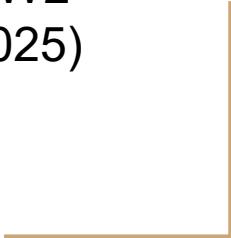




Programming, Problem Solving, and Algorithms

CPSC 203, 2024 W2
(January – April 2025)
Ian M. Mitchell
Lecture 08



Announcements

- Course web page: <https://ubc-cs.github.io/cpsc203/>
 - Weeks 1 – 3 are updated, week 4 is underway.
- Starting this week: Pre-lecture videos
 - I dropped the new videos for today's lecture (watch them for next Tuesday).
- Assessments:
 - Lab 3 (data classes) done.
 - POTW 3 & 4 due next Sunday
 - Test 2 in CBTF today – Monday
 - Book for test 3 now!
- Tech stack: If at first you don't succeed... ask for help.

CPSC 203 weekly schedule

	Monday	Tuesday	Wednesday	Thursday	Friday	Saturday	Sunday
Lectures	videos	12:30 – 14:00	videos	12:30 – 14:00			
Labs	Noon – 13:30 16:00 – 17:00	14:30 – 17:00 17:00 – 20:00		Due @noon	Look over the lab		
POTW	Five problems, five days						Due @noon
Tests in CBTF (~bi-weekly)	Last day to take the test			Slots available to take the test			CBTF closed

- The three projects are multi-week assessments with their own schedules

Today's Plan...

1. Announcements!
2. Knitting data classes
3. Pandas

Pandas and data frames

```
import pandas as pd
```

Imports the pandas library. We will almost always use an abbreviation...

Instead of saying `pandas.read_csv('file.csv')`

we can say `pd.read_csv('file.csv')`

This function returns a DataFrame containing the data from `file.csv`

CSV files

To implement `df = pd.read_csv('file.csv')`

`file.csv` must have field names in row 1, and data beginning in row 2.

bill_week.csv

saved ▼

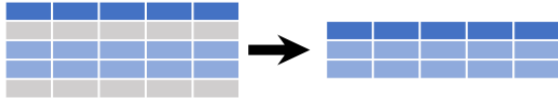
```
1 |,week,title,artist,rank,last_week,peak_pos,weeks_on_chart
2 |0,2019-09-21,Truth Hurts,Lizzo,1,1,1,19
3 |1,2019-09-21,Senorita,Shawn Mendes & Camila Cabello,2,2,1,12
4 |2,2019-09-21,Goodbyes,Post Malone Featuring Young Thug,3,10,3,10
5 |3,2019-09-21,Circles,Post Malone,4,7,4,2
6 |4,2019-09-21,Bad Guy,Billie Eilish,5,3,1,24
7 |5,2019-09-21,Ransom,Lil Tecca,6,4,4,15
8 |6,2019-09-21,No Guidance,Chris Brown Featuring Drake,7,6,6,14
```

Processing Data in CPSC 103

- Typical CPSC 103 workflow for CSV files:
 - Identify or create a data type for the information in each column of interest
 - Create Compound data type to store a row
 - Create List[Compound] data type to store the full data set
 - Write read and parse functions to get the data from the file
 - Write analyze and visualize functions to process the data and produce output
- Pros and Cons
 - Pro: You think carefully about the information and the data representation before coding
 - Pro: You practice your HtDD and HtDF processes and get correct implementations
 - Con: Your code is very specific to a single data set
 - Con: You can easily manipulate rows, but not columns
 - Con: New functions are required for even simple filter or map operations
 - Con: Your code is not particularly efficient

Pandas example: Selecting Rows

Subset Observations (Rows)



`df[df.Length > 7]`

Extract rows that meet logical criteria.

`df.drop_duplicates()`

Remove duplicate rows (only considers columns).

`df.head(n)`

Select first n rows.

`df.tail(n)`

Select last n rows.

`df.sample(frac=0.5)`

Randomly select fraction of rows.

`df.sample(n=10)`

Randomly select n rows.

`df.iloc[10:20]`

Select rows by position.

`df.nlargest(n, 'value')`

Select and order top n entries.

`df.nsmallest(n, 'value')`

Select and order bottom n entries.

```
df.nlargest(10, 'last_week')
```

Returns top 10 hits from last week.

```
df[ df['weeks_on_chart'] > 10 ]
```

Returns all songs that have been on the charts for more than 10 weeks.

Logic in Python (and pandas)

<	Less than	<code>!=</code>	Not equal to
>	Greater than	<code>df.column.isin(values)</code>	Group membership
==	Equals	<code>pd.isnull(obj)</code>	Is NaN
<=	Less than or equals	<code>pd.notnull(obj)</code>	Is not NaN
>=	Greater than or equals	<code>&, , ~, ^, df.any(), df.all()</code>	Logical and, or, not, xor, any, all

Pandas Example: Adding a column

```
df['gradient'] = df['last_week'] - df['rank']
```

Adds a column to the DataFrame containing the difference for every row.

So then we can easily perform the map:

```
df[ df['gradient'] > 10 ]
```

Returns all songs that have moved more than 10 spaces in the last week..

Pandas: A sharp blade

- We will learn to use Pandas through our next exploration topic: the Billboard Hot 100
 - Also: The visualization package Matplotlib and how to scrape data from the web
- Compared to our CPSC 103 process, Pandas
 - Allows us to quickly¹ perform complex data manipulations
 - Allows us to quickly¹ generate bugs that are difficult to spot and even harder to remove

¹ In terms of both coding time and execution speed.