12:00</td>
<td>9.3</td>
<td>8.2</td>
<td>93</td>
<td>13</td>
<td>13</td>
<td>24.1</td>
<td>100.02</td>
<td>Rain</td>
<td></td>
<td></td>
</tr>
<tr>
<td>2010-01-01 13:00</td>
<td>2010</td>
<td>1</td>
<td>1</td>
<td>13:00</td>
<td>9.5</td>
<td>8.2</td>
<td>92</td>
<td>14</td>
<td>15</td>
<td>12.9</td>
<td>100.07</td>
<td>Rain</td>
<td></td>
<td></td>
</tr>
<tr>
<td>2010-01-01 14:00</td>
<td>2010</td>
<td>1</td>
<td>1</td>
<td>14:00</td>
<td>9.2</td>
<td>8.1</td>
<td>93</td>
<td>17</td>
<td>20</td>
<td>16.1</td>
<td>100.1</td>
<td>Rain</td>
<td></td>
<td></td>
</tr>
<tr>
<td>2010-01-01 15:00</td>
<td>2010</td>
<td>1</td>
<td>1</td>
<td>15:00</td>
<td>8.8</td>
<td>7.9</td>
<td>94</td>
<td>17</td>
<td>20</td>
<td>4.8</td>
<td>100.19</td>
<td>Moderate Rain,Fog</td>
<td></td>
<td></td>
</tr>
<tr>
<td>2010-01-01 16:00</td>
<td>2010</td>
<td>1</td>
<td>1</td>
<td>16:00</td>
<td>8.2</td>
<td>7.7</td>
<td>97</td>
<td>14</td>
<td>15</td>
<td>3.2</td>
<td>100.24</td>
<td>Rain,Fog</td>
<td></td>
<td></td>
</tr>
<tr>
<td>2010-01-01 17:00</td>
<td>2010</td>
<td>1</td>
<td>1</td>
<td>17:00</td>
<td>8.6</td>
<td>7.7</td>
<td>94</td>
<td>18</td>
<td>20</td>
<td>6.4</td>
<td>100.3</td>
<td>Rain,Fog</td>
<td></td>
<td></td>
</tr>
<tr>
<td>2010-01-01 18:00</td>
<td>2010</td>
<td>1</td>
<td>1</td>
<td>18:00</td>
<td>9.1</td>
<td>7.2</td>
<td>88</td>
<td>21</td>
<td>17</td>
<td>6.4</td>
<td>100.43</td>
<td>Rain,Fog</td>
<td></td>
<td></td>
</tr>
<tr>
<td>2010-01-01 19:00</td>
<td>2010</td>
<td>1</td>
<td>1</td>
<td>19:00</td>
<td>8.7</td>
<td>7.7</td>
<td>93</td>
<td>14</td>
<td>9</td>
<td>4.8</td>
<td>100.57</td>
<td>Rain,Fog</td>
<td></td>
<td></td>
</tr>
<tr>
<td>2010-01-01 20:00</td>
<td>2010</td>
<td>1</td>
<td>1</td>
<td>20:00</td>
<td>8.8</td>
<td>7.5</td>
<td>92</td>
<td>23</td>
<td>11</td>
<td>16.1</td>
<td>100.71</td>
<td>Rain</td>
<td></td>
<td></td>
</tr>
<tr>
<td>2010-01-01 21:00</td>
<td>2010</td>
<td>1</td>
<td>1</td>
<td>21:00</td>
<td>7.9</td>
<td>7.4</td>
<td>97</td>
<td>24</td>
<td>4</td>
<td>19.3</td>
<td>100.84</td>
<td>Rain</td>
<td></td>
<td></td>
</tr>
<tr>
<td>2010-01-01 22:00</td>
<td>2010</td>
<td>1</td>
<td>1</td>
<td>22:00</td>
<td>7.9</td>
<td>6.9</td>
<td>93</td>
<td>20</td>
<td>15</td>
<td>24.1</td>
<td>100.94</td>
<td>Rain</td>
<td></td>
<td></td>
</tr>
<tr>
<td>2010-01-01 23:00</td>
<td>2010</td>
<td>1</td>
<td>1</td>
<td>23:00</td>
<td>9.1</td>
<td>6.7</td>
<td>85</td>
<td>23</td>
<td>15</td>
<td>32.2</td>
<td>101.03</td>
<td>Rain</td>
<td></td>
<td></td>
</tr>
<tr>
<td>2010-01-02 0:00</td>
<td>2010</td>
<td>1</td>
<td>2</td>
<td>0:00</td>
<td>9.1</td>
<td>6.7</td>
<td>85</td>
<td>24</td>
<td>13</td>
<td>24.1</td>
<td>101.18</td>
<td>Rain</td>
<td></td>
<td></td>
</tr>
<tr>
<td>2010-01-02 1:00</td>
<td>2010</td>
<td>1</td>
<td>2</td>
<td>1:00</td>
<td>9</td>
<td>6.2</td>
<td>83</td>
<td>24</td>
<td>20</td>
<td>24.1</td>
<td>101.23</td>
<td>Rain</td>
<td></td>
<td></td>
</tr>
<tr>
<td>2010-01-02 2:00</td>
<td>2010</td>
<td>1</td>
<td>2</td>
<td>2:00</td>
<td>8.9</td>
<td>5.7</td>
<td>80</td>
<td>23</td>
<td>22</td>
<td>24.1</td>
<td>101.35</td>
<td>Mostly Cloudy</td>
<td></td>
<td></td>
</tr>
<tr>
<td>2010-01-02 3:00</td>
<td>2010</td>
<td>1</td>
<td>2</td>
<td>3:00</td>
<td>8.8</td>
<td>5.6</td>
<td>80</td>
<td>22</td>
<td>22</td>
<td>32.2</td>
<td>101.43</td>
<td>Mostly Cloudy</td>
<td></td>
<td></td>
</tr>
<tr>
<td>2010-01-02 4:00</td>
<td>2010</td>
<td>1</td>
<td>2</td>
<td>4:00</td>
<td>8.8</td>
<td>5.3</td>
<td>79</td>
<td>24</td>
<td>7</td>
<td>32.2</td>
<td>101.51</td>
<td>Cloudy</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

...bit of a pain?....
another method: **Octave**

- programming language primarily intended for numerical computations
- mostly compatible with Matlab
- free open source
- originally released 1988
Tools to facilitate batch downloading of historical station data from Environment Canada

as of Nov21
% Automated download of EC station data -- Octave-friendly version
% Using instructions found here: ftp://client_climateftp.tor.ec.gc.ca/Pub/Get_More_Data_Plus_De_donnees/Suggestions_on_installing_Octave.m
% Station inventory (to identify station ID numbers): ftp://tor.ec.gc.ca/Pub/Get_More_Data_Plus_De_donnees/Station%20Inventory%20EN.csv
% For example, a url that will work (for testing purposes) is: 'http://climate.weather.gc.ca/climate_data/bulk_data_e.html?format=csv&station_id'...

% Created by JJB, November, 2017
% Tested and verified using Octave 4.0.0 on Windows 10 and Ubuntu 16.04 LTS
%
% ## Configuring and running in Octave (for full instructions: https://github.com/jasonbrodeur/EC_Wx_tools):
% - In the command window, change to the EC_Wx_tools-master directory
% - e.g. cd('D:\Yustuff\EC_Wx_tools-master') (Replace D:\Yustuff with your actual directory
% - Open the get_EC_Wx_octave.m function in the Octave editor
% - modify the value of save_dir to match your working directory
% - e.g. in the example provided above, you would change it to 'D:\Yustuff\EC_Wx_tools-master\Data' (note that the trailing slash is important)
% - edit the values for start_year and end_year to match your requirements
% - edit the list provided in station_ids, to match the IDs of the stations you want to download from (there is a link)ftp://tor.ec.gc.ca
% - be sure that each id number is separated by a semicolon (this will make an m x 1 column vector)
% - If you want to compile the data into a single (long) csv file for all years at a given site, keep compile_flag=1; If you want to download individual files for each year, keep compile_flag=0. Once you've done this, you should be able to click 'Run'. It should then work its way through the sites, downloading monthly files to a
% folder that you specify.

url_base = 'http://climate.weather.gc.ca/climate_data/bulk_data_e.html'; % Don't change

% Editable parameters
save_dir = 'D:\Local\EC_Wx\Data\'; % Location to save data files.
if isdir(save_dir) == 0
    mkdir(save_dir);
end

timeframe = 1; % 1 = hourly; 2 = daily; 3 = monthly
station_ids = [27600; 31688]; % The station ID numbers
start_year = 2016;
end_year = 2017;
compile_flag = 1; % If set to 0, script will download separate files to disk; If set to 1, will compile data for all years for a single sta
% Automated download of EC station data -- Octave-friendly version
% Using instructions found here:
ftp://client_climate@ftp.tor.ec.gc.ca/Pub/Get_More_Data_Plus_de_donnees/Suggestions_on_Installing_Cygwin_and_running_the_command_line_to_download_data.docx
% Station inventory (to identify Station ID numbers): ftp://ftp.tor.ec.gc.ca/Pub/Get_More_Data_Plus_de_donnees/Station%20Inventory%20EN.csv
% For example, a url that will work (for testing purposes) is: 'http://climate.weather.gc.ca/climate_data/bulk_data_e.html?format=csv&stationID=17955&year=2018&month=1&day=14&timeframe=2&submit= Download+Data'

% Created by JJB, November, 2017
% Tested and verified using Octave 4.0.0 on Windows 10 and Ubuntu 16.04 LTS

% # Configuring and running in Octave (for full instructions: https://github.com/jasonbrodeur/EC_Wx_tools):
% In the command window, change to the EC_Wx_tools-master directory
% ---- e.g. cd('D:\MyStuff\EC_Wx_tools-master') (Replace D:\MyStuff with your actual directory)
% Open the get_EC_Wx_octave.m function in the Octave editor
% Modify the value of save_dir to match your working directory
% ---- e.g. In the example provided above, you would change it to 'D:\MyStuff\EC_Wx_tools-master\Data\' (note that the trailing slash is important and necessary)
% Edit the values for start_year and end_year to match your requirements
% Edit the list provided in station_ids, to match the IDs of the stations you want to download from (here is a link (ftp://ftp.tor.ec.gc.ca/Pub/Get_More_Data_Plus_de_donnees/Station%20Inventory%20EN.csv) to the Station directory). Use values found in the 'Station ID' field.
% ---- be sure that each id number is separated by a semicolon (this will make an nx1 column vector)
% If you want to compile the data into a single (long) csv for all years at a given site, keep compile_flag=1; If you want to download individual monthly files, change its value to 0;
% Once you've done this, you should be able to click 'Run'. It should then work its way through the sites, downloading monthly files to a file called 'tmp.csv', reopening that file and appending it to the master list, which will be saved at the end.

url_base = 'http://climate.weather.gc.ca/climate_data/bulk_data_e.html'; % Don't change

% Editable parameters
save_dir = 'D:\local\EC_Wx\Data\'; % Location to save data files.
if nargin(save_dir)==0  
 mkdir(save_dir);
end

timeframe = 1; % 1 - hourly; 2 - daily; 3 - monthly
station_ids = [27690; 31688]; % The station ID numbers
start_year = 2015;
end_year = 2017;
compile_flag = 1; % If set to 0, script will download separate files to disk; If set to 1, will compile data for all years for a single station.

% Try to automatically cd into the master directory (assuring the

change what you want here... or....
# Install

<table>
<thead>
<tr>
<th>Source</th>
<th>GNU/Linux</th>
<th>macOS</th>
<th>BSD</th>
<th>Windows</th>
</tr>
</thead>
</table>

The latest stable version is GNU Octave 4.2.1:

- `octave-4.2.1-w32-installer.exe (~ 170 MB)` [signature]
- `octave-4.2.1-w64-installer.exe (~ 184 MB)` [signature]
- `octave-4.2.1-w32.zip (~ 280 MB)` [signature]
- `octave-4.2.1-w64.zip (~ 378 MB)` [signature]

All Windows binaries with corresponding source code can be downloaded from [https://ftp.gnu.org/gnu/octave/windows/](https://ftp.gnu.org/gnu/octave/windows/).
paste code into Octave Editor pane

...change script here...
save_dir = 'C:\Data\EnvrnCan\ECanOctave'; % Location to save data files.

timeframe = 2; % 1 = hourly; 2 = daily; 3 = monthly
station_ids = [27600; 31688]; % The station ID numbers
start_year = 2015;
end_year = 2016;
compile_flag = 1; % If set to 0, script will download separate files to disk; if set to 1, will compile data for all years for a single station.

filename = [save_dir 'tmp.csv']; % saved file name
<table>
<thead>
<tr>
<th>Year</th>
<th>Month</th>
<th>Day</th>
<th>Max Temp (°C)</th>
<th>Min Temp (°C)</th>
<th>Mean Temp (°C)</th>
<th>Heat Deg Days (°C)</th>
<th>Cool Deg Days (°C)</th>
<th>Total Rain (mm)</th>
<th>Note</th>
</tr>
</thead>
<tbody>
<tr>
<td>2016</td>
<td>12</td>
<td>21</td>
<td>1.3</td>
<td>-4.6</td>
<td>-1.7</td>
<td>19.7</td>
<td>0 NaN</td>
<td>N</td>
<td></td>
</tr>
<tr>
<td>2016</td>
<td>12</td>
<td>22</td>
<td>4.1</td>
<td>0.8</td>
<td>2.5</td>
<td>15.5</td>
<td>0 NaN</td>
<td>N</td>
<td></td>
</tr>
<tr>
<td>2016</td>
<td>12</td>
<td>23</td>
<td>3.7</td>
<td>-1.1</td>
<td>1.3</td>
<td>16.7</td>
<td>0 NaN</td>
<td>N</td>
<td></td>
</tr>
<tr>
<td>2016</td>
<td>12</td>
<td>24</td>
<td>4.7</td>
<td>1.7</td>
<td>3.2</td>
<td>14.8</td>
<td>0 NaN</td>
<td>N</td>
<td></td>
</tr>
<tr>
<td>2016</td>
<td>12</td>
<td>25</td>
<td>3.8</td>
<td>-2.2</td>
<td>0.8</td>
<td>17.2</td>
<td>0 NaN</td>
<td>N</td>
<td></td>
</tr>
<tr>
<td>2016</td>
<td>12</td>
<td>26</td>
<td>10.1</td>
<td>-2</td>
<td>4.1</td>
<td>13.9</td>
<td>0 NaN</td>
<td>N</td>
<td></td>
</tr>
<tr>
<td>2016</td>
<td>12</td>
<td>27</td>
<td>8.5</td>
<td>-2</td>
<td>3.3</td>
<td>14.7</td>
<td>0 NaN</td>
<td>N</td>
<td></td>
</tr>
<tr>
<td>2016</td>
<td>12</td>
<td>28</td>
<td>1.5</td>
<td>-2.4</td>
<td>-0.5</td>
<td>18.5</td>
<td>0 NaN</td>
<td>N</td>
<td></td>
</tr>
<tr>
<td>2016</td>
<td>12</td>
<td>29</td>
<td>2.8</td>
<td>-0.1</td>
<td>1.4</td>
<td>16.6</td>
<td>0 NaN</td>
<td>N</td>
<td></td>
</tr>
<tr>
<td>2016</td>
<td>12</td>
<td>30</td>
<td>0.9</td>
<td>-2.3</td>
<td>-0.7</td>
<td>18.7</td>
<td>0 NaN</td>
<td>N</td>
<td></td>
</tr>
<tr>
<td>2016</td>
<td>12</td>
<td>31</td>
<td>NaN</td>
<td>-2.3</td>
<td>NaN</td>
<td>NaN</td>
<td>NaN</td>
<td>N</td>
<td></td>
</tr>
<tr>
<td>2017</td>
<td>1</td>
<td>1</td>
<td>3.1</td>
<td>-0.5</td>
<td>1.3</td>
<td>16.7</td>
<td>0 NaN</td>
<td>N</td>
<td></td>
</tr>
<tr>
<td>2017</td>
<td>1</td>
<td>2</td>
<td>5.5</td>
<td>0.7</td>
<td>3.1</td>
<td>14.9</td>
<td>0 NaN</td>
<td>N</td>
<td></td>
</tr>
<tr>
<td>2017</td>
<td>1</td>
<td>3</td>
<td>4.8</td>
<td>2.4</td>
<td>3.6</td>
<td>14.4</td>
<td>0 NaN</td>
<td>N</td>
<td></td>
</tr>
<tr>
<td>2017</td>
<td>1</td>
<td>4</td>
<td>3.8</td>
<td>-7.6</td>
<td>-1.9</td>
<td>19.9</td>
<td>0 NaN</td>
<td>N</td>
<td></td>
</tr>
<tr>
<td>2017</td>
<td>1</td>
<td>5</td>
<td>-5.3</td>
<td>-10.3</td>
<td>-7.8</td>
<td>25.8</td>
<td>0 NaN</td>
<td>N</td>
<td></td>
</tr>
<tr>
<td>2017</td>
<td>1</td>
<td>6</td>
<td>-5.9</td>
<td>-11.8</td>
<td>-8.9</td>
<td>26.9</td>
<td>0 NaN</td>
<td>N</td>
<td></td>
</tr>
<tr>
<td>2017</td>
<td>1</td>
<td>7</td>
<td>-6.4</td>
<td>-11.8</td>
<td>-9.1</td>
<td>27.1</td>
<td>0 NaN</td>
<td>N</td>
<td></td>
</tr>
<tr>
<td>2017</td>
<td>1</td>
<td>8</td>
<td>-8.2</td>
<td>-11.1</td>
<td>-9.7</td>
<td>27.7</td>
<td>0 NaN</td>
<td>N</td>
<td></td>
</tr>
</tbody>
</table>

separate years in one .csv...

(but has not worked for me yet...)
EnvironmentCanada weather stations as of 2012 (not all active)
Environment Canada weather stations as of 2012 (not all active)
other climate data possibilities... using GIS

WorldClim

WorldClim is a set of global climate layers (gridded climate data) with a spatial resolution of about 1 km². These data can be used for mapping and spatial modeling.

The new Version 2.0 is now available (current climate only --- more coming soon)

The old version is Version 1.4.
For this version you can get data for past, current and future climates.

Version2: gridded raster climate data of minimum, mean, maximum temperature and precipitation for 1970-2000 averages by months Jan-Dec

data interpolated to 1 square kilometre grid
WorldClim.org raster data & EnvironmentCanada weather stations
WorldClim raster data
average August temperature 1970-2000

Identify Results

Feature: wc2.0_30s_tavg_08
Value: wc2.0_30s_tavg_08
Band 1
(Derived)
Value: 13.8
DayMet much more complicated...

...but possible...

DayMet V3

Data Access

Direct
Daymet Data Sets
THREDDS Data Server

Tools
Single Pixel Extraction
Daymet Tile Selection

Cite Daymet
How to cite Daymet data

Daymet

What’s new in Version 3

Archived and distributed through the ORNL DAAC, the Daymet data set provides gridded estimates of daily weather parameters for North America, including daily continuous surfaces of minimum and maximum temperature, precipitation occurrence and amount, humidity, shortwave radiation, snow water equivalent, and daylight length. The daily time step, 1 km x 1 km spatial resolution, and North American spatial extent of the data set makes it a unique and valuable contribution to scientific, research, and educational communities. The literature shows that Daymet data have been broadly applied to fields including hydrology, terrestrial vegetation growth models, carbon cycle science, and regional to large scale climate change analysis.

Access to the Daymet data set is available through a variety of tools and formats allowing a rich resource of daily surface meteorology.

Daymet data are available for 1980 through the latest full calendar year and includes the United States, Mexico, Canada, Hawaii, and Puerto Rico.
Summary

- Python 2.7 script
  - ‘simplest’ script but need to strip out station information when downloading multiple months, years
  - long term support??

- Cygwin
  - separate .csv’s need to be combined / merged

- Octave
  - complex script
  - currently for me not compiling multiple months, years in one .csv

- WorldClim: 20 year averages; raster data for GIS

- DayMet: NetCDF files need to be converted to raster for GIS