



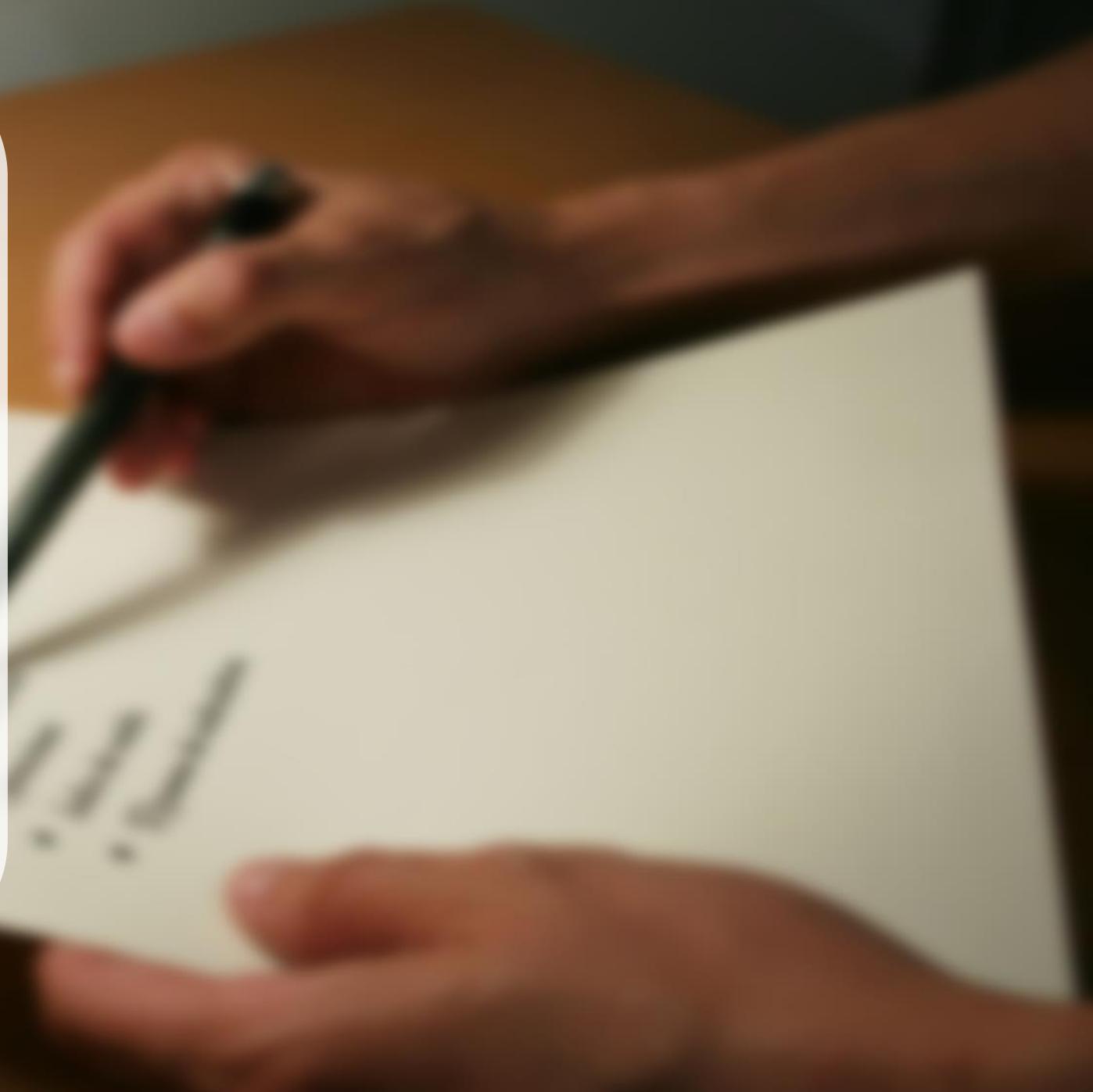
Working with Messy Data in OpenRefine

Nicholas Field, Reference Specialist

Kelly Schultz, Data Visualization Librarian

Agenda

- Learning Objectives
- Introduction to OpenRefine
- Demonstrations and hands-on activities
- Wrap-up: Map & Data Library Services



Learning objectives

- Participants will be able to use OpenRefine to:
 - ✓ Manipulate both textual and numeric data
 - ✓ Create new data, transform and reshape datasets, and search and filter data in a variety of ways. GREL expressions and regular expressions will be introduced
 - ✓ Use APIs to bring data into OpenRefine
- Participants will be aware of Map & Data Library services and know where to go for more help



*A free, open source, powerful tool
for working with messy data*

[Home](#)

[Download](#)

[Documentation](#)

[Community](#)

[Post archive](#)

[OpenRefine News:
Spring 2016](#)

[OpenRefine News:
December 2015](#)

[OpenRefine News:
November 2015](#)

Welcome!

OpenRefine (formerly Google Refine) is a powerful tool for working with messy data: cleaning it; transforming it from one format into another; and extending it with web services and external data.

Please note that since October 2nd, 2012, Google is not actively supporting this project, which has now been rebranded to OpenRefine. Project development, documentation and promotion is now fully supported by volunteers. Find out more about the [history of OpenRefine](#) and how you can help the community.

2017 OpenRefine User Survey

It's been a while since our last user survey (see the result from the [2014 edition](#)), we would like to know who you are, how you use OpenRefine and what your expectations are. So here it is the 2017 edition of the OpenRefine user survey! Thank you for sharing it with your friends, coworker, and communities!

[Take the survey](#)

Using OpenRefine - The Book

Why OpenRefine?



vs



Installing OpenRefine

- OpenRefine is installed locally on your computer, even though it uses a web browser as the user interface
- A copy of your data files are saved locally on your computer





Demonstrations/Hands-on practice

A photograph of a man jogging on a paved path. He is wearing a yellow tank top and black shorts. The background shows a white fence and green grass under a clear sky.

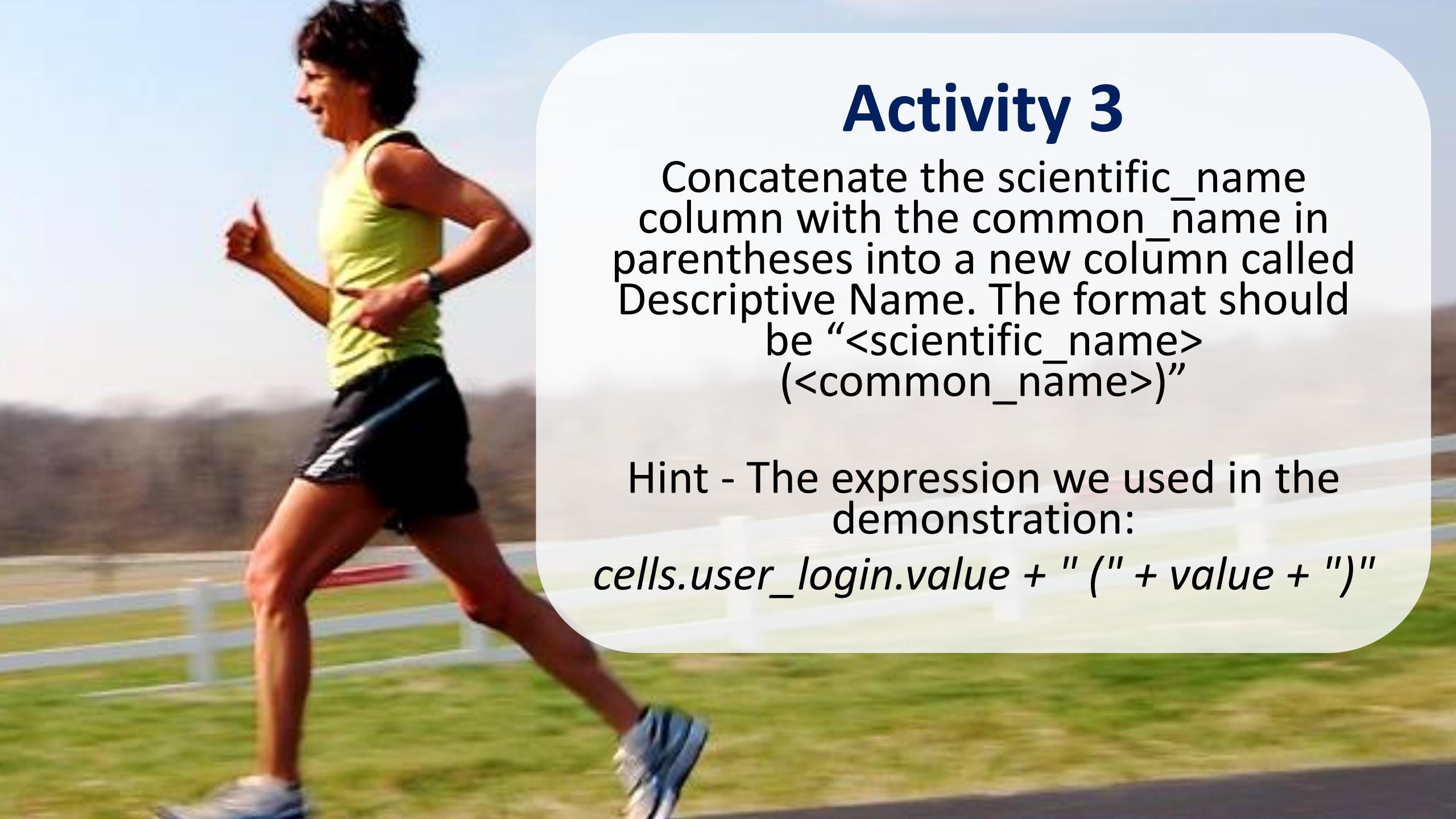
Activity 1

Add another text facet on the household expenditures column. What did the average household in British Columbia spend on pet food in 2016?

Activity 2

Try playing around with the cluster methods and see if you can clean up the species_guess column's values more.



A photograph of a man jogging on a paved path through a park. He is wearing a yellow tank top and black shorts. The background shows green grass and trees.

Activity 3

Concatenate the scientific_name column with the common_name in parentheses into a new column called Descriptive Name. The format should be “<scientific_name> (<common_name>)”

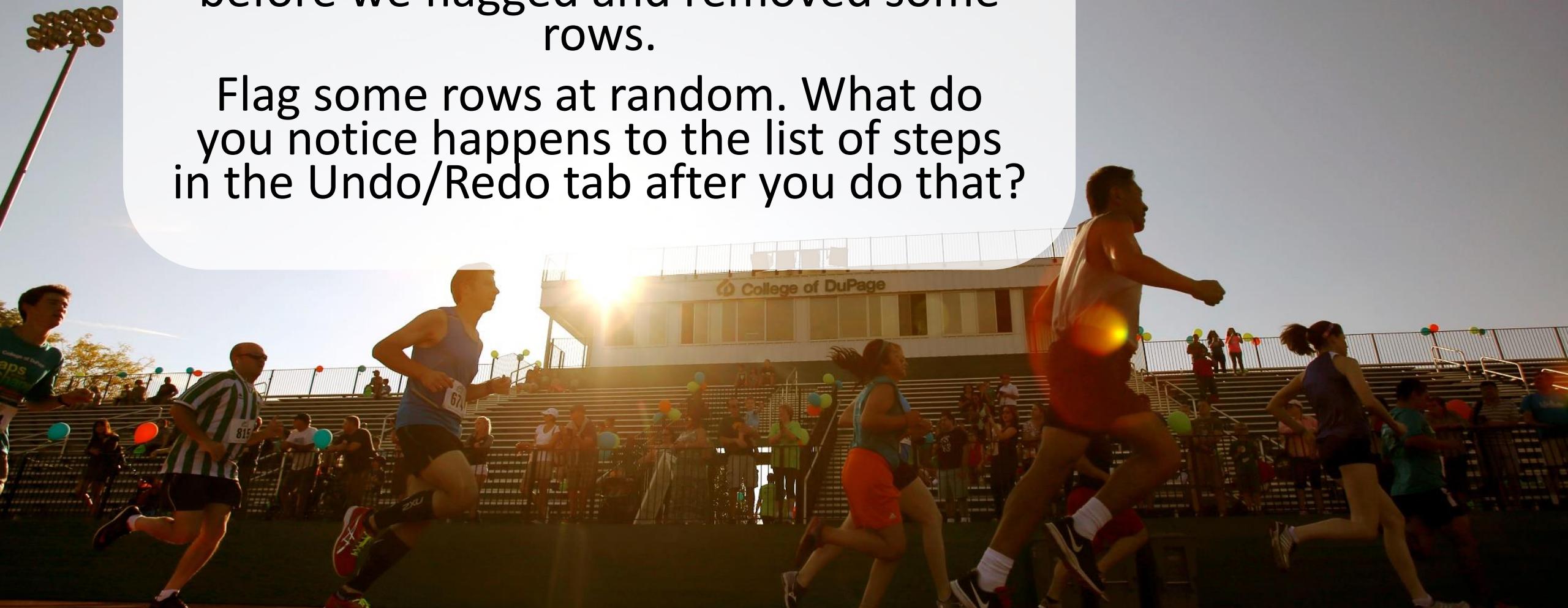
Hint - The expression we used in the demonstration:

cells.user_login.value + " (" + value + ")"

Activity 4

Use Undo/Redo to go back to the step “Reorder columns” to see the dataset before we flagged and removed some rows.

Flag some rows at random. What do you notice happens to the list of steps in the Undo/Redo tab after you do that?

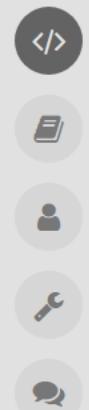


A photograph of two white ceramic cups filled with coffee, each featuring intricate latte art on the surface. The cups are placed on a dark brown wooden table with visible planks. In the foreground, the bottom half of another cup and saucer is visible, along with a spoon resting on the saucer. The lighting is warm and focused on the cups.

Break

What is a “regular expression”?

- A sequence of characters that define a search pattern
- Used to search for 'matches' within text (strings)
- Can also be called:
 - Regex
 - Regexp
 - Pattern
 - Rational expression



SAVE & SHARE

save regex ctrl+s

FLAVOR

</> **pcre (php)** ✓

</> javascript

</> python

</> golang

TOOLS

code generator

regex debugger

REGULAR EXPRESSION

/ insert your regular expression here / g

TEST STRING

insert your test string here

SWITCH TO UNIT TESTS

Learning to write them

- To practice, we'll use a web-based tool to see results in real time

regex101.com

- Later, we'll use them in OpenRefine to do something useful

EXPLANATION

An explanation of yo

MATCH INFORMATION

Detailed match info

QUICK REFERENCE

Search reference

- all tokens
- common tokens
- general tokens
- anchors

Notebook 1

Site	Date	Measurement
Baker	2009-11-17	1223.0
Baker	2010-06-24	1122.7
Baker	2009-05-24	2819.0
Baker	2010-08-25	2971.6
Baker	2011-01-05	1410.0
Baker	2010-09-04	4671.6

Notebook 2

Site	Date	Measurement
Davison	May 23, 2010	1724.7
Pertwee	May 24, 2010	2103.8
Davison	June 19, 2010	1731.9
Davison	July 6, 2010	2010.7
Pertwee	Aug 4 2010	1731.3
Davison	Apr 22, 2010	2122.2
Pertwee	Sept 3, 2010	3981.0

Sample data

Accessing the sample data

- Open text file "sample_data.txt"
- Copy the contents
- Paste it into regex101.com, in the "TEST STRING" box

Type your regular expression here

Add mode modifiers here

The screenshot shows the regex101.com web application. On the left, there's a sidebar with icons for saving and sharing, and dropdown menus for 'FLAVOR' (set to 'pcre (php)') and 'TOOLS' (showing 'code generator' and 'regex debugger'). The main area has tabs for 'REGULAR EXPRESSION' (active), 'TEST STRING', and 'SWITCH TO UNIT TESTS'. The 'REGULAR EXPRESSION' tab contains a text input field with the placeholder '/ insert your regular expression here' and a dropdown menu with 'no match' and '/gi'. The 'TEST STRING' tab displays a table of data with columns 'Site', 'Date', and 'Measurement'. The data includes rows for Baker from 2009-11-17 to 2010-09-04, Davison from May 23 to Sept 3, 2010, and Pertwee from May 24 to Sept 3, 2010. A red arrow points from the 'Type your regular expression here' text to the input field. Another red arrow points from the 'Add mode modifiers here' text to the dropdown menu. To the right, there are two sections: 'EXPLANATION' (with the sub-instruction 'An explanation of your regex will be automatically generated as you type.') and 'MATCH INFORMATION' (with the sub-instruction 'Detailed match information will be displayed here automatically.'). Red arrows point from the 'Information about your expression' and 'Matches made by your expression' text to these respective sections.

Site	Date	Measurement
Baker	2009-11-17	1223.0
Baker	2010-06-24	1122.7
Baker	2009-05-24	2819.0
Baker	2010-08-25	2971.6
Baker	2011-01-05	1410.0
Baker	2010-09-04	4671.6
Davison	May 23, 2010	1724.7
Pertwee	May 24, 2010	2103.8
Davison	June 19, 2010	1731.9
Davison	July 6, 2010	2010.7
Pertwee	Aug 4, 2010	1731.3
Davison	Apr 22, 2011	2122.2
Pertwee	Sept 3, 2010	3981.0

Sample data

Information
about your
expression

Matches made
by your
expression



Mode modifiers

no match EXPLANATION

REGEX FLAGS

global ✓
Don't return after first match

multi line
^ and \$ match start/end of line

insensitive ✓
Case insensitive match

extended
Ignore whitespace

eXtra
Disallow meaningless escapes

single line
Dot matches newline

unicode
Match with full unicode

Ungreedy
Make quantifiers lazy

Anchored
Anchor to start of pattern

Jchanged
Allow duplicate subpattern names

Dollar end only
\$ matches only end of pattern

your regex will be case sensitive.

ION

formation will be displayed.

Metacharacters & operators

There are certain characters that have special meaning in regular expressions, these are \ , | , () . [] * + ? { } ^ \$.

These are used in combination with text characters to construct regular expressions.

Operator	Description
\	Escape character - use when you need to include a metacharacter as a literal in a regular expression e.g. \.txt
	OR (alternation) e.g. wom[a e]n
()	Group e.g. (....)-(..)-(..) for a date
.	Match any single character e.g. wom.n
[abc]	Match any of a, b, or c. e.g. [btr]ent
[a-c]	Match any character between a and c. e.g. [a-z]ent
*	The preceding item will be matched zero or more times e.g. teen[a-z]*
+	The preceding item will be matched one or more times organi.+
?	The preceding item is optional and will be matched once at most, e.g. colo?r Used to make a quantifier 'lazy', e.g. .+?
{N}	The preceding item is matched exactly N times [0-9]{4}
{N,}	The preceding item is matched N or more times [0-9]{4,}
{N,M}	The preceding item is matched at least N times, but not more than M times [0-9]{4,6}
^	Anchor: matches only at the start of the string e.g. ^b When used within a character set, negates the set, i.e. matches all characters <i>not</i> in the set - e.g. [^abc]
\$	Anchor: matches only at the end of the string e.g. b\$
\b	Anchor: matches at "word boundary" (zero length position), i.e. transition from word to non-word characters. This allows you to perform "whole word only" searches e.g. \bword\b
\w	Shorthand character class: "word". Matches all the ASCII characters [A-Za-z0-9_]
\s	Shorthand character class: "whitespace". Matches non-word characters including space, tab, line break or form feed.

Metacharacters & operators

APIs (Application Programming Interface)

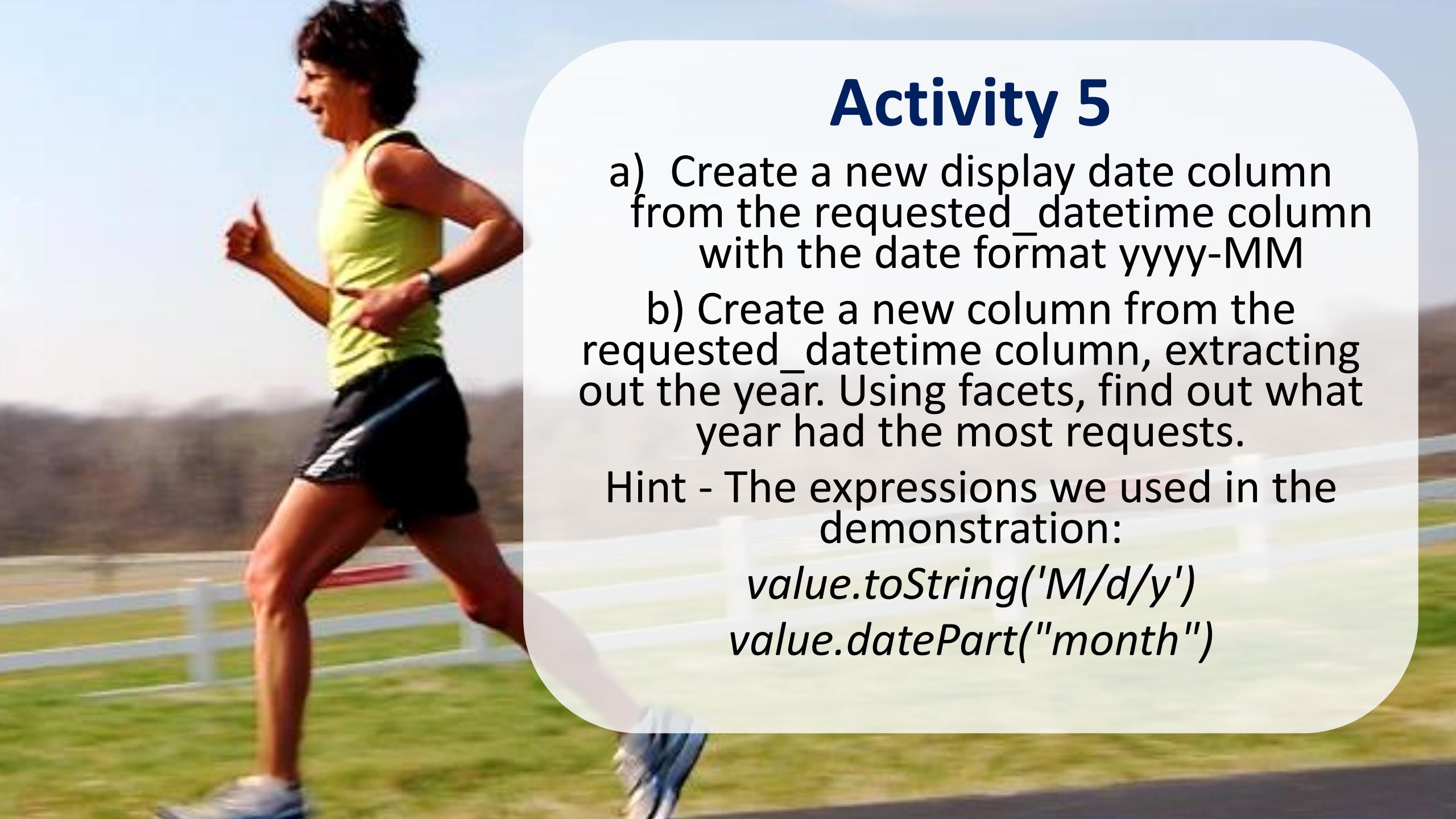
- Web APIs – construct a URL and specify the parameters you want to make a data request
- Example:

[https://secure.toronto.ca/webwizard/ws/requests.json?
service_code=CSROSC-14&jurisdiction_id=toronto.ca](https://secure.toronto.ca/webwizard/ws/requests.json?service_code=CSROSC-14&jurisdiction_id=toronto.ca)

JSON

```
[  
 {  
   "id": 2,  
   "name": "An ice sculpture",  
   "price": 12.50,  
   "tags": ["cold", "ice"],  
   "dimensions": {  
     "length": 7.0,  
     "width": 12.0,  
     "height": 9.5  
   },  
   "warehouseLocation": {  
     "latitude": -78.75,  
     "longitude": 20.4  
   }  
 },  
 ]
```

- Made up of key/value pairs
- Uses curly brackets and colons to structure the data and create the nested format
- Example:
“price”: 12.50

A photograph of a man jogging on a paved path. He is wearing a yellow tank top and black shorts. The background shows a grassy field and a clear sky.

Activity 5

- a) Create a new display date column from the requested_datetime column with the date format yyyy-MM
- b) Create a new column from the requested_datetime column, extracting out the year. Using facets, find out what year had the most requests.

Hint - The expressions we used in the demonstration:

value.toString('M/d/y')

value.datePart("month")

OpenRefine Reconciliation Services

- Look up values in a reconciliation database to find potential matches
- Use matched information to standardize dataset terms and add additional data

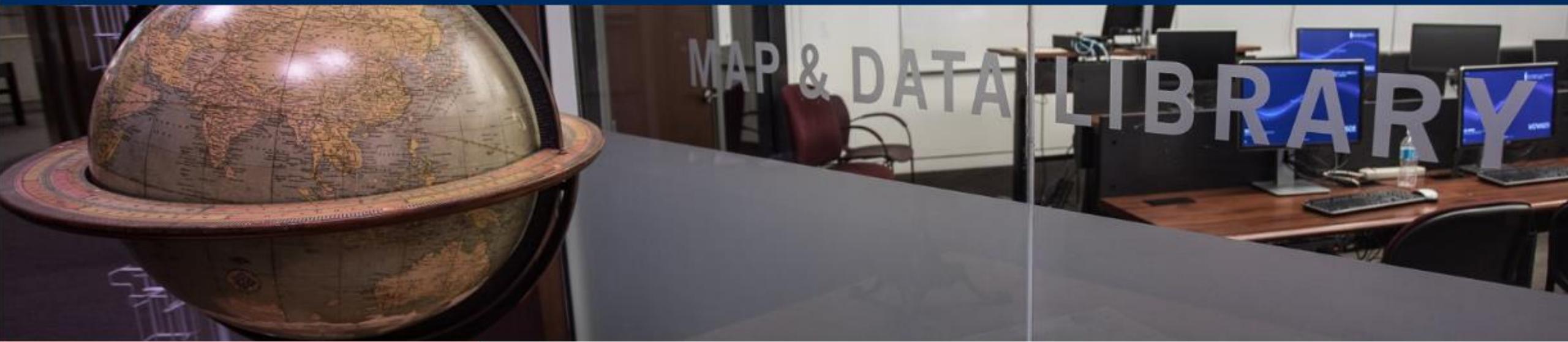
Class	Subject	Reconcile	Standardized Subject List	Department
GEO101	Geography and Planning		Geography	Geography & Planning
MTH101	Mathematics		Biology	Ecology & Evolutionary
ENG101	English		Sociology	Biology
MTH102	Math		English	Sociology
ANT101	Anthro		Math	English
SOC101	Sociology		Anthropology	Mathematics & Statistics
GEO102	Geography	Result		Anthropology

Class	Reconciled Subjects	Department
GEO101	Geography	Geography & Planning
MTH101	Math	Mathematics & Statistics
ENG101	English	English
MTH102	Math	Mathematics & Statistics
ANT101	Anthropology	Anthropology
SOC101	Sociology	Sociology
GEO102	Geography	Geography & Planning

Closing OpenRefine

- Switch to the command window open and press the **Ctrl key** and **C** together
- Wait until there's a message that says the shutdown is complete





About MDL

Our collection includes hundreds of geospatial and numeric datasets, over hundreds of thousands of maps, photographs, and more! We provide assistance finding maps and data and using GIS and statistical software.

Start your search **-> mdl.library.utoronto.ca**

Find maps, data, books, and library info



search by title Map and Data only

Geospatial data

[Scholars GeoPortal](#) | [Geospatial data](#) | [Remote sensing](#) | [Air photo](#)

Numeric data

[Microdata](#) | [Statistics](#) | [Census of Canada](#)

Maps and atlases

[Scanned maps](#) | [Fire insurance plans](#) | [Rare maps](#)

Contact us: **mdl@library.utoronto.ca**

Helpful resources

- OpenRefine documentation wiki:
<https://github.com/OpenRefine/OpenRefine/wiki/Documentation-For-Users>
- OpenRefine Tutorial from John Little (Duke University):
<https://libjohn.github.io/openrefine/index.html>
- Software Carpentry OpenRefine Workshop for Librarians: <https://librarycarpentry.github.io/lc-open-refine/>
- Cleaning Data with OpenRefine from the Programming Historian:
<https://programminghistorian.org/lessons/cleaning-data-with-openrefine>
- Fetching and Parsing Data from the Web with OpenRefine from the Programming Historian:
<https://programminghistorian.org/lessons/fetch-and-parse-data-with-openrefine>
- Regex Cheat Sheet: <http://www.rexegg.com/regex-quickstart.html>

Activity (for GPS Credit)

Email me (kelly.schultz@utoronto.ca) the spreadsheet and reflection paragraph within 1 week

Cleaned Dataset

1. Go through the activity handouts called:
AugmentingActivity1_PreparingtheData.pdf and
AugmentingActivity2_Reconciliation.pdf
2. Export the resulting cleaned dataset as an Excel file

Reflection – One Paragraph

1. What you thought were the most important things you learned today
2. Why are they important
3. How you will apply this knowledge in the future

A large, diverse crowd of people is shown from the waist up, all with their hands raised in the air. They are smiling and appear to be at a public event or workshop. The background is a dark, indoor setting.

Wrap-Up

- 1) Key lesson?**
 - 2) Anything unclear?**
 - 3) Useful workshop?**
- Why/Why not?**

Data sources

- Statistics Canada CANSIM Table 203-0021 - Survey of household spending (SHS), household spending, Canada, regions and provinces, annual (dollars):
<http://www5.statcan.gc.ca/cansim/a26?lang=eng&retrLang=eng&id=2030021&tabMode=dataTable&p1=1&p2=-1&srchLan=-1&pattern=household+spending>
- Wolfpack Citizen Science Challenge Spring 2017 - North Carolina State iNaturalist Observations:
<https://data.world/wcsc/spring-2017-challenge/workspace/file?datasetid=north-carolina-state-inaturalist-observations&filename=observations-18840.csv> (Must complete a free registration with data.world to access the data)
- The sample data is adapted from the Software Carpentry lesson: <http://v4.software-carpentry.org/regexp/patterns.html>
- 311 - Open311 API Calls for Service Requests: <https://www.toronto.ca/city-government/data-research-maps/open-data/open-data-catalogue/#e2634d40-0dbf-4d91-12fa-83f039307e93>
- Project Gutenberg (<http://www.gutenberg.org/>) top 32 English books, accessed via this API:
<https://github.com/garethbjohnson/gutendex>

Image credits

- **Slide 1:** *The Hand of Saruman.. or Amy*, flickr.com/Juhan Sonin, <https://flic.kr/p/nN8L1f>
- **Slide 2:** *Manos*, morguefile.com/xololounge, <http://mrg.bz/Yda9ls>
- **Slide 3:** *Drawing and coloring*, Freelmages.com/ Ove Tøpfer, <http://www.freeimages.com/photo/drawing-1313453>
- **Slide 5:** *R logo* by Hadley Wickham and others at RStudio. Licensed under the Creative Commons Attribution-Share Alike 4.0 International license via Wikimedia Commons - https://commons.wikimedia.org/wiki/File%3AR_logo.svg
- **Slide 6:** *This is what using the computers at work feels like*, flickr.com/david, <https://flic.kr/p/7fc4yi>
- **Slide 7:** *Starting line*, flickr.com/Jon Marshall, <https://flic.kr/p/p4zWb>
- **Slides 8, 10, & 22:** *run*, flickr.com/brett lohmeyer, <https://flic.kr/p/68oeCp>

Image credits

- **Slides 9 & 11:** *Fifth Annual Laps with the Chaps 5K Attracts Record Crowd 172*, flickr.com/COD Newsroom, <https://flic.kr/p/zcGYvk>
- **Slide 12:** *Coffee break*, flickr.com/Berit Watkin, <https://flic.kr/p/dzBrCi>
- **Slide 13:** *Pattern*, flickr.com/Ben Williams, <https://flic.kr/p/7C8ppd>
- **Slide 16:** *data.path Ryoji.Ikeda – 4*, flickr.com/r2hox, <https://flic.kr/p/gdMuhT>
- **Slide 18:** *Wave pattern*, flickr.com/inkelv1122, <https://flic.kr/p/6a9yCg>
- **Slide 19:** *Cobblestone pattern*, flickr.com/Chris Waits, <https://flic.kr/p/orhwp7>
- **Slide 20:** *Téléphone ancient*, flickr.com/Frédéric BISSON, <https://flic.kr/p/cFMG6E>
- **Slide 21:** *Building a product schema*. (n.d.). Retrieved July 11, 2017 from Wikipedia: <http://json-schema.org/example1.html>

Image credits

- **Slide 24:** *Sorry WE'RE CLOSED*, flickr.com/FraserElliot, <https://flic.kr/p/cSqUvU>
- **Slide 27:** *Drawing and coloring*, Freelmages.com/ Ove Tøpfer, <http://www.freeimages.com/photo/drawing-1313453>
- **Slide 28:** *audience wave*, flickr.com/Gavin Tapp, <https://flic.kr/p/aqvnet>; *Post-it sticker small yellow emtpy single left up - GIMP 2.8* by User:Mattes (creator), eyeknife (idea) - Own work. Licensed under GFDL via Wikimedia Commons - https://commons.wikimedia.org/wiki/File:Post-it_sticker_small_yellow_emtpy_single_left_up_-_GIMP_2.8.png#/media/File:Post-it_sticker_small_yellow_emtpy_single_left_up_-_GIMP_2.8.png