



**Egad! It's Excel**

**Data Boot Camp**  
Lesson 1.2



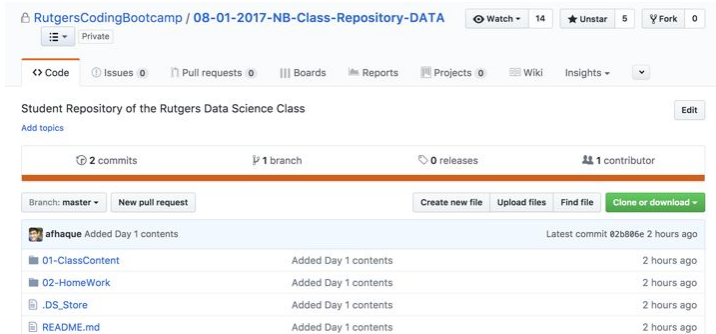
# A Few Admin Things

# Class Repository and Panopto Video Feed

01

## Class Git Repository

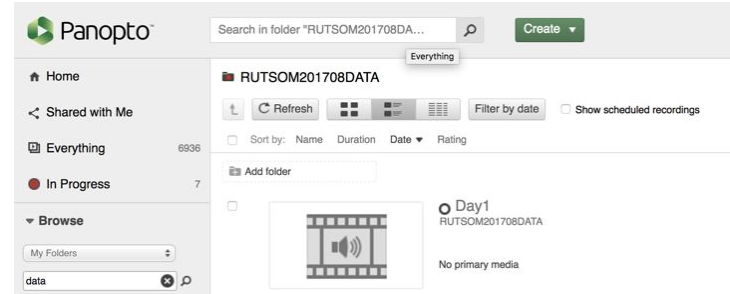
Classroom content  
Homework assignments



02

## Class Videos

Automatically uploaded  
On-demand videos



# Quick Refresher



Data analytics is about  
what **two** things?



Fundamentally, data analytics  
is about **storytelling** and  
**truth-telling.**



What are the steps in  
the **Analytics Paradigm**?

# Analytics Paradigm

---

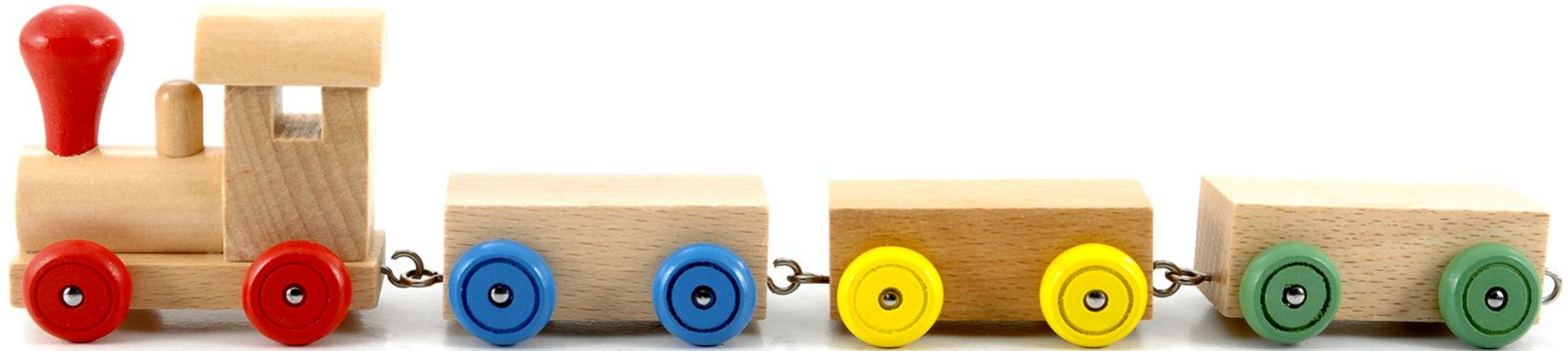
Regardless of type or industry, this paradigm provides a repeatable pathway for effective data problem solving.

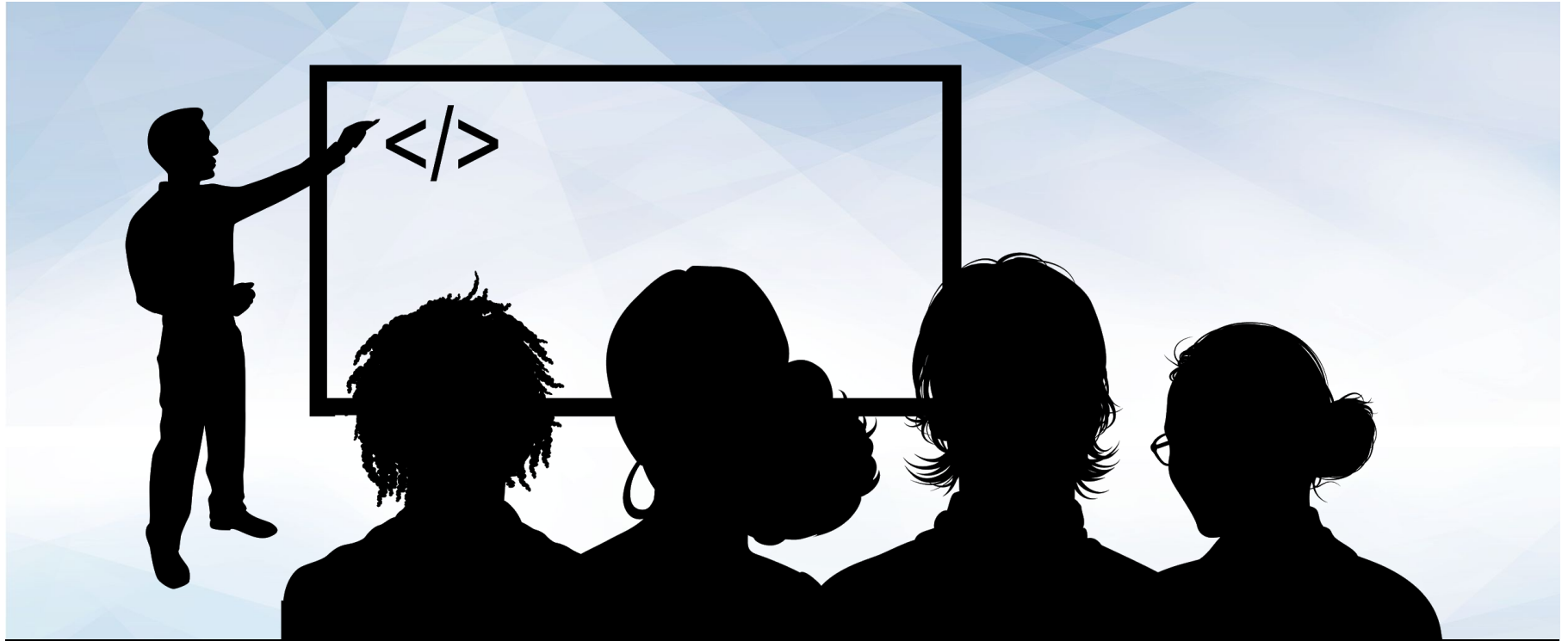




# Let's Start with the Basics

---





# Instructor Demonstration

## Excel Playground

Excited to get started?!



# Formulas

# Ooh...Coding! (Sort Of)

---

Excel has introduced you to a sort of proto-programming. When you write scripts, you will rely on **functions** (methods) that do something to or with **arguments**.



Function

Arguments

Function

# Ooh...Coding! (Sort Of)

---

When we reference a range or a set of ranges, Excel is given a set of **variable** inputs. Excel will determine the actual values of these inputs prior to executing the function.

=

AVG(

F4:F6

)

Function

Variable Arguments

Function

# Ooh...Coding! (Sort Of)

---



**What about this example?**

Which is the **function**?

Which are the **arguments**?

```
= SUM(  AVG(F4:F6),  AVG(G4:G6)  )
```

# Ooh...Coding! (Sort Of)

---



**What about this example?**  
Which is the **function**?  
Which are the **arguments**?



The **AVG functions** take the provided ranges as their arguments.

```
= SUM(  AVG(F4:F6),  AVG(G4:G6)  )
```



# Ooh...Coding! (Sort Of)

---



What about this example?  
Which is the **function**?  
Which are the **arguments**?



This is a **nested function**.  
We'll be doing plenty  
of complex nests in  
this class.

```
= SUM(  AVG(F4:F6) ,  AVG(G4:G6)  )
```

# You Can Code Too!

---

Here's a Python snippet from the last class.

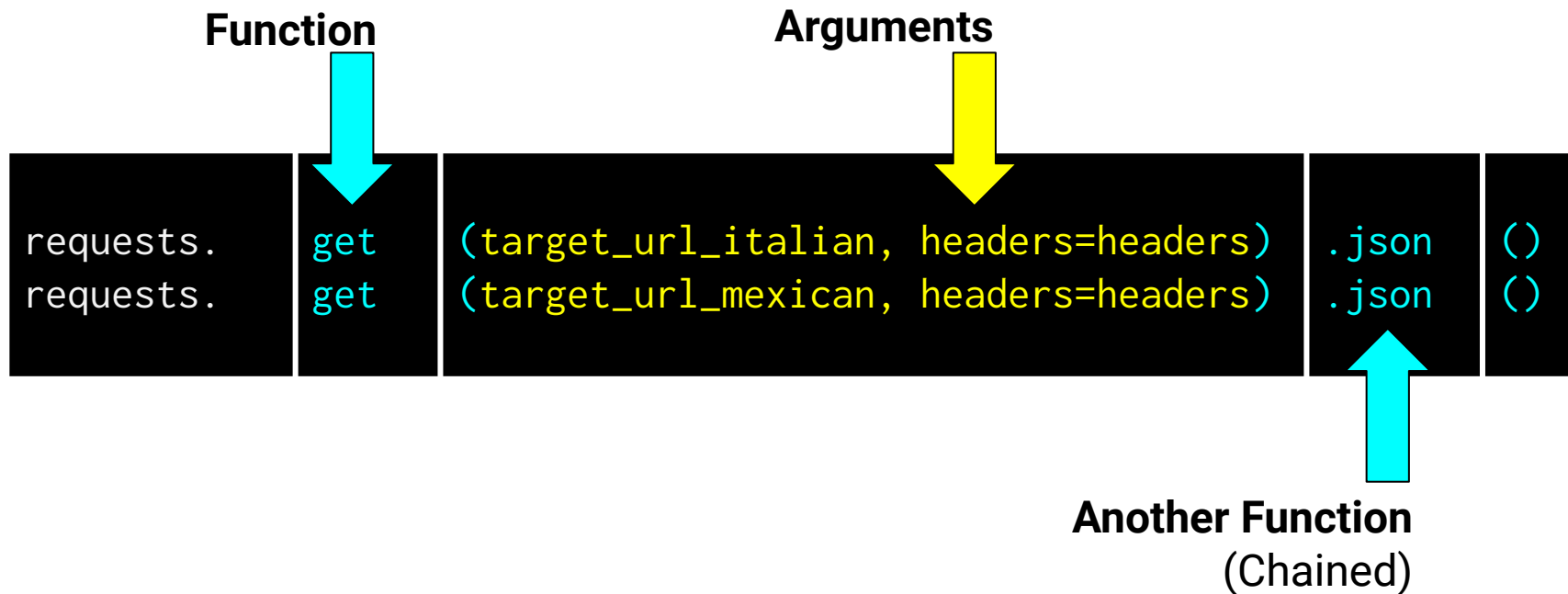


```
requests.get(target_url_italian, headers=headers).json()  
requests.get(target_url_mexican, headers=headers).json()
```

# You Can Code Too!

---

Syntax and capabilities may differ across technologies and platforms, but fundamental concepts remain the same.



# <Time for Excel>





# Instructor Demonstration

## Named Ranges

# There are multiple ways to select data in a formula

---

Most of us learned to select a range of cells to input into a function

```
=AVG(A1:A10)
```

# There are multiple ways to select data in a formula

---

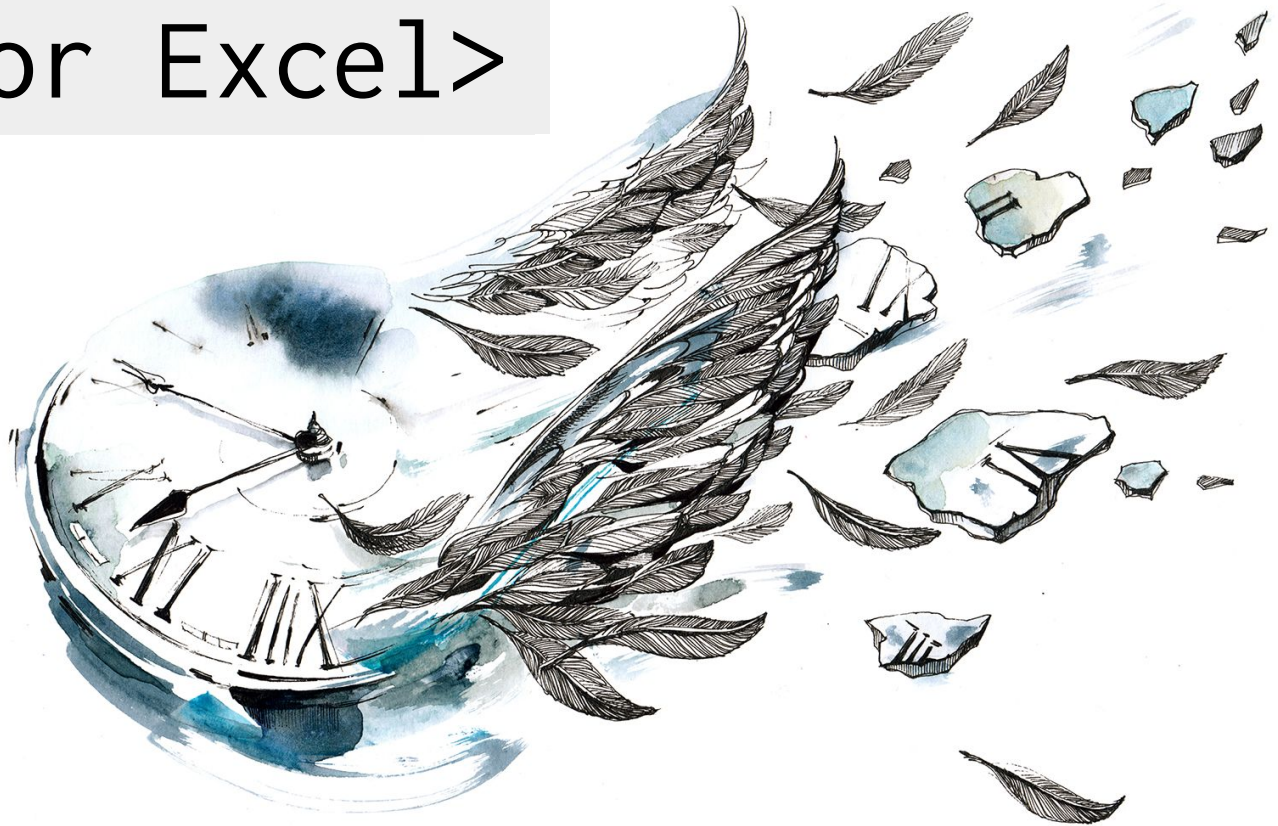
But we can name a range of values to make interpreting formulas easier!

```
=AVG(A1:A10)
```



```
=AVG(prices)
```

# <Time for Excel>







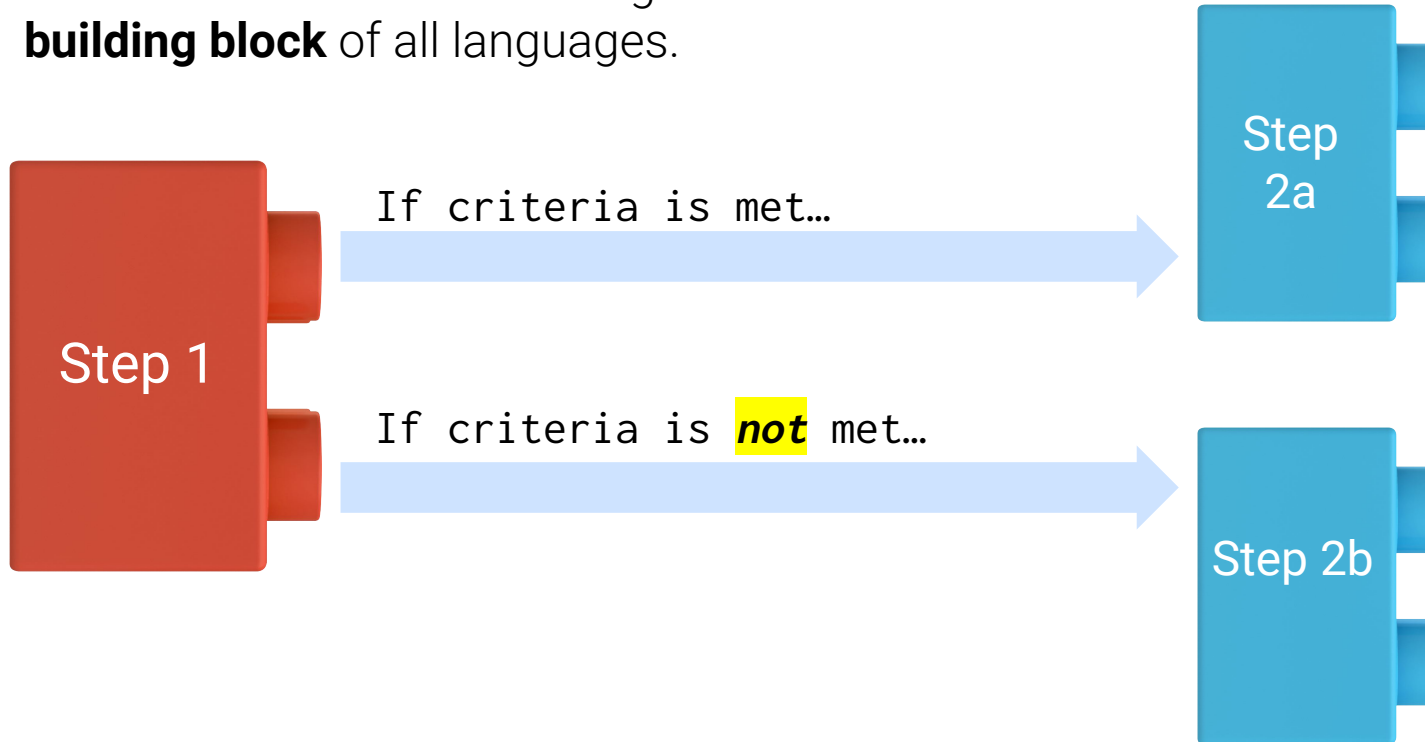
# Instructor Demonstration

## Color Counter

# Conditionals: If This, Then That

---

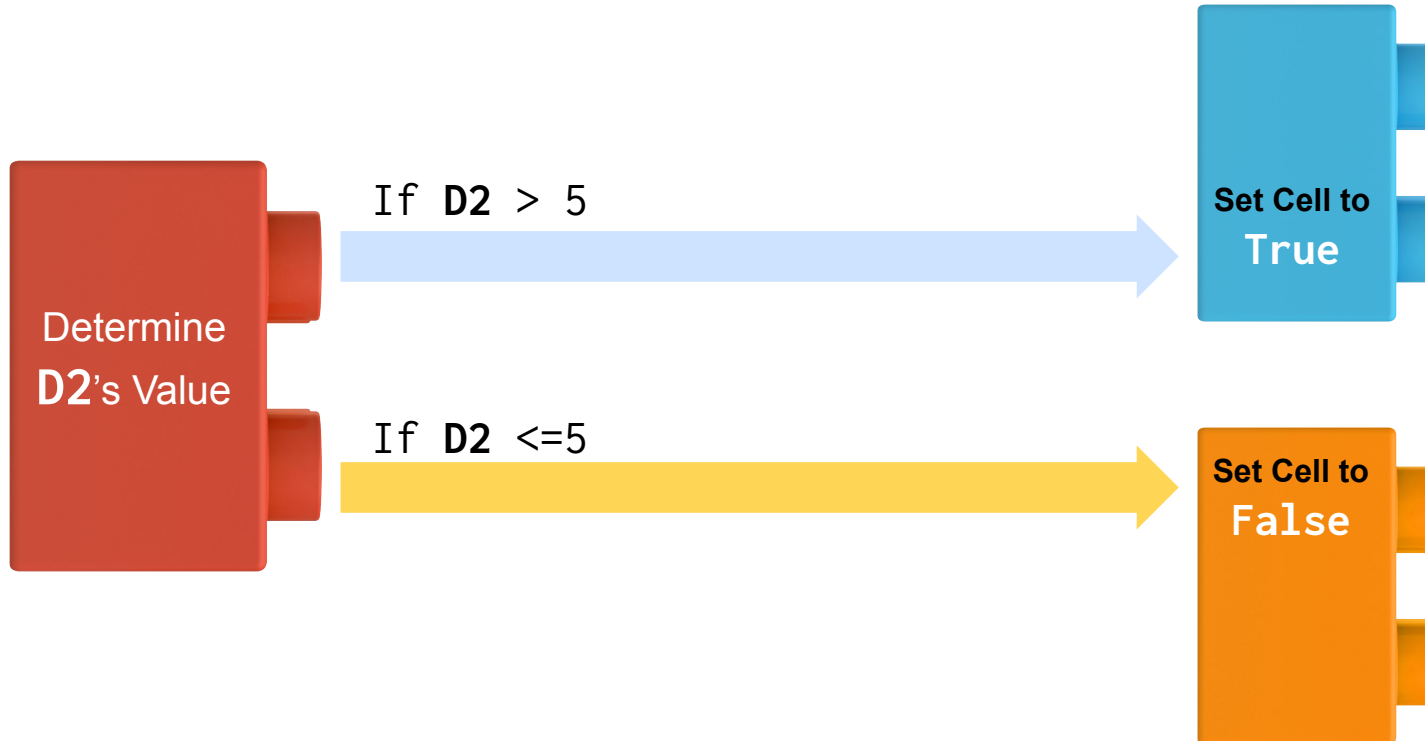
Conditionals present a way to control the flow of logic based on certain criteria being met. This is a **core building block** of all languages.



# Conditionals: If This, Then That

---

=IF(D2>5, TRUE, FALSE)





But what if we want to  
**combine** conditions?



AND, NOT, OR

## Ooh...Coding! (Sort Of)

---



But what if we want  
to **combine** conditions?



AND, NOT, OR

```
=IF(AND(D2>5, D2<10),TRUE,FALSE)
```

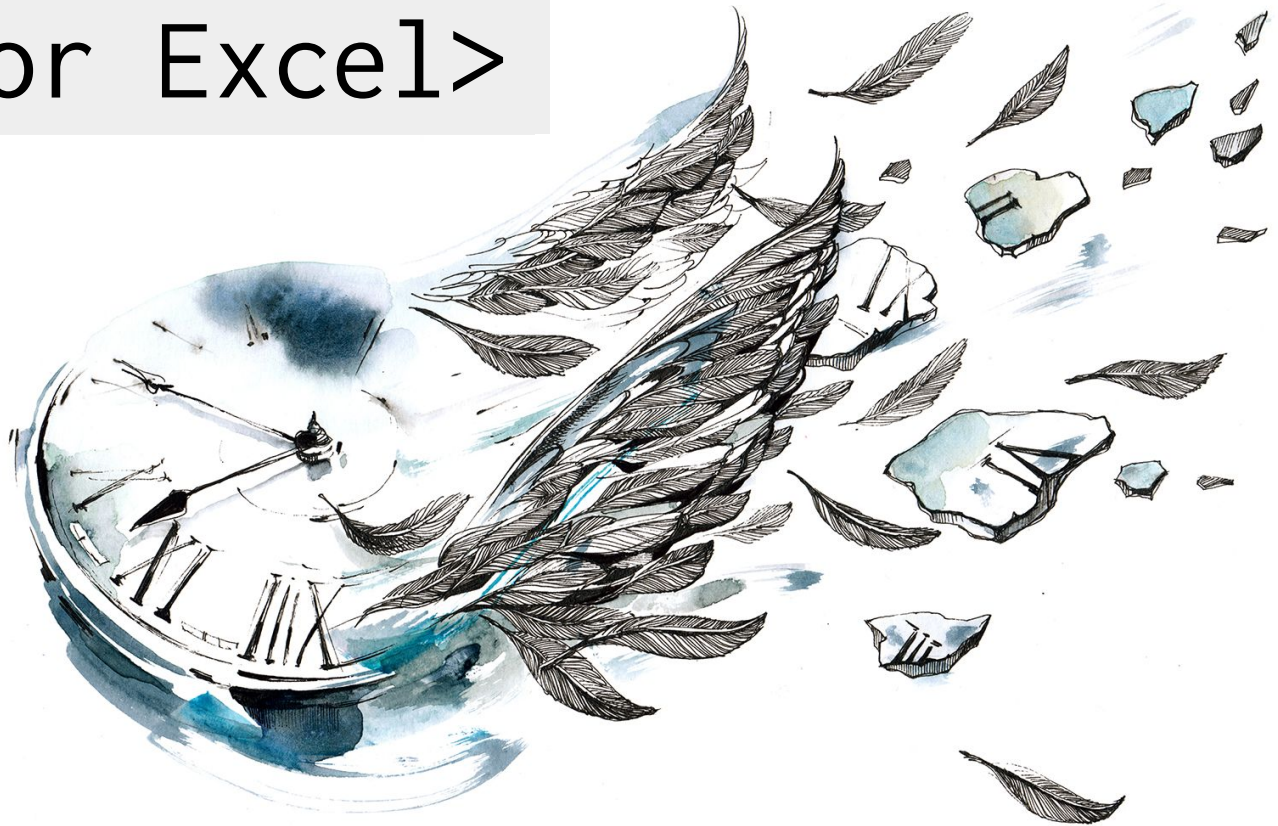
# Conditionals: If This, Then That

---

Nesting conditionals are powerful, but can become convoluted very quickly!



# <Time for Excel>







## **Activity:** Gradebook

Create a formula that calculates the final grade for a student based on their previous exams and papers.

**Suggested Time:**  
15 minutes



# Activity: Gradebook

---

## Todo:

- Create a formula which calculates the final grade for a student based upon their previous exams and papers.

## When making this calculation:

- Consider every paper and exam to be equal in weight; each should comprise one-fourth of the overall grade.
- Round the result to the nearest integer.
- Using conditionals, create a formula that returns **PASS** if a student's final grade is greater than or equal to 60. If a student's final grade is below 60, the formula should return **FAIL**.

## Bonus:

Create a nested **IF()** formula that returns a letter grade based on a student's final grade.

- Greater than or equal to 90 = A
- Greater than or equal to 80 and less than 90 = B
- Greater than or equal to 70 and less than 80 = C
- Greater than or equal to 60 and less than 70 = D
- Anything less than 60 = F

Suggested Time: 15 minutes





**Time's Up!** Let's Review.



# Instructor Demonstration

## Measures of Central Tendency



What are “measures of central tendency”?



Values used to describe  
the center of a data set.

# Three most common measures of central tendency

---

01

## Mean

- The “arithmetic” average
- **To calculate:** The sum of all values, divided by the number of values

02

## Median

- The middle value of a data set
- **To calculate:** Sort the data set and find the center

03

## Mode

- The most frequent value of a data set
- **To calculate:** Count the frequency of each value in a data set, determine the most frequent value

# <Time for Excel>





# Take a Break!

---





# Instructor Demonstration

## Formatting

# Formatting in Excel falls into two categories

---

01

## Data Formatting

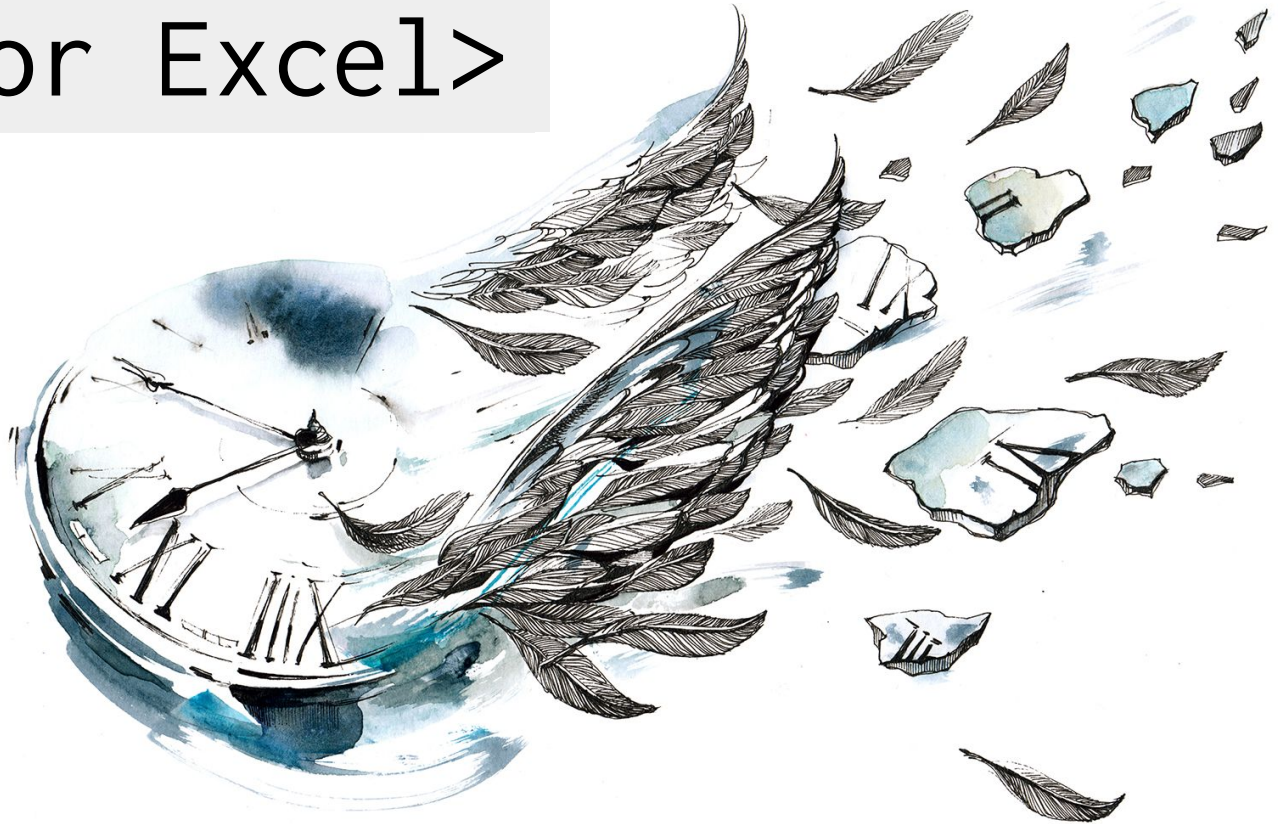
- Changes the way a value is represented in a cell.
- Used to help with interpretation or to add context to the range of values
- Examples:
  - Date and Time
  - Currency
  - Percentage
  - Scientific Notation

02

## Style Formatting

- Changes the way the cell and text are viewed
- Can include font color, cell highlighting, borders, etc.
- Can be performed manually or using formulas/logic (conditional formatting)

# <Time for Excel>





# Instructor Demonstration

## Pivot Tables

# Get Pivot With It

Pivot tables are one of the most important data visualization concepts to master in this class. (Don't worry. They are a cinch to deal with.)

The screenshot displays a spreadsheet with a PivotTable and the PivotTable Builder dialog box open. The PivotTable shows the sum of revenue by month and location. The PivotTable Builder dialog is configured to show the average revenue by reservation type.

Sum of Revenue	Column Labels			
Row Labels	Cambridge	Piccadilly	Grand Total	
2014	\$ 1,111,886	\$ 1,214,733	\$ 2,326,619	
January	\$ 90,005	\$ 94,910	\$ 184,915	
February	\$ 104,397	\$ 133,914	\$ 238,311	
March	\$ 53,546	\$ 80,115	\$ 133,661	
April	\$ 103,543	\$ 98,960	\$ 202,503	
May	\$ 111,353	\$ 93,664	\$ 205,017	
June	\$ 94,292	\$ 98,108	\$ 192,400	
July	\$ 112,334	\$ 73,953	\$ 186,287	
August	\$ 68,446	\$ 76,590	\$ 145,036	
September	\$ 82,581	\$ 152,078	\$ 234,659	
October	\$ 103,366	\$ 78,984	\$ 182,350	
November	\$ 82,564	\$ 134,740	\$ 217,304	
December	\$ 105,459	\$ 98,717	\$ 204,176	
2015	\$ 1,286,966	\$ 1,523,054	\$ 2,810,020	
January	\$ 134,521	\$ 96,206	\$ 230,727	
February	\$ 85,955	\$ 140,144	\$ 226,099	
March	\$ 129,781	\$ 151,357	\$ 281,138	

The PivotTable Builder dialog box is open, showing the following configuration:

- Name:** AverageRevenue
- Formula:** = Revenue / Reservations
- Fields:** Year, Quarter, Month, RoomType, Revenue, Reservations
- Filters:** (Empty)
- Columns:** RoomType
- Rows:** Year, Month
- Values:** Sum of Revenue

# Get Pivot With It

---

In essence, a pivot table is a **summative** analytic tool that allows us to perform aggregate functions that allow any combination of fields. (The term *pivot table* comes from the fact that we are pivoting along a data axis).

Seller	Qty. Sold	Date
Joseph	\$42.50	1/1/17
Jacob	\$65.00	1/3/17
Jacob	\$5.25	1/6/17
Joseph	\$125.00	1/6/17
Jacob	\$3.50	1/7/17
Matt	\$32.00	1/9/17

Seller	Total Sold
Joseph	\$167.50
Jacob	\$73.75
Matt	\$32.00

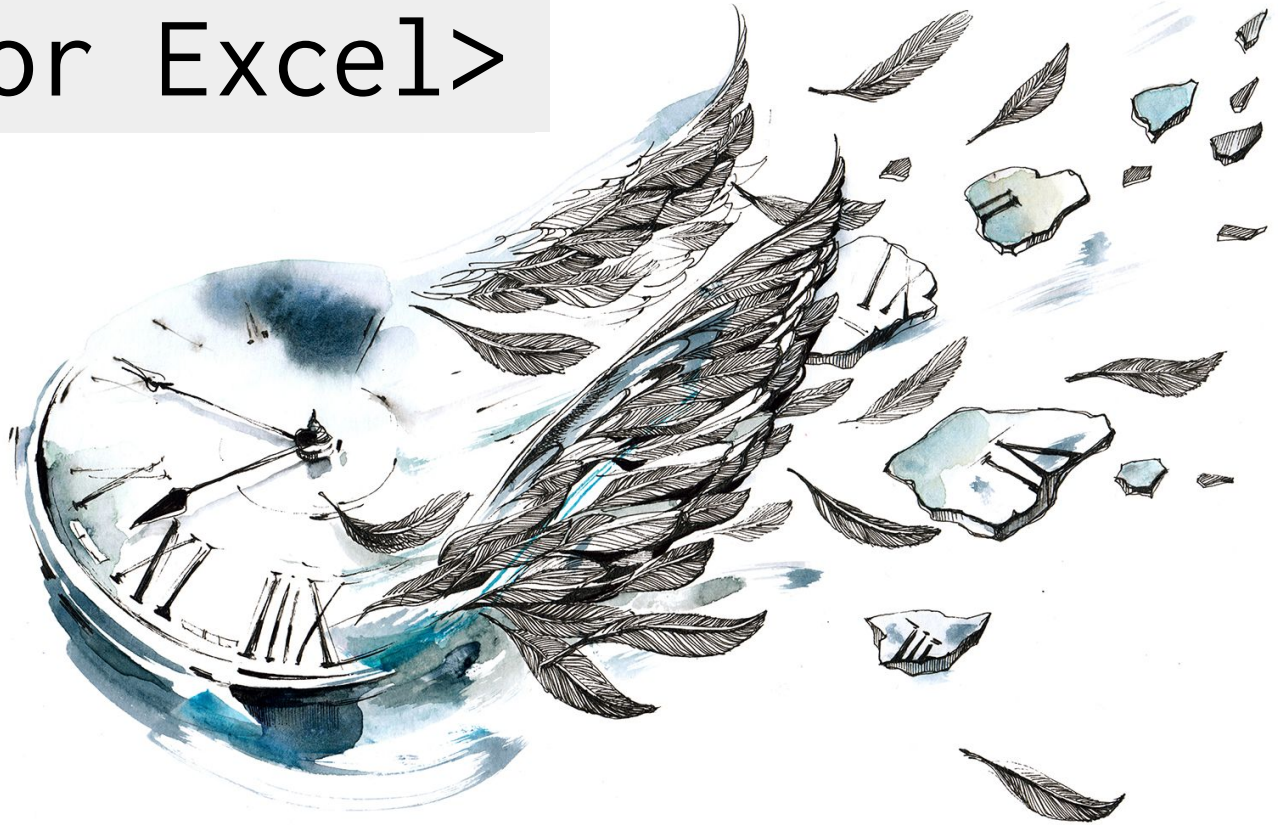
# Word to the Wise: Keep It Flat!

Modern Business Intelligence (BI) tools like Tableau, Sisense, and Salesforce work best if data is stored in flat CSVs—meaning column headers represent fields (vertically) on the spreadsheet. This is largely because all of these technologies heavily utilize pivot tables as a tool for their visualizations. **Don't try to confuse this simplicity. "Spreadsheet magic" is a nightmare to analyze.**

B	C	D	E	F	G	H
DateTime	Week #	Section?	Pace	Academic Support	Self-Mastery	Instructor Error
2016-09-11T04:00:00.000Z	18	RCB0503FSF - CCC	3	5	5	4
2016-09-11T05:00:00.000Z	6	UT0726FSF	3	5	3	4
2016-09-12T04:00:00.000Z	11	UCF062016FSF	4	4	3	5
2016-09-12T04:00:00.000Z	23	UCF0329FSF	2	4	5	1
2016-09-12T04:00:00.000Z	9	UNC0712FSF	3	4	4	3
2016-09-12T04:00:00.000Z	23	UCF0328FSF	4	3	2	3
2016-09-12T04:00:00.000Z	6	RUT0725FSF-NB	5	4	4	5
2016-09-12T04:00:00.000Z	6	RUT0725FSF-NB	5	5	4	5
2016-09-12T04:00:00.000Z	6	RUT0725FSF-NB	2	4	4	4
2016-09-12T04:00:00.000Z	11	UCF062016FSF	4	5	4	5
2016-09-12T04:00:00.000Z	13	UCF061416FSF	4	5	1	5



# <Time for Excel>





## **Activity:** Top Songs Pivot Table

In this activity, you will use a 5000 row spreadsheet containing data for the top 5000 songs from 1901 onward. Using pivot tables, you will uncover which artists have the most songs in the top 5000, the song titles, and the year each song was released.

**Suggested Time:**  
17 minutes



# Top Songs Pivot Table Instructions

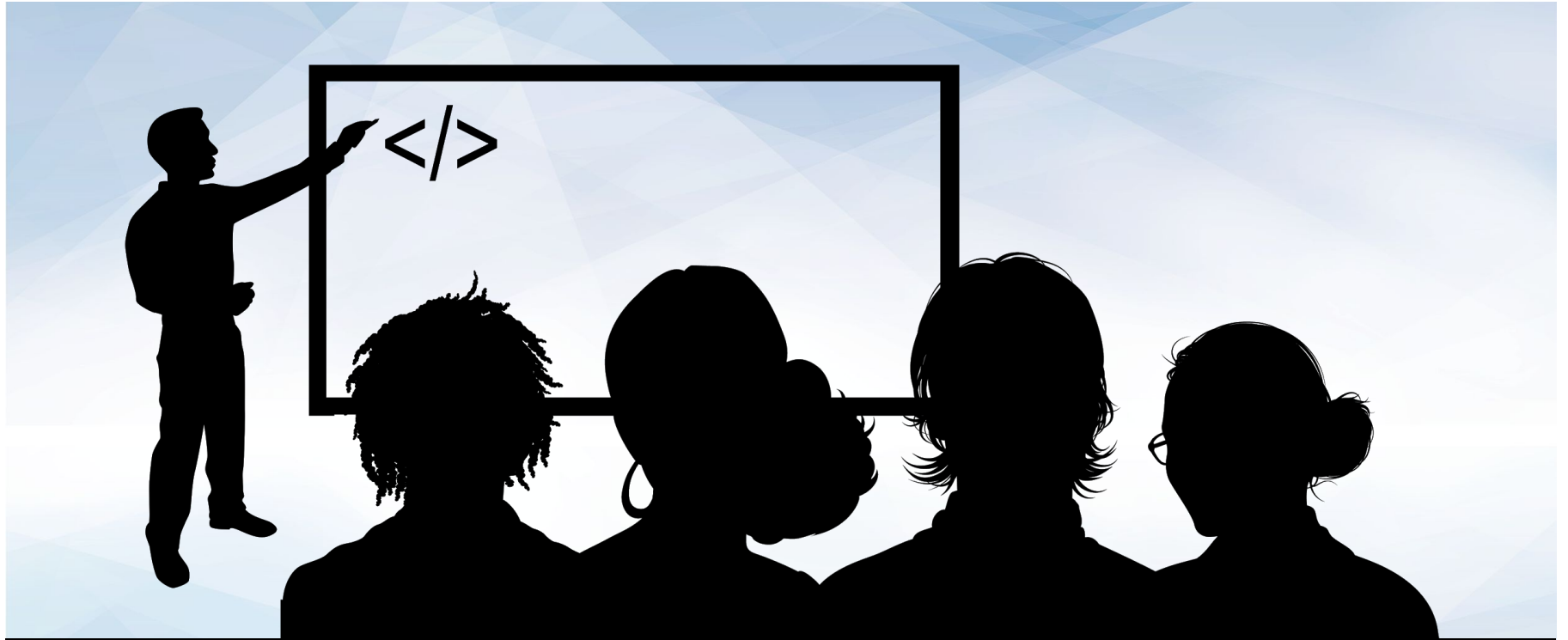
---

- Select all of the data in your worksheet and create a new pivot table.
- Make a pivot table that can be filtered by year and contains two rows: *Artist* and *Name*.
- All of an artist's songs should be listed below their name.
- Update your pivot table to contain values for:
  - How many songs an artist has in the top 5000
  - The sum of the `final_score` of their songs.
- Sort your pivot table by descending sum of the `final_score`.





**Time's Up!** Let's Review.



# Instructor Demonstration

## Lookups

# Look It Up with Lookups



Assume this table is gigantic. How would we **retrieve** the population of a specific planet for use in another formula?

Planet	Population
Zeelo	5020
Merinoa	380
Cardboard Box	2
...	...
Asteroid 9	95

# Look It Up with Lookups



Assume this table is gigantic. How would we **retrieve** the population of a specific planet for use in another formula?



```
=vlookup( <value>, <full table>,  
<column to retrieve>,<match parameter>)
```

Planet	Population
Zeelo	5020
Merinoa	380
Cardboard Box	2
...	...
Asteroid 9	95

# Look It Up with Lookups

---



What will this yield?

`=vlookup( "Asteroid 9", Planets, 3, FALSE)`

Planet	Population	Species
Zeelo	5020	Zoltans
Merinoa	380	Murphies
Cardboard Box	2	Hambones
...	...	
Asteroid 9	95	Asterisks



# Look It Up with Lookups



What will this yield?

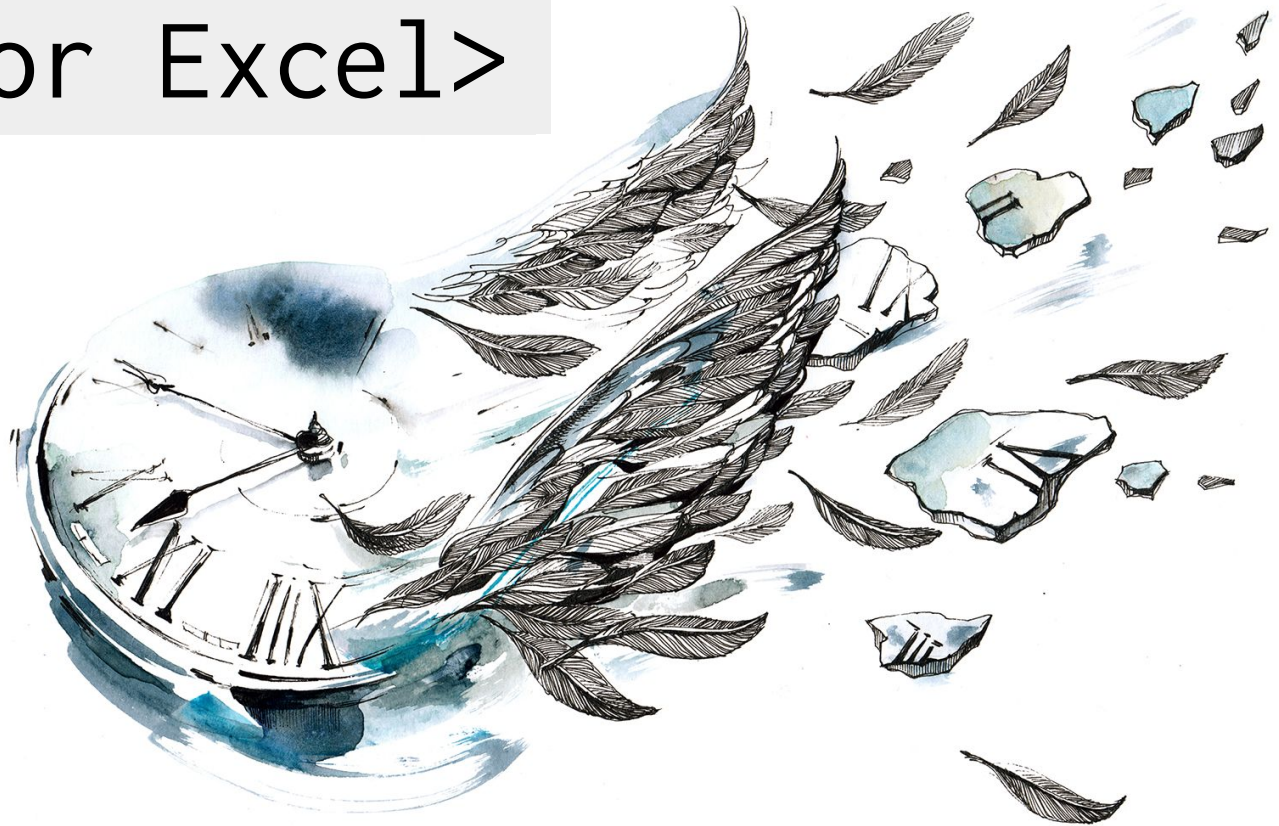
`=vlookup( "Asteroid 9", Planets, 3, FALSE)`

Planet	Population	Species
Zeelo	5020	Zoltans
Merinoa	380	Murphies
Cardboard Box	2	Hambones
...	...	
Asteroid 9	95	Asterisks



Asterisks

# <Time for Excel>





## Partner Activity: Product Pivot

A small company selling electronics and electronic media has asked our class to create a table that visualizes the cost of their recent orders. Using lookups, create a pivot table that serves this purpose.

**Suggested Time:**  
15 minutes



# Activity: Product Pivot

---

- Determine the "Product Price" of each row in the "Orders" sheet by using a **VLOOKUP()** that references each row's "Product ID"
  - The "Product Price" of a row does not include shipping
- Determine the "Shipping Price" of each row in the "Orders" sheet by using a **VLOOKUP()** that references each row's "Shipping Priority"
- Select all of the data on the "Orders" sheet and create a new pivot table that calculates the sum of both "Product Price" and "Shipping Price" for each "Order Number" and "Product ID"

Suggested Time: 15 minutes

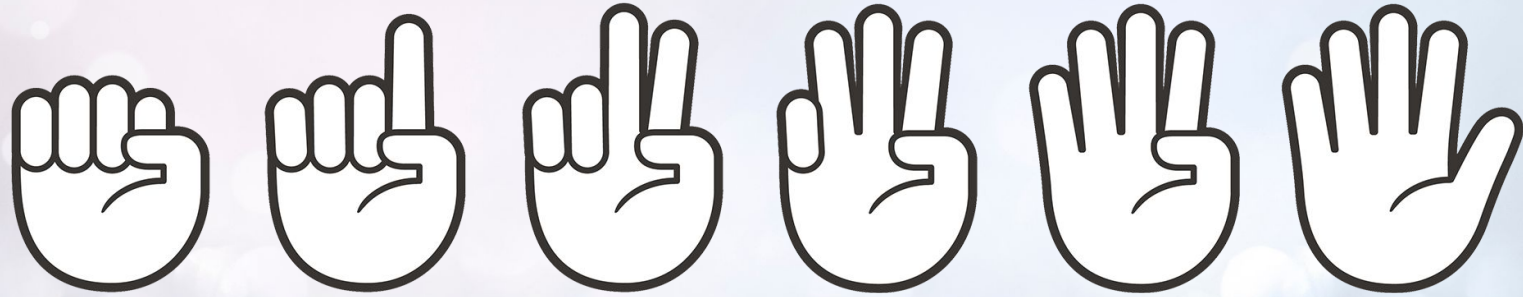




**Time's Up!** Let's Review.



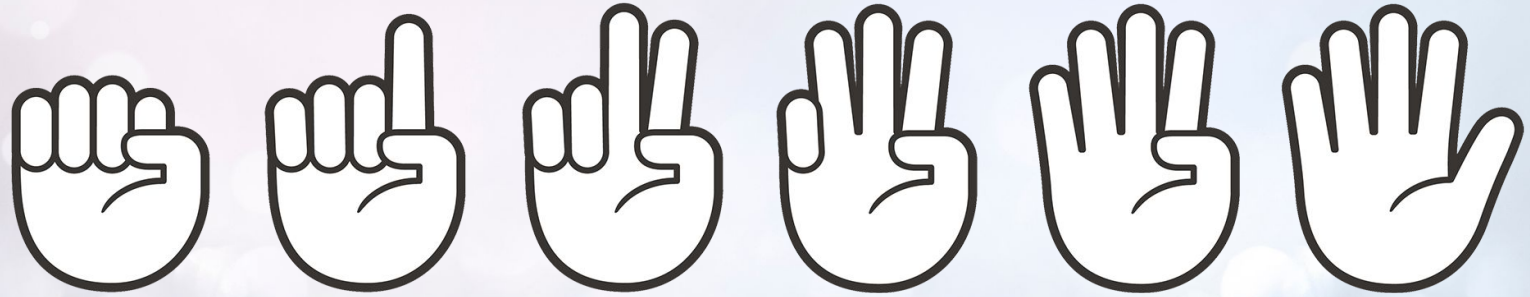
Questions?



## **FIST TO FIVE:**

---

Who feels comfortable  
with pivot tables in Excel?



## **FIST TO FIVE:**

---

Who feels comfortable  
with the Measures of Central Tendency?