Tuesday, October 6, 2020 | **Class #1**

Data science, ocean data, Google Colab, and algorithmic thinking

OCEAN 215 | Autumn 2020

Ethan Campbell and Katy Christensen

In yesterday's news...

Search jobs | Sign in | Search | US edition

The Guardian

How Excel may have caused loss of 16,000 Covid tests in England

Public Health England data error blamed on limitations of Microsoft spreadsheet

▲ More than 50,000 potentially infectious people may have been missed by contact tracers after 15,841 positive tests were left off the daily figures. Photograph: Simon Leigh/Alamy

A million-row limit on Microsoft's Excel spreadsheet software may have led to Public Health England misplacing nearly 16,000 Covid test results, it is understood.

The data error, which led to 15,841 positive tests being left off the official daily figures, means than 50,000 potentially infectious people may have been missed by contact tracers and not told to self-isolate.

In this case, the Guardian understands, one lab had sent its daily test report to PHE in the form of a CSV file - the simplest possible database format, just a list of values separated by commas. That report was then loaded into Microsoft Excel, and the new tests at the bottom were added to the main database.

But while CSV files can be any size, Microsoft Excel files can only be 1,048,576 rows long - or, in older versions which PHE may have still been using, a mere 65,536. When a CSV file longer than that is opened, the bottom rows get cut off and are no longer displayed. That means that, once the lab had performed more than a million tests, it was only a matter of time before its reports failed to be read by PHE.

Microsoft's spreadsheet software is one of the world's most popular business tools, but it is regularly implicated in errors which can be costly, or even dangerous, because of the ease with which it can be used in situations it was not designed for.

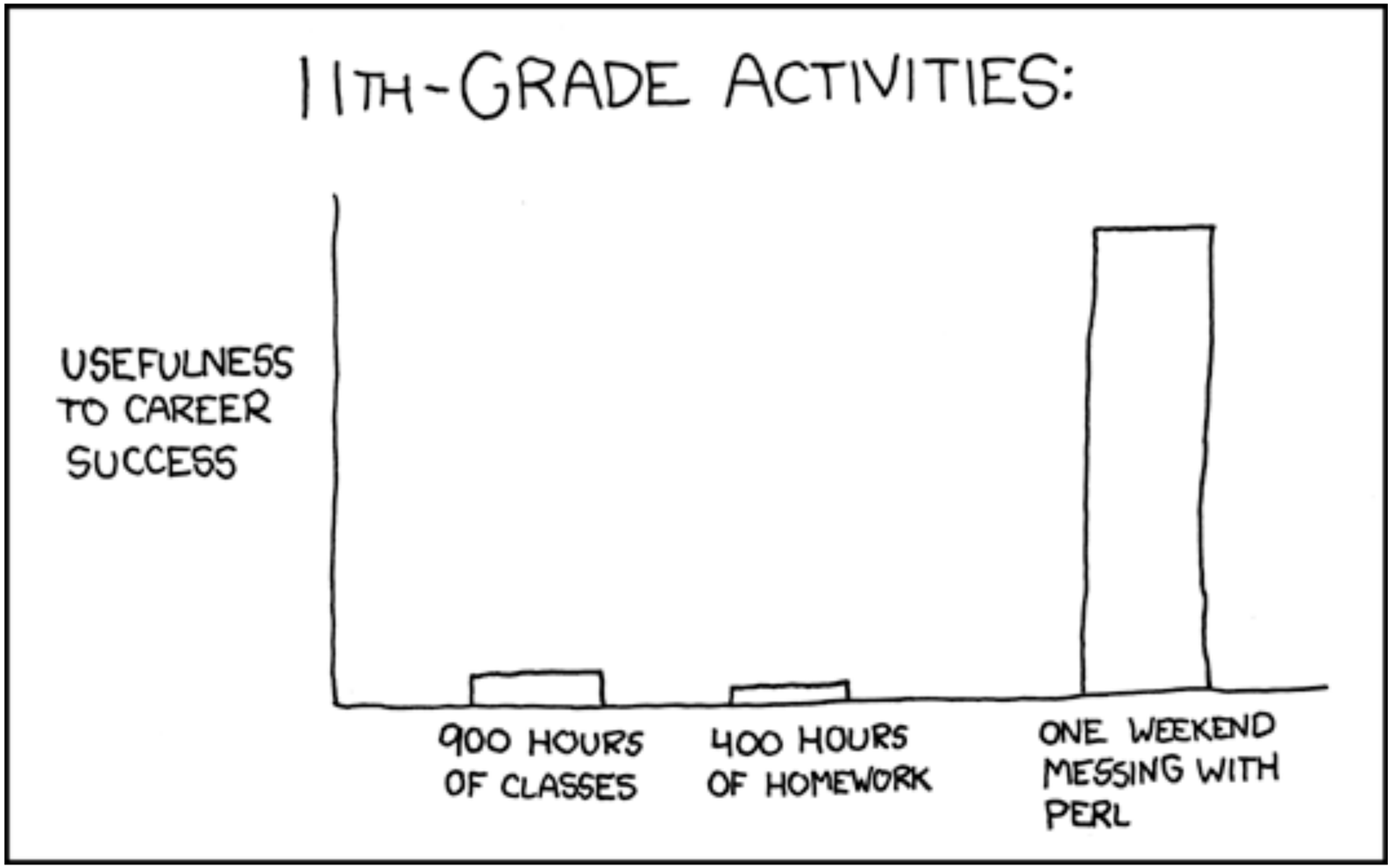
In 2013, an Excel error at JPMorgan masked the loss of almost $6bn (£4.6bn), after a cell mistakenly divided by the sum of two interest rates, rather than the average. The news led James Kwak, a professor of law at the University of Connecticut, to warn that Excel is "incredibly fragile".

"There is no way to trace where your data comes from, there's no audit trail (so you can overtype numbers and not know it), and there's no easy way to test spreadsheets, for starters. The biggest problem is that anyone can create Excel spreadsheets - badly. Because it's so easy to use, the creation of even important spreadsheets is not restricted to people who understand programming and do it in a methodical, well-documented way," Kwak wrote.

Errors from the spreadsheet software have even changed the very foundations of human genetics. The names of 27 genes have been changed over the past year by the Human Gene Nomenclature Committee, after Microsoft's program continually misformatted them. The genes SEPT1 and MARCH1, for instance, have been changed to SEPTIN1 and MARCHF1 after they were repeatedly turned into dates, while symbols that were common words have been altered so that grammar tools didn't autocorrect them: WARS is now WARS1, for instance.

Source: The Guardian

Other reasons to learn programming



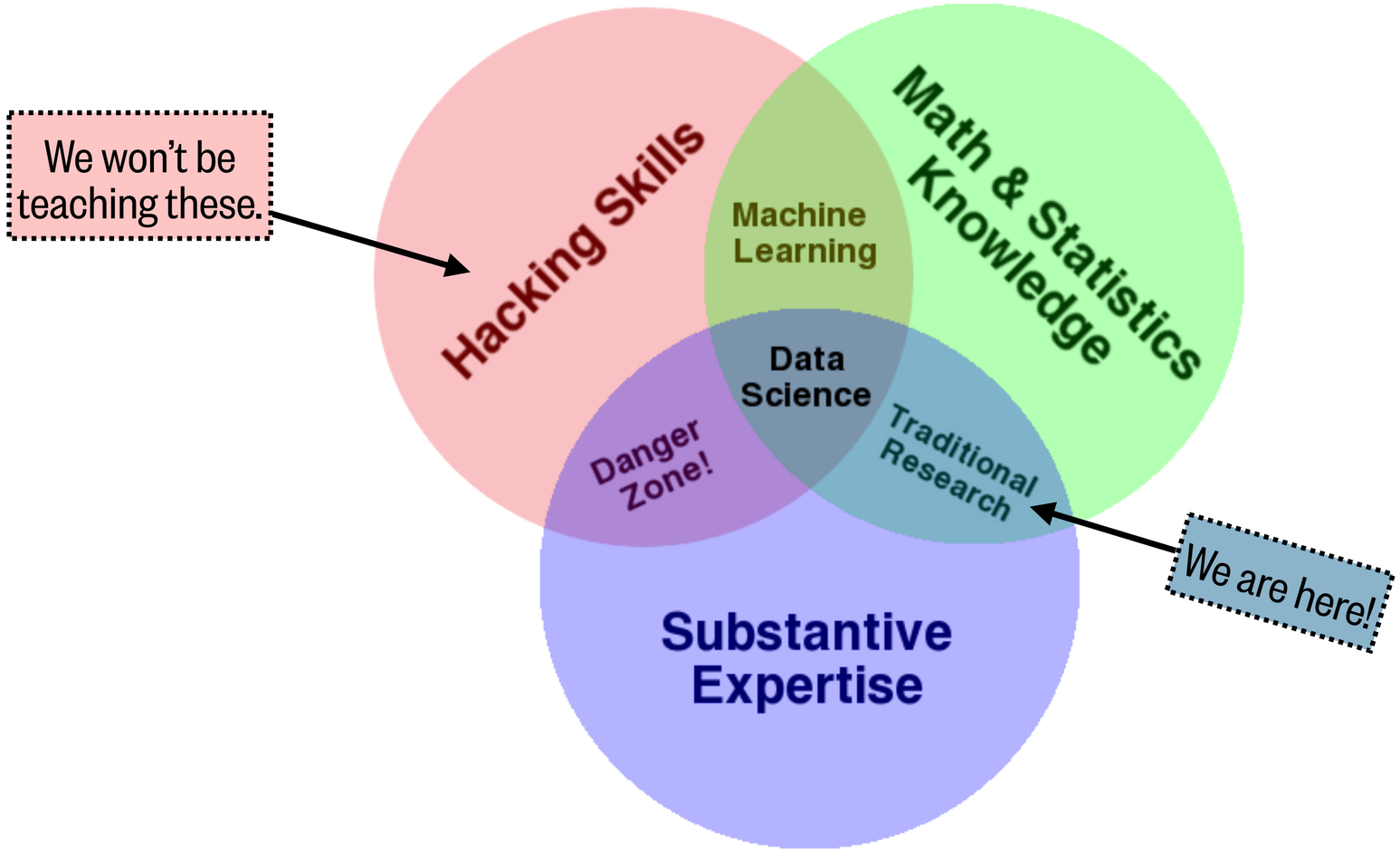
Source: XKCD

Data science is lucrative

"According to the McKinsey Global Institute's 'The Age of Analytics: Competing in a Data-Driven World' report, the U.S. will experience a shortage of 250,000 employees trained in data skills by 2024. Employers are responding by increasing salaries."

Source: Earth Lab, UC-Boulder

But what is data science, exactly?



Source: Drew Conway / Jake VanderPlas, Python Data Science Handbook

We can’t teach you to become a data scientist in a quarter

Technical skills and tools of a Data Scientist
Math (e.g. linear algebra, calculus and probability)
Statistics (e.g. hypothesis testing and summary statistics)
Machine learning tools and techniques (e.g. k-nearest neighbors, random forests, ensemble methods, etc.)
Software engineering skills (e.g. distributed computing, algorithms and data structures)
Data mining
Data cleaning and munging
Data visualization (e.g. ggplot and d3.js) and reporting techniques
Unstructured data techniques
R and/or SAS languages
SQL databases and database querying languages
Python (most common), C/C++ Java, Perl
Big data platforms like Hadoop, Hive & Pig
Cloud tools like Amazon S3

Source: Kempler and Mathews (2017)

Data science techniques have become integral to science

*"Until recently... one could be a world-class oceanographer without possessing knowledge of data science [techniques]. **But no more.** Oceanography, like so many other disciplines, is becoming an information field, through rapid advances in chemical, physical, biological, and video sensors that stream data with **unprecedented volume, velocity, and variety**; remotely operated vehicles; and observatories that extend the internet to the seafloor. Sophisticated analysis of data and innovation in data-analysis methods have become integral to the field."*

– Prof. Ed Lazowska (UW CSE; eScience Institute founder), writing in The Chronicle of Higher Education

Note: this is just one person's opinion.

“Unprecedented volume, velocity, and variety”

Let’s find some ocean data! In groups…

- Introduce yourselves (pronouns, years, major [and primary area of interest if Oceanography]). Assign timekeeper, notetaker, reporter roles.
- Locate an interesting ocean data set in your assigned category using the suggested database (or a different database).
- Skim the documentation. How was the data collected or created?
- Try to characterize its **volume, velocity, and variety** (it’s okay if you can’t track down all of these!):
    - **Volume:** time span, approximate size (e.g. in MB or GB), number of files, etc.
    - **Velocity:** how frequently is/was the data collected?
    - **Variety:** how many parameters were measured or recorded, and what are they?
- Brainstorm **two** scientific questions that one might be able to answer using your data.
- Take notes in this **shared Google Doc**: https://tinyurl.com/OCEAN215Class1
- Report back, including on your experience searching for the data.

Group	Category	Suggested database to search
1	Satellite remote sensing (physical)	NASA PO.DAAC (Physical Oceanography Distributed Active Archive Center)
2	Satellite remote sensing (biological)	NASA Giovanni
3	Biological or chemical ship measurements and field campaigns	NSF BCO-DMO (Biological & Chemical Oceanography Data Management Office)
4	Historical records (data from before 1980)	NOAA NCEI (National Centers for Environmental Information)
5	Earth system model (climate model) output	IPCC (Intergovernmental Panel on Climate Change) Data Archive
6	Laboratory experiments	PANGAEA

Shifting gears... the Python `print()` statement

Python will display:

```
print()
```
← *[blank new line]*

```
print('Show this message')
print("Show this message")
print('Show this', 'message')
```
← Show this message
[blank new line assumed hereafter]

```
print('5.3')
print(5.3)
```
← 5.3

```
print(5.3,'is a number')
```
← 5.3 is a number

Variable assignment

```
n = 5
```
↑ notice the spaces!

← n is assigned the value 5

Python will display:

```
print(n)
```
← 5 *[blank new line assumed hereafter]*

```
n
```
← 5

```
n = 'Hey ya'
```
← n is now assigned the value `'Hey ya'`

Python will display:

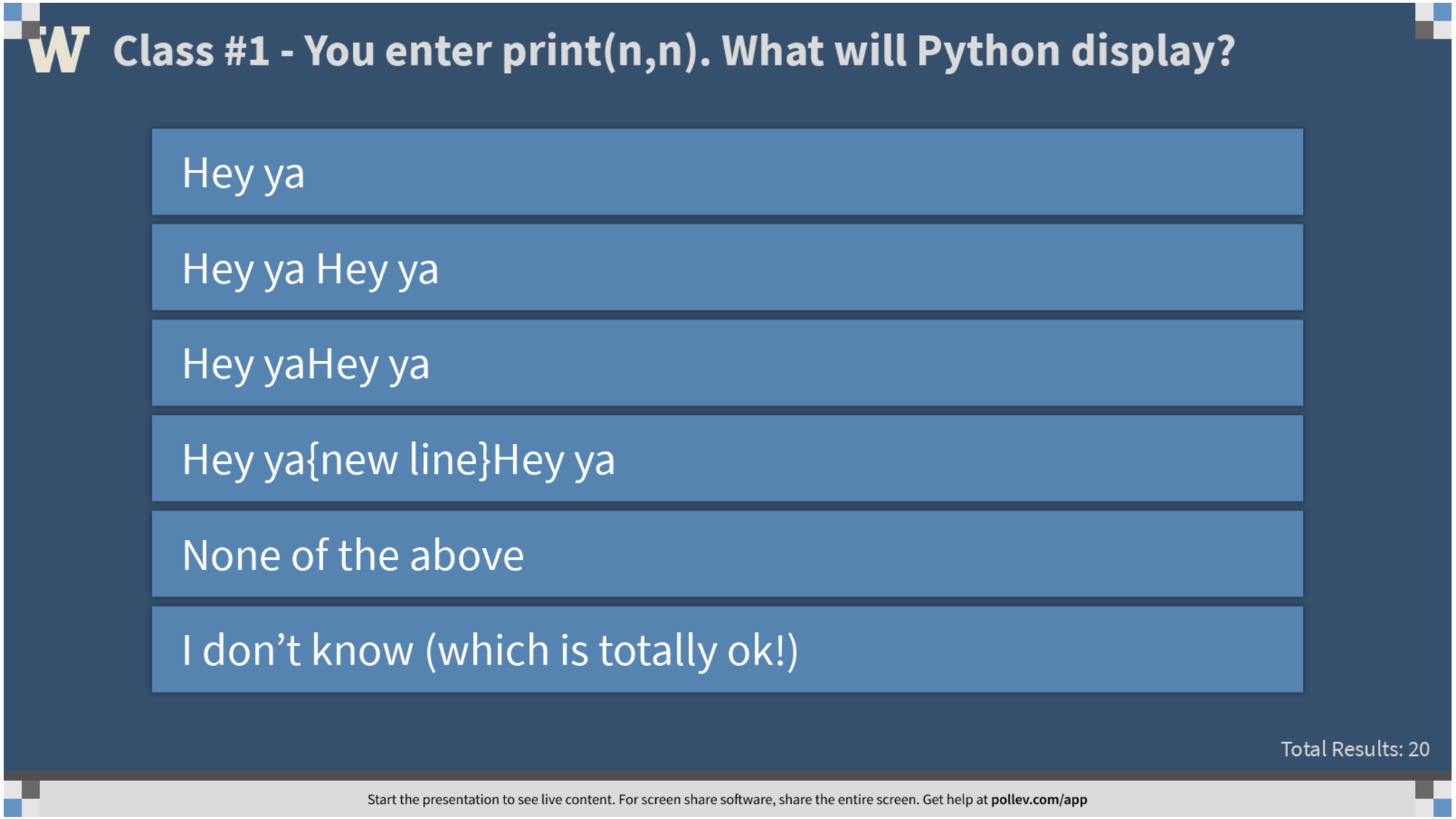
```
print(n)
```
← Hey ya

```
print(n, n)
```
↑ notice the space!

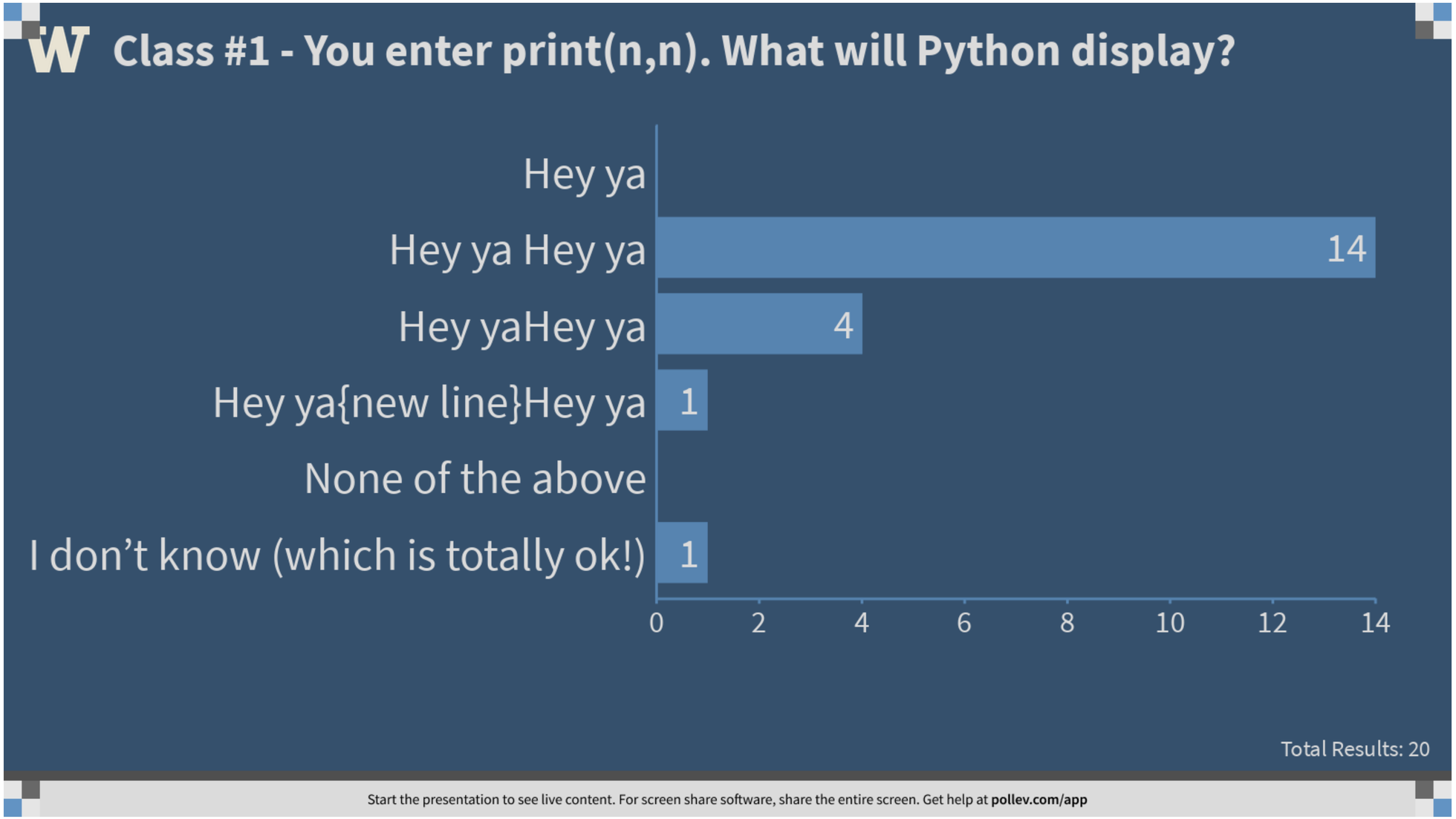
What do you think Python will display?

(Save that thought for the poll on the next slide…)

Variable assignment

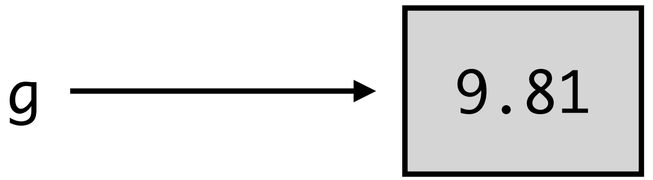


Variable assignment



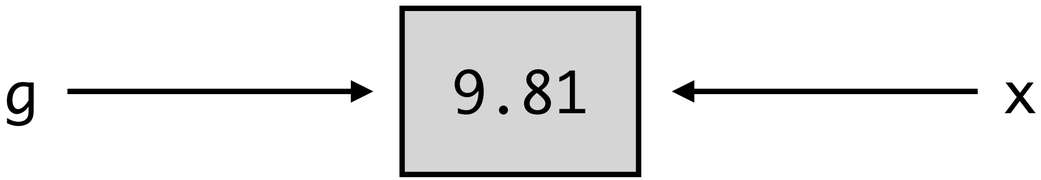
Object references

```
g = 9.81
```



g is actually referencing an "object" with the value of 9.81

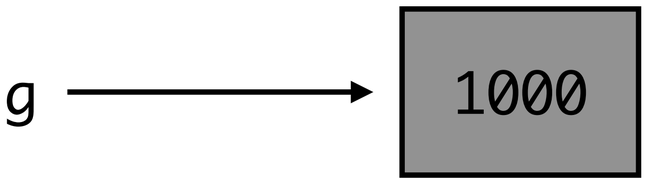
```
x = g
```



now g and x are referencing the same object

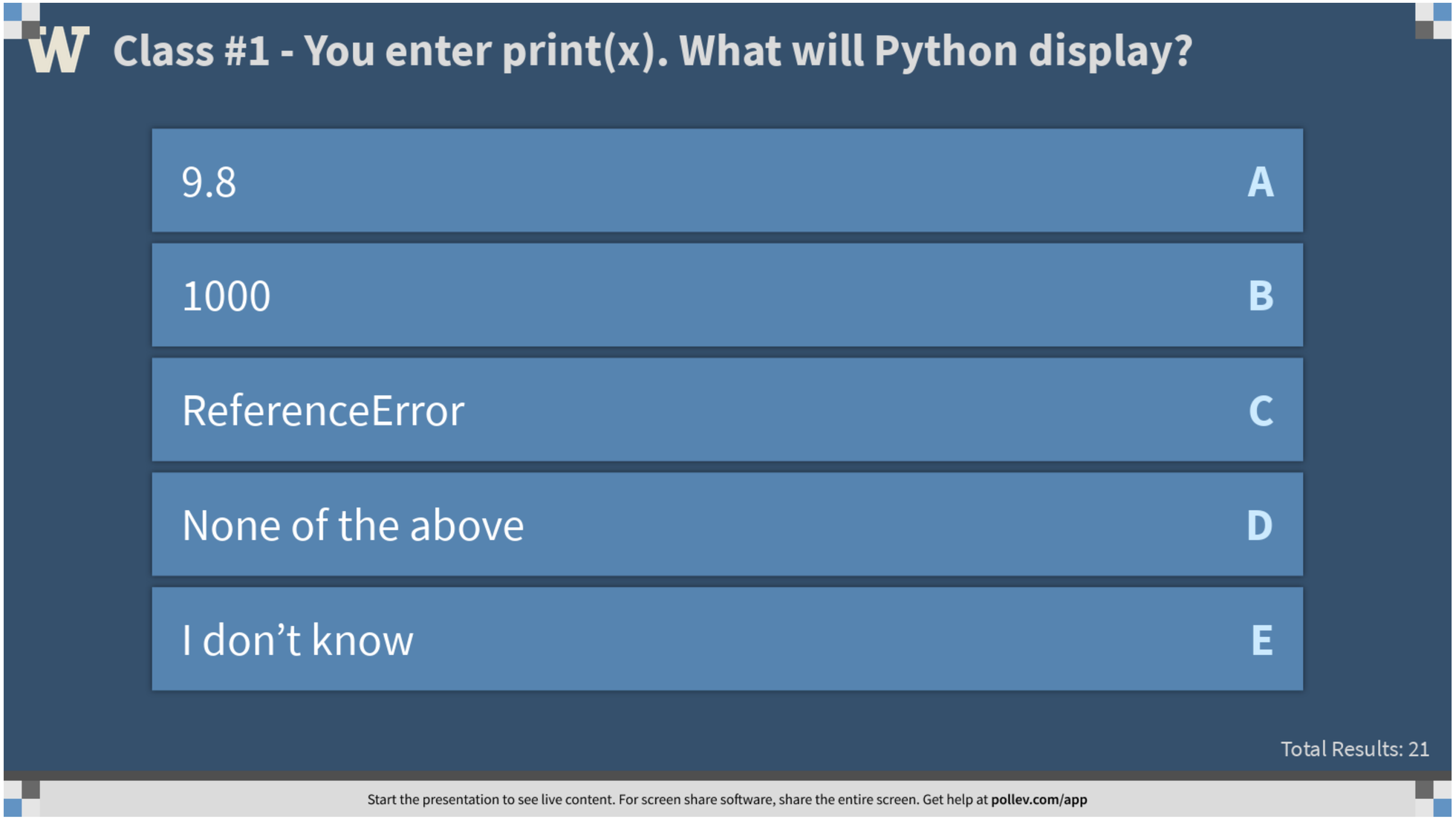
```
g = 1000

print(x)
```

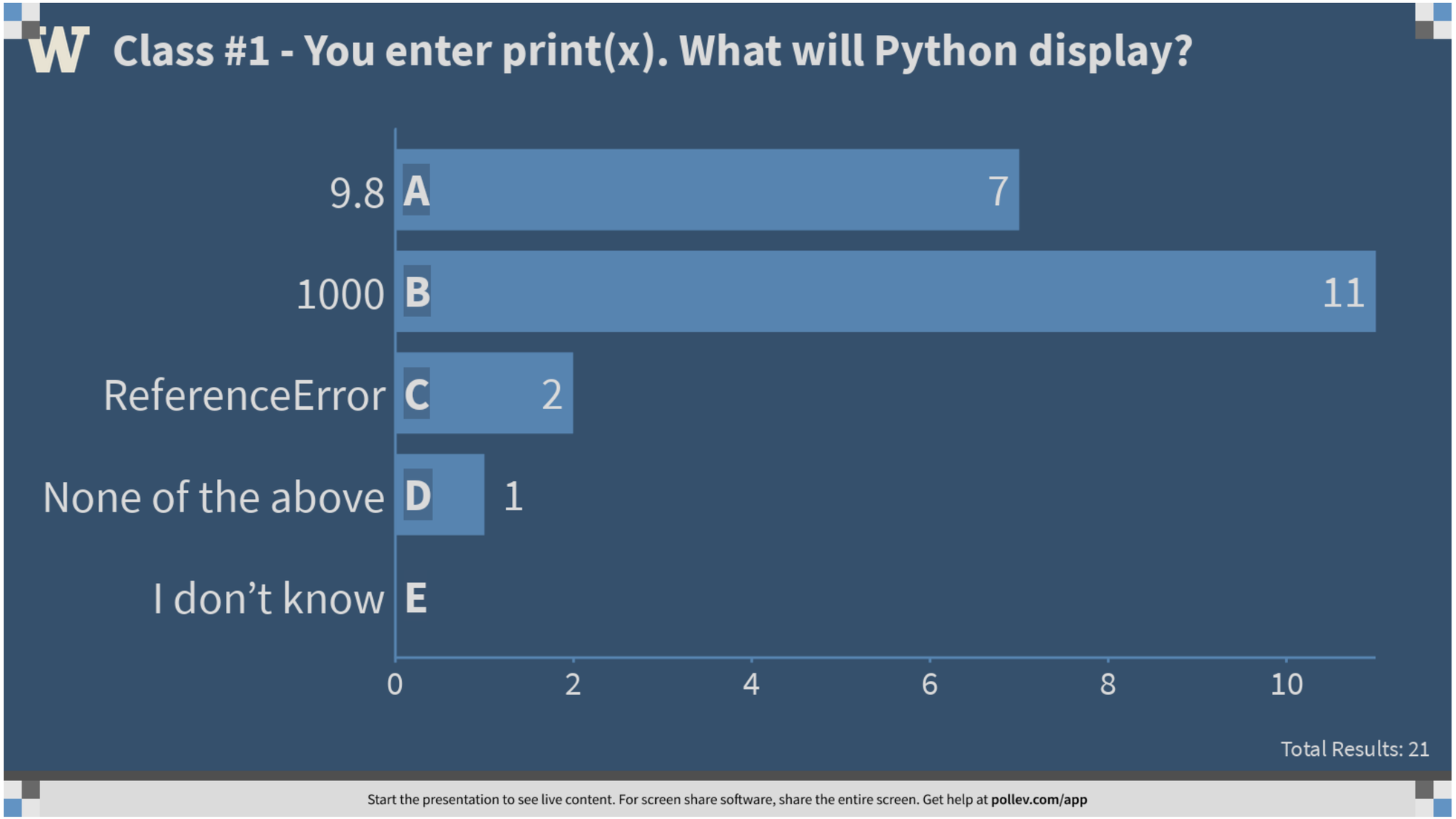


What do you think Python will display?
(Save that thought for the poll on the next slide…)

Object references



Object references



Variable name conventions

`g`	**single letters okay for physical constants or mathematical variables** (g is the gravitational constant)
`total`	**single words are the best variable names**
`number_of_students`	**use underscores for longer variable names** (notice all lowercase!)
`n_students`	**single letters are okay within longer names if their meaning is clear**
`numberOfStudents`	**this is called "Camel Case" – try to avoid it in variable names!**
`4th_student`	**variables cannot begin with a number** (Python will not let you do this!)
`i`	**avoid using single letters if their meaning is unclear**

(variables also must be alphanumeric and cannot contain spaces)

Style conventions, in general

> *"One of Guido [van Rossum]'s key insights is that* **code is read much more often than it is written**. *The guidelines provided here are intended to improve the readability of code and make it consistent across the wide spectrum of Python code."*

For more information on Python style conventions, see:

- Katy's Lesson #2 videos on Panopto
- The official Python PEP 8 Style Guide (the source of this quote)
- Google's Python Style Rules
- UW CSE 160's Python Style Guide

Commenting conventions

```
# This is a comment, so Python will ignore this entire line
#This is also a valid comment, but try to include the space next time

n = 5    # You can add comments on the same line as code

# print(5)       # Python will ignore this entire line

# This is a
# multiline comment

"""
This is another way
to write really long
multiline comments
"""
```

← Python won't execute these lines as code, but will print the text

Hands-on introduction to Google Colab

Open your browser (e.g. Safari, Firefox, Chrome) and navigate to:

colab.research.google.com

Computer recipes: some definitions

Algorithm: “a process or set of rules to be followed in calculations or other problem-solving operations, especially by a computer”

Pseudocode: “a plain language description of the steps in an algorithm... intended for human reading rather than machine reading”