Research Design and *pandas*

Ivan Corneillet

Data Scientist

GENERAL ASSEMBLY

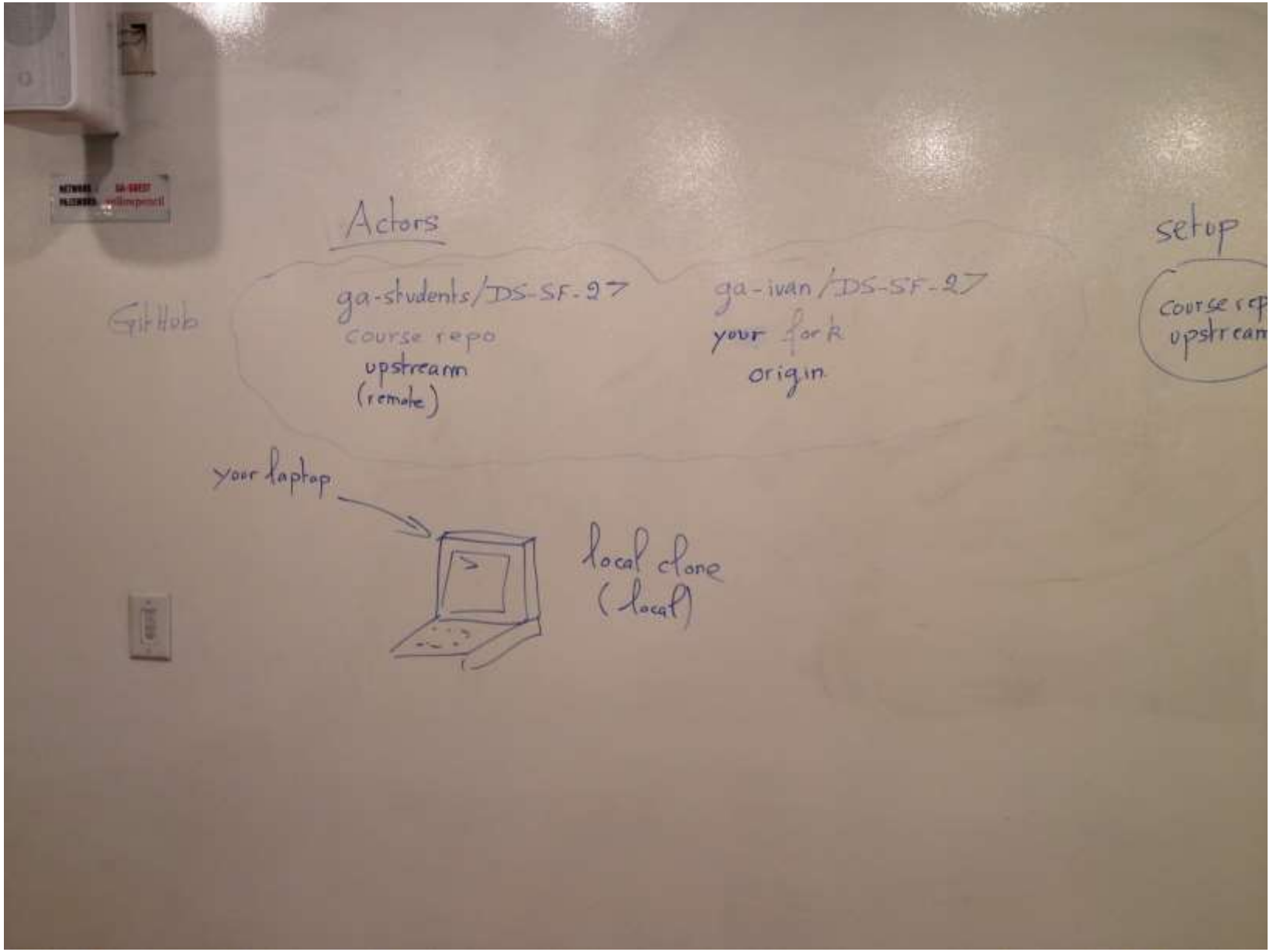
Learning Objectives

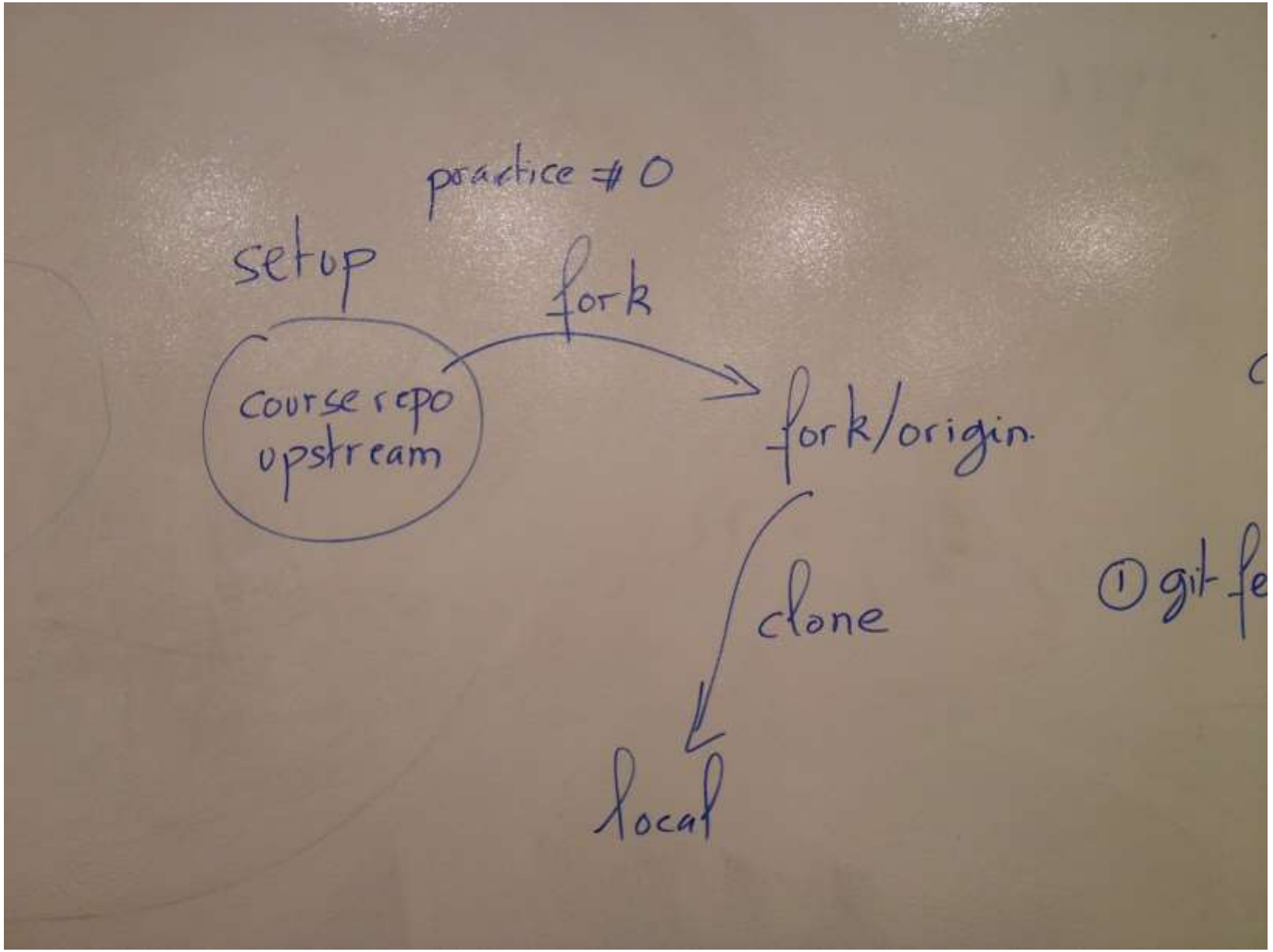
After this lesson, you should be able to:

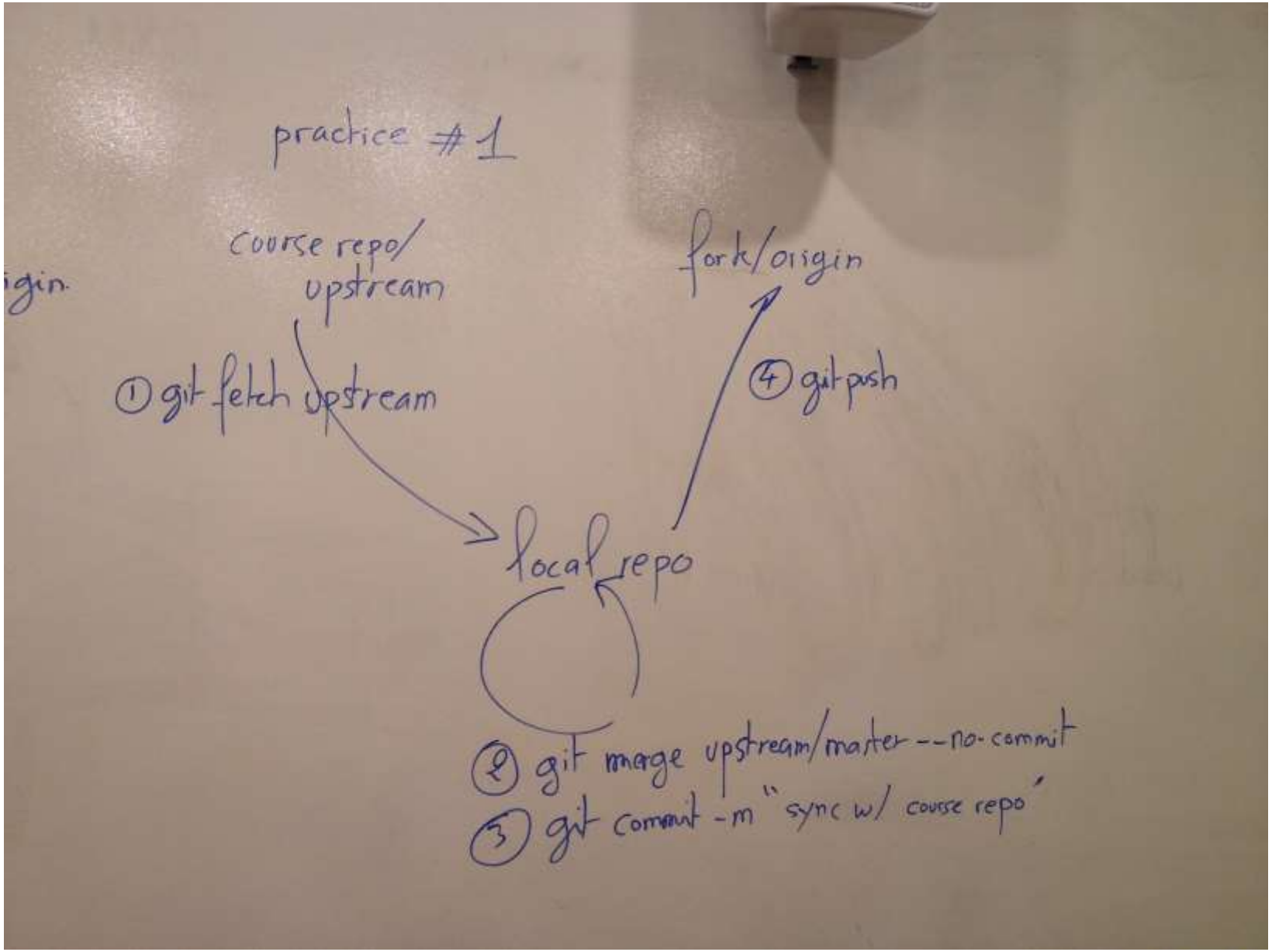
- Define a problem and types of data
- Identify dataset types
- Apply the data science workflow in the *pandas* context
- Write an Jupyter notebook to import, format, and clean data using the *pandas* library

Announcements and Exit Tickets

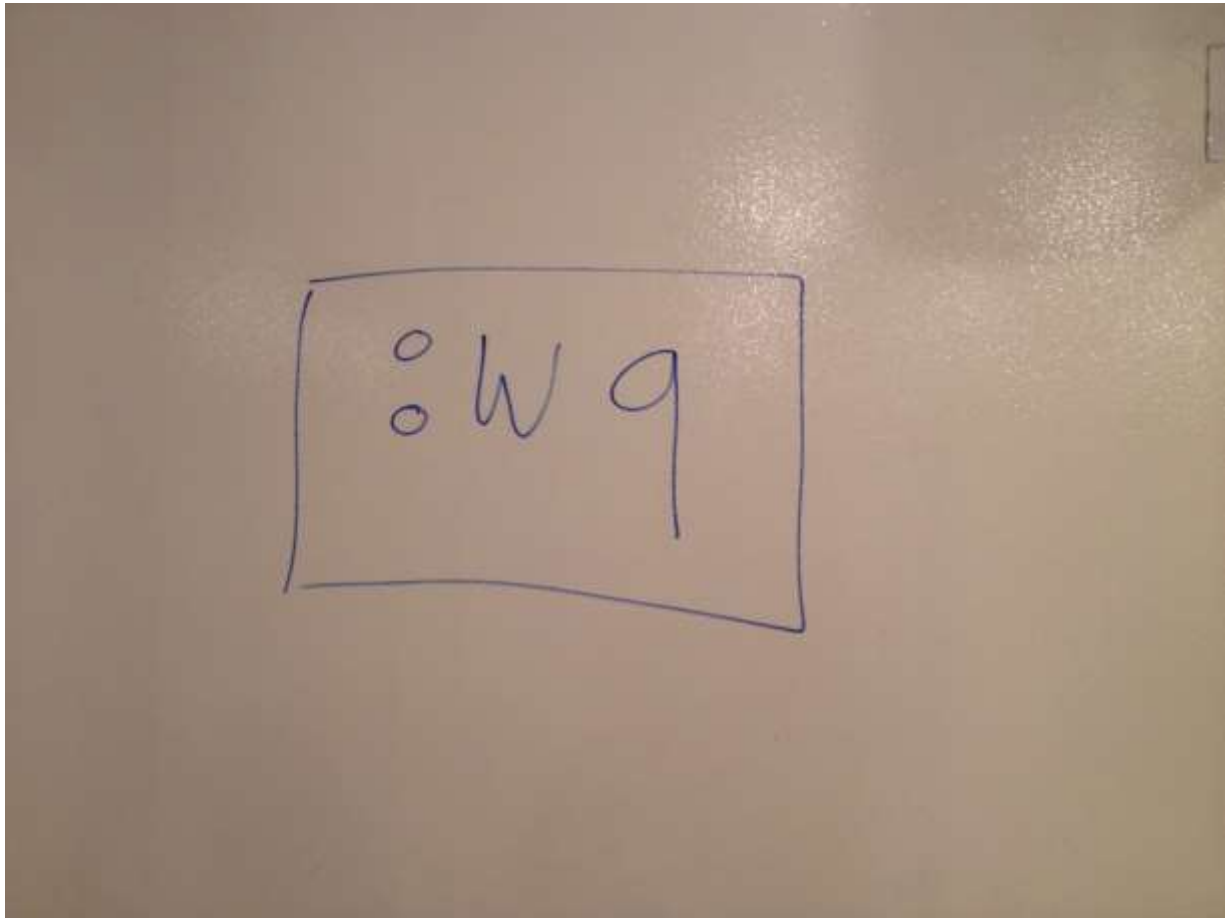
Review


Actors
GitHub
ga-students/DS-SF-27
course repo
upstream
(remote)
ga-ivan/DS-SF-27
your fork
origin
setup
course rep
upstream
your laptop
local clone
(local)


practice #0
setup
fork
course repo
upstream
fork/origin
clone
local

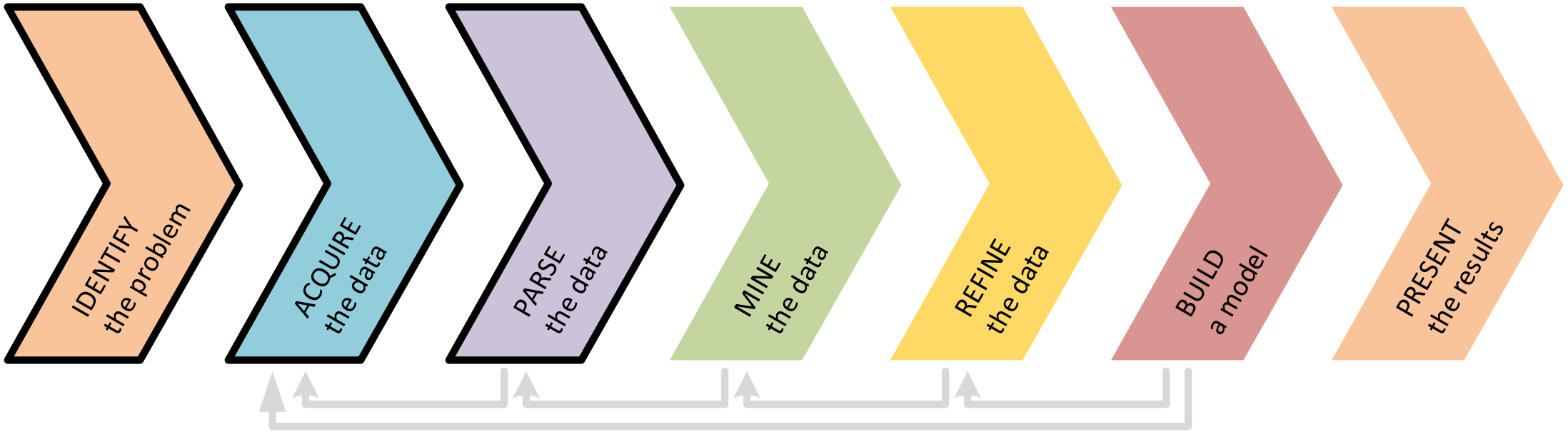

practice #1
course repo/
upstream
fork/origin
① git fetch upstream
④ git push
local repo
② git merge upstream/master --no-commit
③ git commit -m "sync w/ course repo"

Forgot the `--no-commit` flag and now "stuck" in *vi*?



Today

Today we'll focus on the first three (❶ IDENTIFY the problem, ❷ ACQUIRE the data, and ❸ PARSE the data)



Today, we are covering Research Design and introducing the *pandas* library

Research Design and Data Analysis	Research Design	Data Visualization in *pandas*	Statistics	Exploratory Data Analysis in *pandas*
Foundations of Modeling	Linear Regression	Classification Models	Evaluating Model Fit	Presenting Insights from Data Models
Data Science in the Real World	Decision Trees and Random Forests	Time Series Models	Natural Language Processing	Databases

Here's what's happening today:

- Announcements and Exit Tickets
- Review
- Wired's "The Rise of Artificial Intelligence and the End of Code" | Class Discussion
- ❶ Identify the Problem
  - The Why's and How's of a Good Question
  - The SMART Goals Framework for Data Science
- ❷ Acquire the Data
  - Data Types
  - Logistics of Acquiring Data
  - SF Housing Dataset
  - Wrangling Data
- ❸ Parse the Data
  - Documentation and Data Dictionaries
  - Codealong – Introduction to *pandas* and wrangling the SF Housing dataset
- Lab – Introduction to *pandas*
- Review
- Exit Tickets

Pre-Work

Pre-Work

- Complete your development environment setup; complete the onboarding pre-work and practice the different workflows that we will use in this course
- Look into the first unit project and start ideating about your final project's topic
- Read the two articles briefly mentioned in class, we will discuss then further in the next class:
  - Harvard Business Review | "Data Scientists: The Sexiest Job of the 21st Century" (2012) (https://hbr.org/2012/10/data-scientist-the-sexiest-job-of-the-21st-century/)
  - Wired | "The Rise of Artificial Intelligence and the End of Code" (2016) (http://www.wired.com/2016/05/the-end-of-code/)

Wired | “The Rise of Artificial Intelligence and the End of Code”

(http://www.wired.com/2016/05/the-end-of-code/)

Class Discussion

Wired's "The Rise of Artificial Intelligence and the End of Code" (2016) | Class Discussion

Behaviorism/Behavioral Psychology

- Brain as a black box
  - Stimulus and response, feedback and reinforcements
    - "ring bell, dog salivates"

Cognitive Psychology

- Brain more like a computer
  - Thoughts as programs
  - Absorb, process, and act upon information

Wired's "The Rise of Artificial Intelligence and the End of Code" (2016) | Class Discussion (cont.)

Machine Learning

- Humans *train* computers
  - Keep showing cats to a computer and eventually it will *learn* to recognize cats (https://www.wired.com/2012/06/google-x-neural-network/)
  - No symbols, no rules; instead an unparsable machine learning

Traditional Programming

- Humans *write code* (as explicit step-by-step-instructions) for computers to follow
  - Rule-based determinism
    - "Write enough rules and eventually, we'd create a system sophisticated enough to understand the world"
  - For years, Google Search relied mostly on these human-written rules (https://www.wired.com/2016/02/ai-is-changing-the-technology-behind-google-searches/)

Wired's "The Rise of Artificial Intelligence and the End of Code" (2016) | Class Discussion (cont.)

Age of Entanglement

- Outside-in view of how machine work
  - "Code doesn't just determine behavior, behavior also determine code"

Age of Enlightenment

- Inside-out view of how machine work
  - "First, we write the code, then the machine expresses it"

❶ IDENTIFY the Problem

❶ Identify the Problem

- Identify the Problem
  - Identify business/product objectives
  - Identify and hypothesize goals and criteria for success
  - Create a set of questions for identifying correct dataset

- The Why's and How's of a Good Question
- The SMART Goals Framework

By asking a good question and setting a clear aim:

- You set yourself up for success
  - “A problem well stated is half solved” – Charles Kettering
- You help other data scientists learn from and reproduce your work
  - You establish the basis for making your analysis reproducible
- You also help them expand on your work in the future

The SMART Goals Framework for Data Science

(https://en.wikipedia.org/wiki/SMART_criteria)

SPECIFIC	The dataset and key variables are clearly defined
MEASURABLE	The type of analysis and major assumptions are articulated
ATTAINABLE	The question you are asking is feasible for your dataset and is not likely to be biased
REPRODUCIBLE	Another person (or you in 6 months!) can read your state and understand exactly how your analysis is performed
TIME-BOUND	You clearly state the time period and population for which this analysis will pertain

Trends often change over time and vary by the population of source of your data. It is important to clearly define who/what you included in your analysis as well as the time period for the analysis

Activity | A SMART Goal for Your Final Project

EXERCISE

DIRECTIONS (10 minutes)

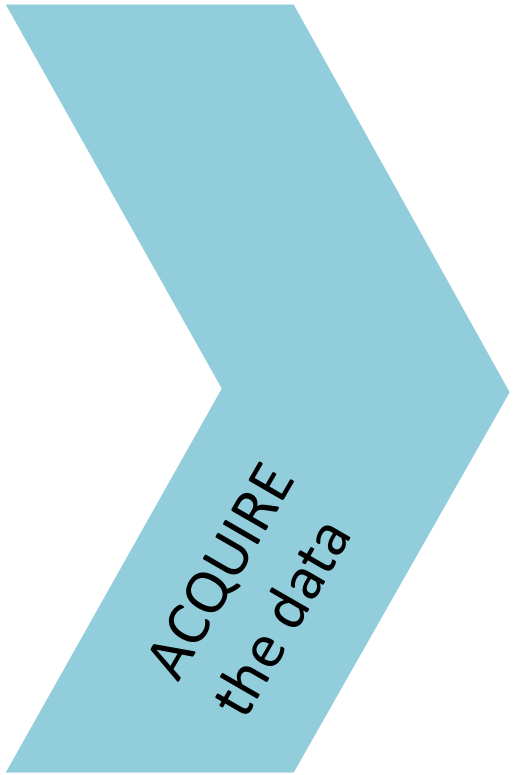
1. After the first class, you probably started brainstorming on an idea for your final project. If not, here's an opportunity!
2. If your idea is cool and interesting, that's great. But it is a SMART idea?
3. Assess your idea using the Data Science-tuned SMART Goal Framework
    a. If you have just a couple of gaps, how can you close them?
    b. On the other end, if you have too many and closing these gaps would be difficult, you might want to consider something else
4. After 5 minutes, share your idea and gaps in pairs and offer advise to each other, again using the SMART Framework (2.5 minutes each)

DELIVERABLE

Answers to the above questions

❷ ACQUIRE the Data

❷ Acquire the Data



- Acquire the Data
  - Identify the “right” dataset(s)
  - Import data and set up local or remote data structure
  - Determine most appropriate tools to work with data

❷ Acquire the Data (cont.)

- Questions to ask:
  - What type of data is it, cross-sectional or longitudinal?
  - How well was the data collected?
  - Is there much missing data?
  - Was the data collection instrument calibrated?
  - Is the dataset aggregated?
  - Do we need pre-aggregated data?

- Data Types
- Logistics of Acquiring Data
  - The SF Housing Dataset
- Wrangling Data

❷ ACQUIRE the Data

Data Types

Different data types have different limitations and strengths, e.g., certain types of analyses aren't possible with certain data types

Cross-Sectional Data

- Collect observations of many samples at the same point of time, or without regard to differences in time
- Issue: TEMPORALITY
  - No distinction between *exposure* and *outcome*

Longitudinal Data (i.e., Time Series)

- Tracks the same sample (e.g., individual, household, or establishment) at different points in time

Cross-Sectional and Longitudinal Data (cont.)

	✓	✗
Cross-sectional data	❑ Often population-based and therefore more generalizable ❑ Less expensive compared to other types of data collection methods	❑ Separation of cause and effect may be difficult or impossible
Longitudinal data	❑ Unambiguous temporal sequence; *exposure* precedes *outcome* ❑ Multiple outcomes can be measured	❑ Vulnerable to missing data ❑ Takes a long time to collect data ❑ More expense compared to other types of data collection methods

❷ ACQUIRE the Data

Logistics of Acquiring Data

Logistics of Acquiring Data

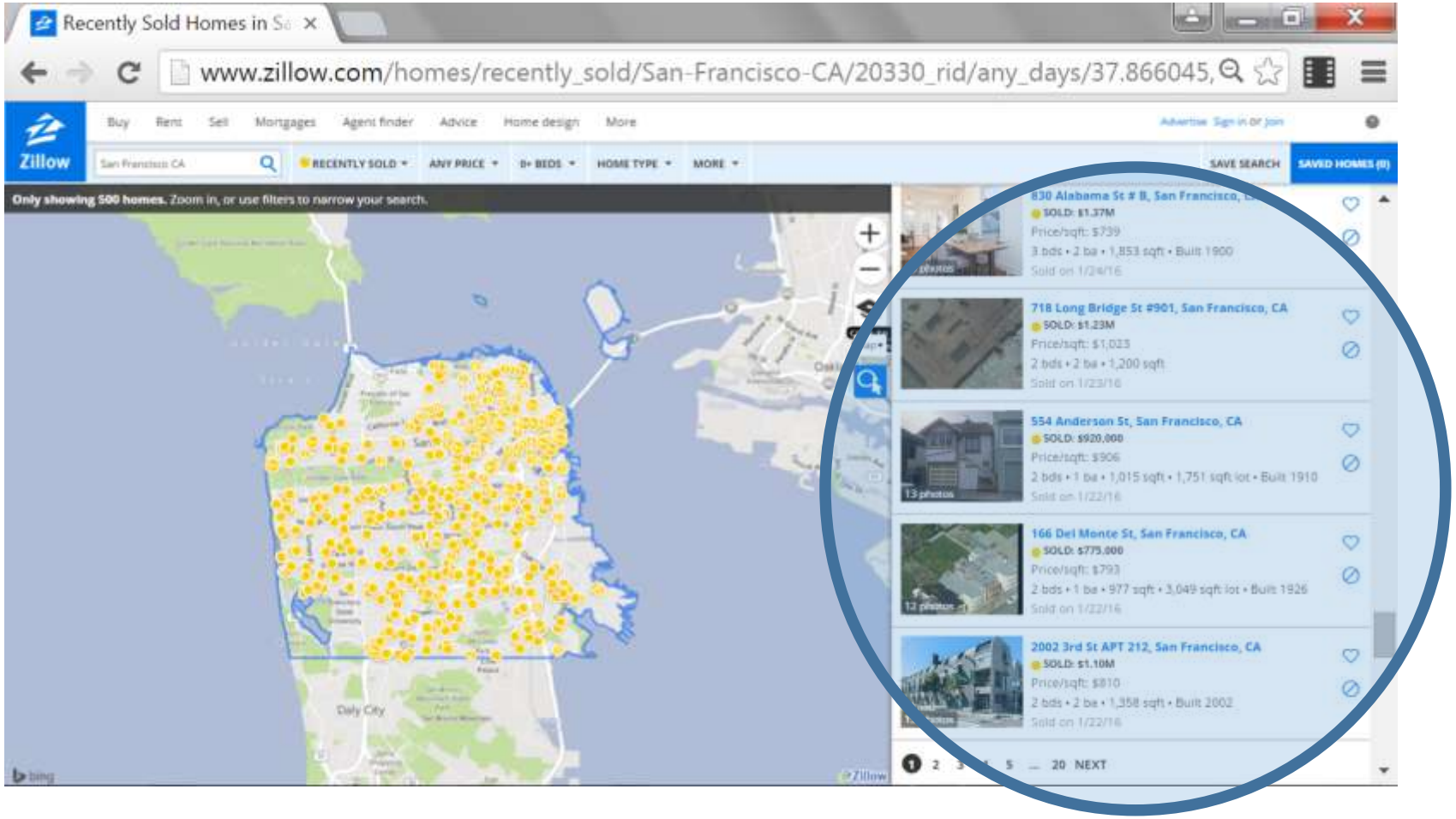
- Data can be acquired through a variety of sources
  - Web (e.g., HTML)
  - Databases
    - SQL (Structured Query Language)
    - NoSQL (“Not only SQL”)
- File Formats
  - CSV (Comma-Separated Values)
  - TSV/TXT (Tab-Separated Values)
  - JSON (JavaScript Object Notation)
  - XML (eXtensible Markup Language)

SF Housing Dataset: a dataset we will use throughout this course

CASE STUDY

- Recently Sold Homes (Source: Zillow)
  - 1,000 homes sold in San Francisco between 11/10/2015 and 2/12/2106

Raw data was scrapped from the Zillow website
(20 pages, each listing 50 homes for a total of 1,000 homes)



Raw data is Messy™…

```
<div class="property-info"
id="yui_3_18_1_1_1456167242885_71870"><strong
id="yui_3_18_1_1_1456167242885_71869"><dt class="property-address"
id="yui_3_18_1_1_1456167242885_71868"><a href="/homedetails/149-
Shipley-St-San-Francisco-CA-94107/15147894_zpid/" class="hdp-link
routable" title="149 Shipley St, San Francisco, CA Real Estate"
id="yui_3_18_1_1_1456167242885_71873">149 Shipley St, San
Francisco, CA</a></dt></strong><dt class="listing-type zsg-
content_collapsed" id="yui_3_18_1_1_1456167242885_71875"><span
class="zsg-icon-recently-sold type-icon"></span>Sold:
$1.18M</dt><dt class="zsg-fineprint"
id="yui_3_18_1_1_1456167242885_71877">Price/sqft: $1,116</dt><dt
class="property-data" id="yui_3_18_1_1_1456167242885_71880"><span
class="beds-baths-sqft">3 bds • 2 ba • 1,057 sqft</span><span
class="built-year" id="yui_3_18_1_1_1456167242885_71879"> • Built
1992</span></dt><dt class="sold-date zsg-fineprint"
id="yui_3_18_1_1_1456167242885_71975">Sold on 2/22/16</dt></div>
```

… and needs to be wrangled before we can apply any kind of machine learning algorithms

```
<div class="property-info"
id="yui_3_18_1_1_1456167242885_71870"><strong
id="yui_3_18_1_1_1456167242885_71869"><dt class="property-address"
id="yui_3_18_1_1_1456167242885_71868"><a href="/homedetails/149-
Shipley-St-San-Francisco-CA-94107/15147894_zpid/" class="hdp-link
routable" title="149 Shipley St, San Francisco, CA Real Estate"
id="yui_3_18_1_1_1456167242885_71873">149 Shipley St, San
Francisco, CA</a></dt></strong><dt class="listing-type zsg-
content_collapsed" id="yui_3_18_1_1_1456167242885_71875"><span
class="zsg-icon-recently-sold type-icon"></span>Sold:
$1.18M</dt><dt class="zsg-fineprint"
id="yui_3_18_1_1_1456167242885_71877">Price/sqft: $1,116</dt><dt
class="property-data" id="yui_3_18_1_1_1456167242885_71880"><span
class="beds-baths-sqft">3 bds • 2 ba • 1,057 sqft</span><span
class="built-year" id="yui_3_18_1_1_1456167242885_71879"> • Built
1992</span></dt><dt class="sold-date zsg-fineprint"
id="yui_3_18_1_1_1456167242885_71975">Sold on 2/22/16</dt></div>
```

❷ ACQUIRE the Data

Wrangling Data

Raw data is Messy™…

EXAMPLE

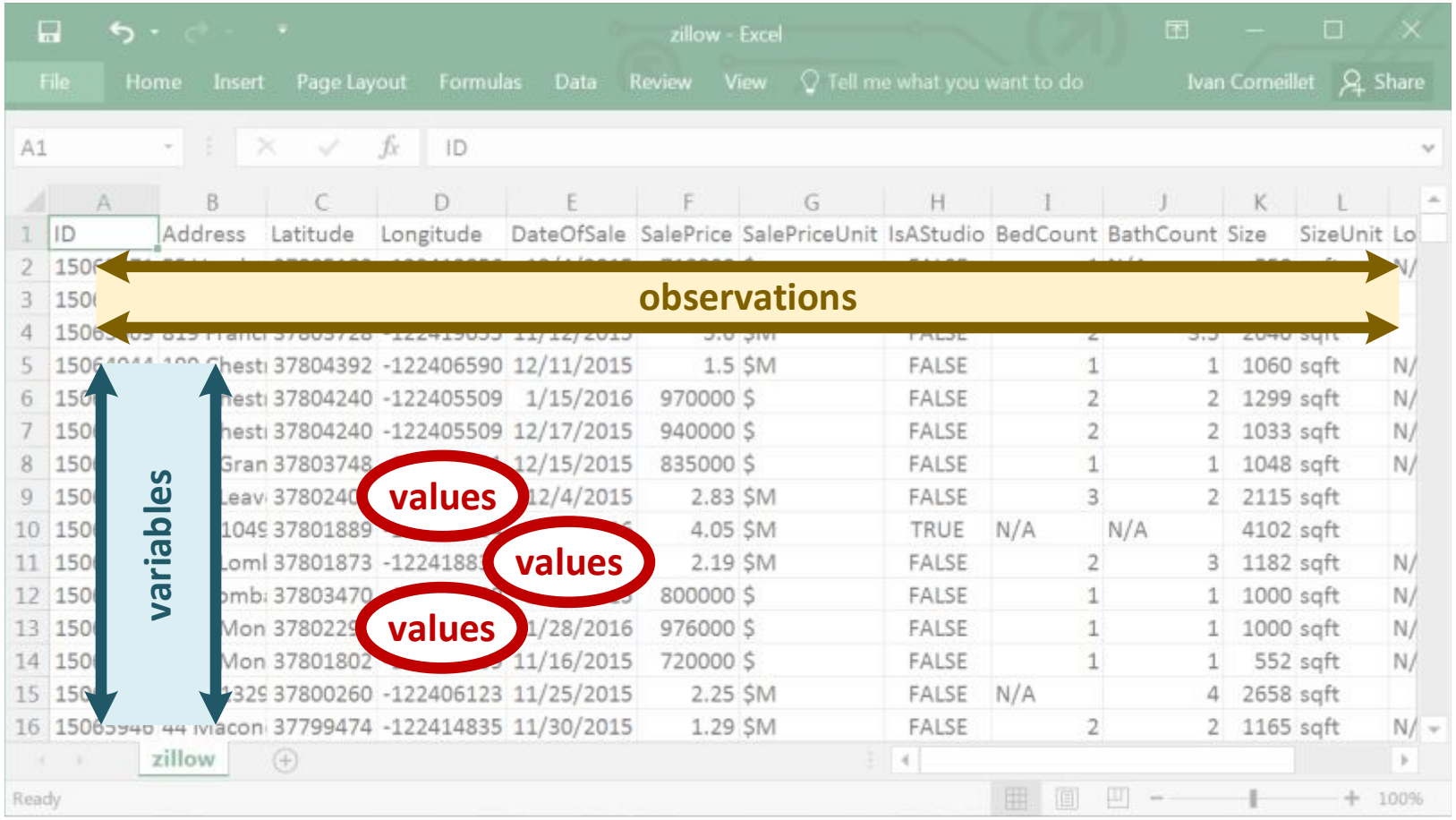
- Trouble tickets inspect and maintain manholes in New Year City
- "Service box," a common piece of infrastructure, had at least 38 variants, including `SB`, `S`, `S/B`, `S.B`, `S?B`, `S.B.`, `SBX`, `S/BX`, `SB/X`, `S/XB`, `/SBX`, `S.BX`, `S &BX`, `S?BX`, `S BX`, `S/B/X`, `S BOX`, `SVBX`, `SERV BX`, `SERV-BOX`, `SERV/BOX`, and `SERVICE BOX`

(Source: Big Data: A Revolution That Will Transform How We Live, Work, and Think)

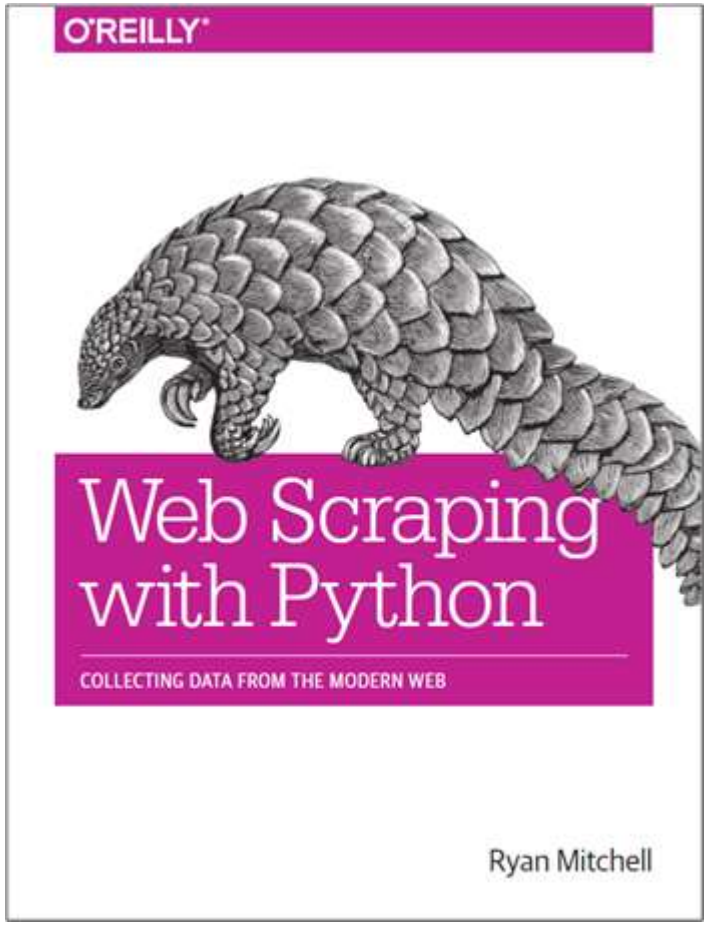
Wrangling Data

- Wrangling data is the most fruitful skill you can learn as a data scientist
  - It will save you hours of time and make your data much easier to visualize, manipulate, and model
- Many data science tools follow a set of conventions that makes one layout of tabular data much easier to work with than others. Your data will be easier to work with if you follow three rules:
  - Each observation is placed in its own row
  - Each variable in the dataset is placed in its own column
  - Each value is placed in its own cell

Wrangling Data (cont.)

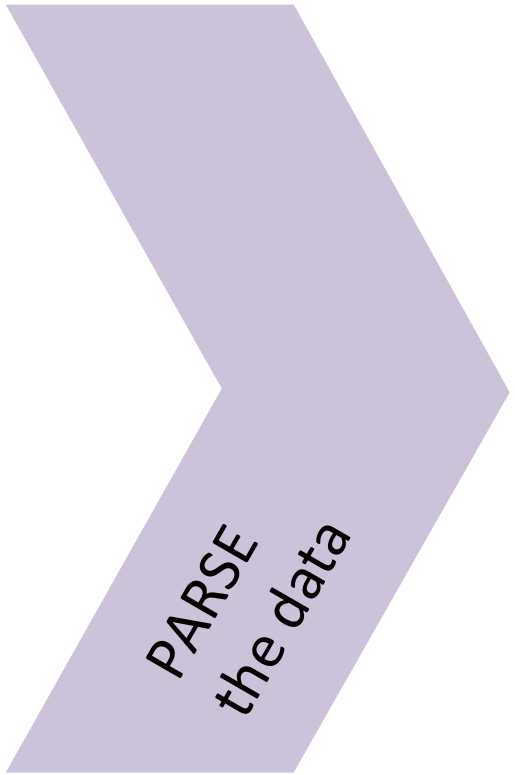


A good resource to get started with web scraping using Python *(optional; not required for the course)*



❸ PARSE the Data

❸ Parse the Data



- Parse the Data
  - Read any documentation provided with the data (session 2)
  - Perform exploratory data analysis (session 3)
  - Verify the quality of the data (sessions 2/3)

❷ Acquire the Data (cont.)

- You need to understand what you're working with
- To better understand your data
  - Create or review the data dictionary
  - Perform exploratory surface analysis
  - Describe data structure and information being collected
  - Explore variables and data types

- Documentation and Data Dictionary
- Introduction to *pandas* + codealong
- Codealong: Wrangling the SF Housing dataset (take 2) with *pandas*
- Lab

❸ PARSE the Data

Documentation and Data Dictionary

Documentation and Data Dictionary

- Data dictionaries
  - Help you judge the quality of the data
  - Also help understand how it's coded
    - Does "`gender = 1`" mean female or male?
    - Is the currency dollars or euros?
  - Help identify any requirements, assumptions, and constraints of the data
  - Make it easier to share data

Kaggle's Titanic Data Dictionary

EXAMPLE

```
VARIABLE DESCRIPTIONS:
survival        Survival
                (0 = No; 1 = Yes)
pclass          Passenger Class
                (1 = 1st; 2 = 2nd; 3 = 3rd)
name            Name
sex             Sex
age             Age
sibsp           Number of Siblings/Spouses Aboard
parch           Number of Parents/Children Aboard
ticket          Ticket Number
fare            Passenger Fare
cabin           Cabin
embarked        Port of Embarkation
                (C = Cherbourg; Q = Queenstown;
                 S = Southampton)

SPECIAL NOTES:
Pclass is a proxy for socio-economic status (SES)
 1st ~ Upper; 2nd ~ Middle; 3rd ~ Lower

Age is in Years; Fractional if Age less than One (1)
 If the Age is Estimated, it is in the form xx.5
```

```
With respect to the family relation variables (i.e.
sibsp and parch) some relations were ignored.  The
following are the definitions used for sibsp and
parch.

Sibling: Brother, Sister, Stepbrother, or
         Stepsister of Passenger Aboard Titanic
Spouse:  Husband or Wife of Passenger Aboard
         Titanic (Mistresses and Fiancés Ignored)
Parent:  Mother or Father of Passenger Aboard
         Titanic
Child:   Son, Daughter, Stepson, or Stepdaughter of
         Passenger Aboard Titanic

Other family relatives excluded from this study
include cousins, nephews/nieces, aunts/uncles, and
in-laws.  Some children travelled only with a nanny,
therefore parch=0 for them.  As well, some travelled
with very close friends or neighbors in a village,
however, the definitions do not support such
relations.
```

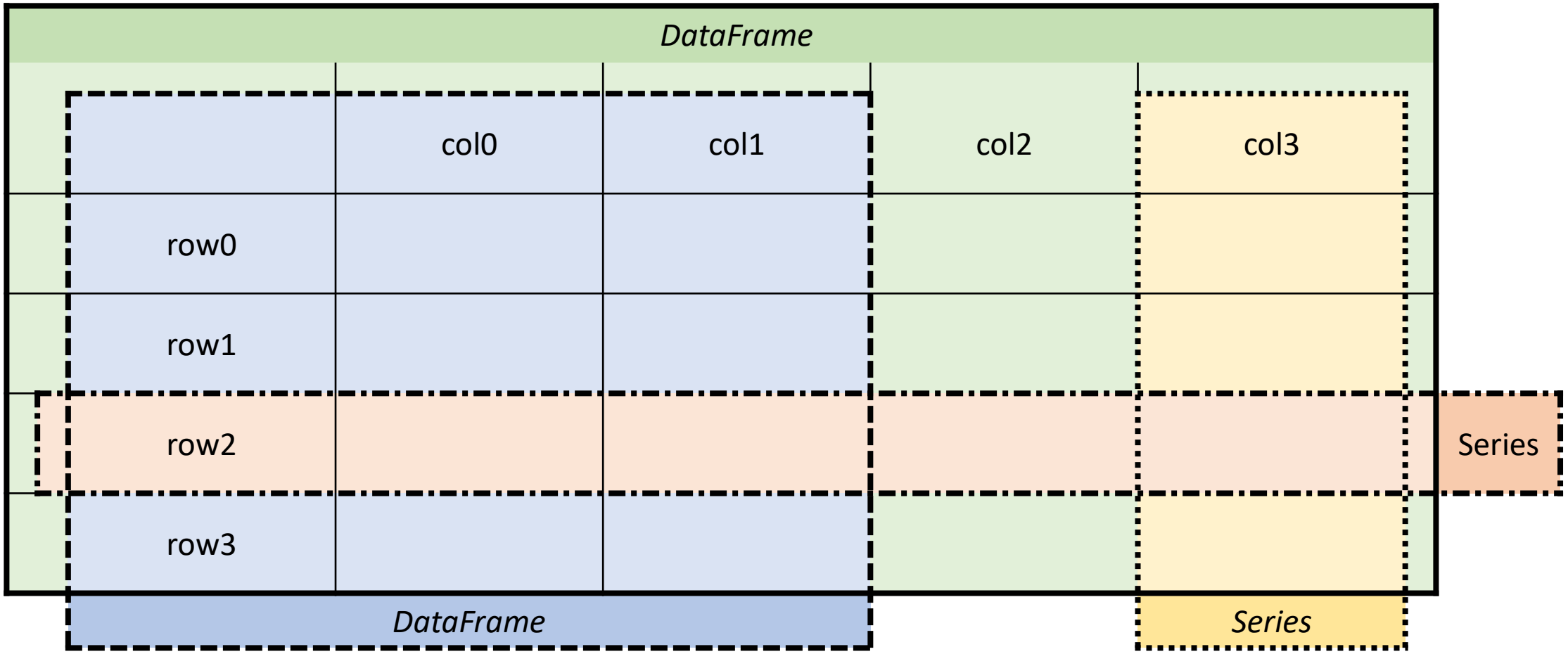
❸ PARSE the Data

Introduction to pandas

pandas is a Python library to manipulate and perform statistical and mathematical analysis on tabular and multidimensional datasets

- *pandas* provides the ability to *index*, *retrieve*, *tidy*, *reshape*, *combine*, *slice*, and perform various analyses on both single and multidimensional data
- It also includes loading and saving data from local and Internet-based resources
- We will use *pandas* to explore and manipulate the SF Housing dataset

pandas.DataFrame and *pandas.Series*



pandas data structures are very important. We will use them to model our tabular data as inputs to our machine learning algorithms

Feature Matrix X

DataFrame

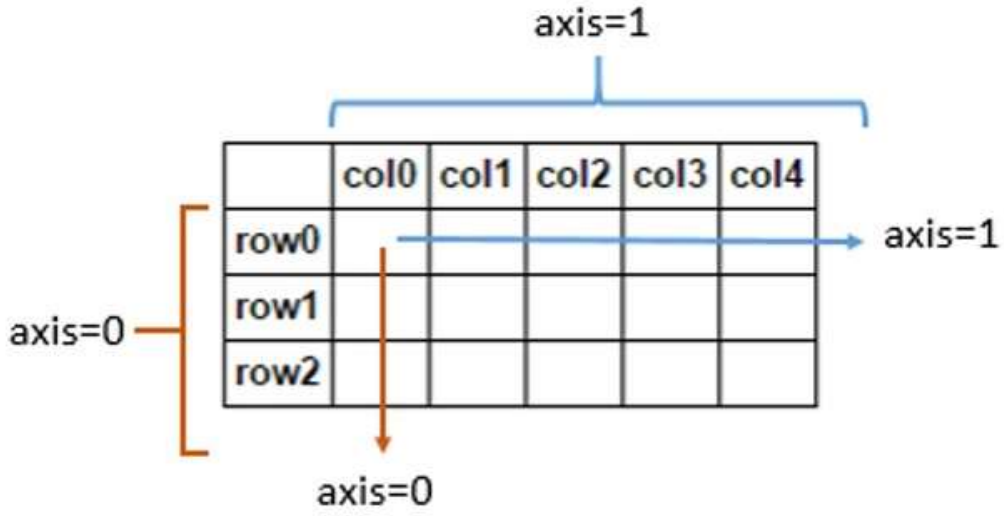
	col0	col1	col2	col3
row0				
row1				
row2				
row3				

Response Vector y

Series

	col
row0	
row1	
row2	
row3	

pandas axes



Source: Stack Overflow

- Axes are defined for arrays with more than one dimension. A 2-dimensional array has two corresponding axes: the first running **vertically downwards across rows (axis 0)**, and the second running **horizontally across columns (axis 1)**

❸ PARSE the Data

Codealongs

Part A – Introduction to pandas with the SF Housing dataset

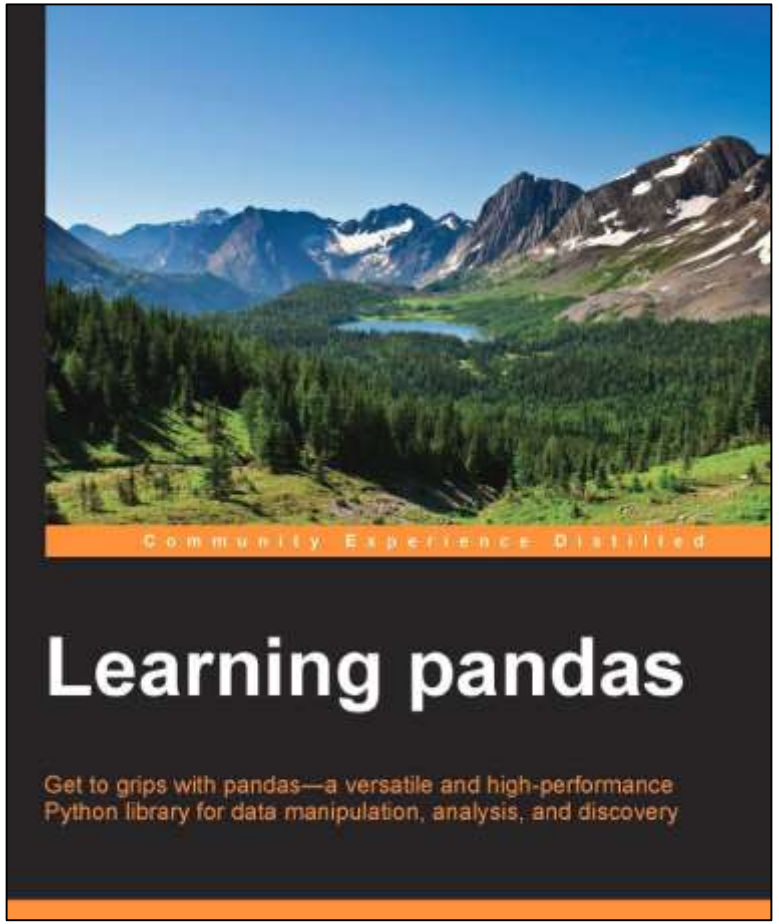
Part B – Wrangling the SF Housing dataset (take 2) with pandas

The codealong was just the tip of the iceberg. For more, check out the *pandas* documentation; also a good book *(again optional; not required for the course)*

- *pandas* documentation

  (which is very well written…)

  - http://pandas.pydata.org/pandas-docs/stable/



Lab

Research Design and pandas

Review

Review

You should now be able to:

- Define a problem and types of data
- Identify dataset types
- Apply the data science workflow in the *pandas* context
- Write an iPython notebook to import, format, and clean data using the *pandas* library

Next Class

Exploratory Data Analysis

Learning Objectives

After the next lesson, you should be able to:

- Identify variable types
- Use the *pandas* (and *NumPy)* libraries to analyze datasets using basic summary statistics: mean, median, mode, max, min, quartile, inter-quartile range, variance, standard deviation, and correlation
- Create data visualizations – including boxplots, histograms, and scatter plots – to discern characteristics and trends in a dataset

Exit Ticket

Don’t forget to fill out your exit ticket here