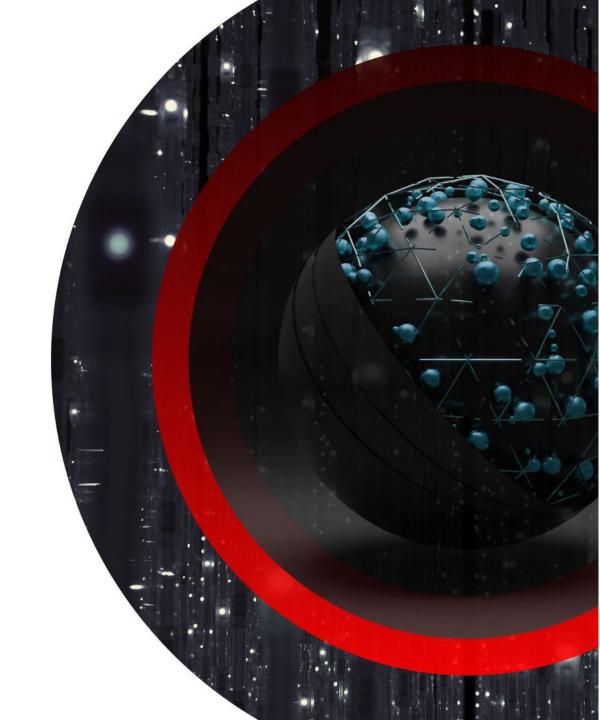
## DATA SCIENCE AND ANALYTICS

Introductory Course



**CLASS** 

**NORMS** 











#### **COURTESY IN CLASS**

Remaining on mute unless called on, exercising courtesy during breakout rooms, using the chat box for questions only

#### **ATTENDANCE**

100% attendance is expected and contributes to success in passing the course and the program

#### **PARTICIPATION**

Keeping an open mind in discussions and sharing experiences, making contributions during team assignments, submitting assignments in Canvas, and participating in discussion boards

#### **USE OF CLASS RESOURCES**

Follow along during the lecture with the lesson companion and download any in-class documents prior to class.



**INSTRUCTIONAL TEAM** 



### INSTRUCTIONAL TEAM

Contact information

Name

Contact information

Name

Contact information

Name





**PROGRAM** 

**PATH** 











#### **1 Introductory Course**

**2** SQL and Databases

**3** Statistics and Probability

4 Data Storytelling

Milestone 1: Building and

Presenting Data Stories

#### **5** Python Programming

**6** Data Wrangling

**7** Visual Communications

8 Advanced SQL Programming

Milestone 2: Data Integration,

Preparation, Reporting, and

Presentation

**9** Business Intelligence

**10** Big Data

11 Machine Learning

**12** Applied Al

Milestone 3: Capstone Project:

Delivering Insights and

Presentations











## INTRODUCTORY COURSE PATH

3

PROGRAMMING CONCEPTS

4

DISCOVERING AND CURATING DATA

5

**INTRODUCTION TO** 

DATA SCIENCE AND ANALYTICS

STRUCTURING AND ANALYZING DATA

6

2

**COMPUTING** 

**PRIMER** 

CLEANING AND ENRICHING DATA

7

VALIDATING AND PRESENTING DATA

8

INTRODUCTION TO DATA SCIENCE PROJECTS

9

**ASSESSMENT NIGHT** 









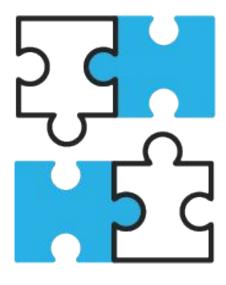


## MODULE OUTLINE

**Lesson 1:** Variables and Operations

**Lesson 2:** Functions

**Lesson 3:** Objects













## MODULE GOALS

Identify different variable types to understand how each one functions.

Perform operations on variables to demonstrate how they interact with one another.

Identify and describe the components of a function such as its name, parameters, and outputs.

Explain how objects store and classify data.











# LESSON 1: VARIABLES AND OPERATIONS.











## LESSON OUTLINE.

- Variables and what they mean for our data
- Different variable types
- Variable interactions with different operations













### DISCUSSION TIME

Why does it make sense to give elements of your data a name?















## WHAT ARE THE GOALS?

- Define variables in our data to identify them.
- Identify different variable types to understand how each one functions.
- Perform operations on variables to demonstrate how they interact with one another.

## WHY ARE THEY IMPORTANT?

Throughout your career, you will work with different types of variables and will need to know their characteristics to perform accurate analysis. Operations on variables will help you find key insights about a data set that are not immediately obvious.













What do you wish to accomplish by completing this module?













### WHAT IS A VARIABLE?

A variable is composed of three parts: a name, a type, and a value.

The variable must be assigned a **name**, and this name should represent the value it contains.

This **value** can be any piece of our data, such as the number 5, the word *coconut*, a chapter in a book, an image, a sound file, or even the whole data set.











### **TYPES OF VARIABLES**

Туре	Use	Examples	
String (str or text)	A combination of any characters, such symbols, letters, or numbers	"moo" "this is a string"	
Character (char)	Single letters	"a", "b"	
Integer (int)	Whole numbers	5, -3, 0	
Boolean (bool)	True/false, yes/no True, False		
Float (real)	Numbers with decimal points or fractions	95.80	



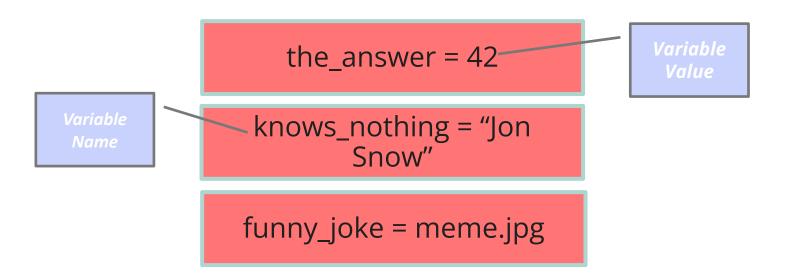








### VARIABLE EXAMPLES













### DISCUSSION TIME

Assign a variable type to each variable listed below:

- 1. this\_number = 3.6
- 2. total = 400
- 3. birthdate = 10-09-1981
- 4. sentence = "The quick brown fox jumps over the lazy dog"













### VARIABLES IN PRACTICE

#### Calculator

\$3.00
\$4.50
\$8.00
\$15.50

#### **Excel**



#### Programmer

```
soup = 3
salad = 4.50
entree = 8

total = soup + salad + entree

total

15.5
```











## **NUMERICAL OPERATIONS**

Operation	Symbol	Examples (assume foo = 3 and bar = 4)			
Addition	+	3 + 3 = 6 $4 + bar = 8$	foo + bar = ?		
Subtraction	-	4 - 2 = 2 6 - bar = 2	foo – bar = -1		
Multiplication	*	3 * 2 = 6 3 * foo = 9	foo * bar = 12		
Division	/	12/3 = 4 16/bar = 4	foo / foo = 1		











## BOOLEAN (COMPARISON) OPERATIONS

Operation	Symbol	Examples (assume <i>foo</i> = 3, <i>bar</i> = 4)			
Greater Than	>	2 > 5	False	5 > 2	True
Greater Than or Equal To	>=	foo >= 3	True	bar >= foo	True
Less Than	<	2 < 5	True	5 < 2	False
Less Than or Equal To	<=	foo <= 3	True	bar <= foo	False
Equal To	Varies by language	3 equal to 3	True	"cat" equal to "dog"	False
Not Equal To	Varies by language	3 not equal to foo	False	"cat" not equal to "dog"	True













### BREAKOUT ROOMS

You will need the 1.3.1 Activity document and the Excel program to get started.









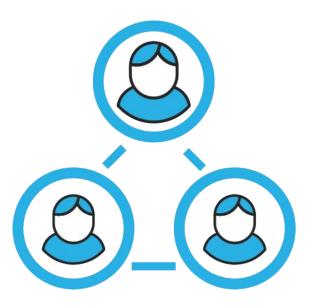






## INTERVIEW QUESTION!

What is a variable?











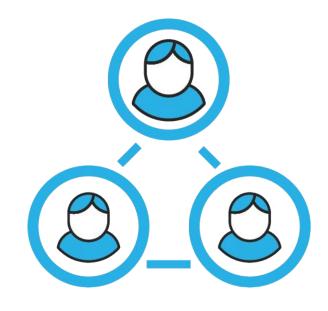


## INTERVIEW FEEDBACK.

#### What is a variable?

A variable is a placeholder for an expression or quantity.

You assign a variable a name to reflect the data it holds.





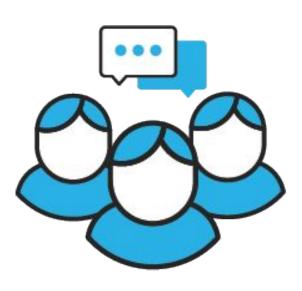
























### BREAK TIME























## LESSON OUTLINE

- Review
- Define a function
- Inputs and outputs
- Components of functions











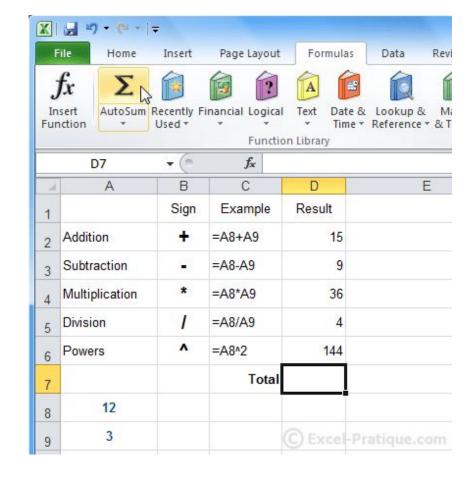


### DISCUSSION TIME

In this example, we have two numbers in cells A8 and A9.

Excel allows us to perform calculations using operations such as addition, subtraction, etc.

How might this change if we had a range of numbers to add, subtract, etc.? What would this look like?















## WHAT ARE THE GOALS?

- Identify and describe the components of a function such as its name, parameter, and outputs.
- Demonstrate knowledge of Excel functions by summarizing their purpose.

## WHY ARE THEY IMPORTANT?

There will be many instances when we will take variables and pass them to a function to manipulate and process data, returning new output back to us.











## **REVIEW**

You have learned about:

- Variables
- Conditions and statements

Today, we'll examine how functions can make your life a lot easier when programming!













## WHAT IS A FUNCTION?

A function is a name block of reusable code that will perform a specific task.

Functions contain instructions that will transform inputs and create an output.







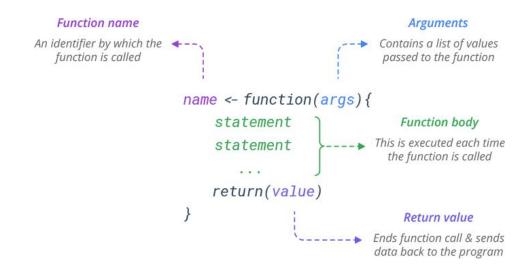






## **COMPONENTS OF FUNCTIONS**

- A unique name
- Parameters (i.e., the inputs)
- One or more instructions (called the body of the function or an algorithm)
- A return value (i.e., the output)







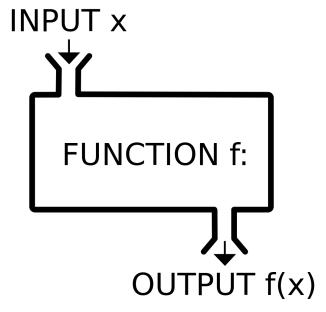






## VISUAL REPRESENTATION

Given an input of X, we want our function, f, to perform some operation on X and transform the data to output f(x).



From: MonoVektor





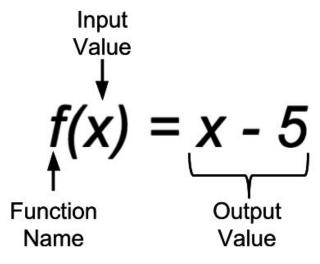






## TEST A FUNCTION

- What happens if we pass 6 to this function?
   What's the expected output?
- What happens if we pass 10 to this function? What's the expected output?



From: Flexbooks.ck12.org





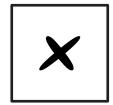












- 1. Which of the following is *not* part of a function definition?
- a) Function name
- b) Function trailer
- c) Function parameters
- d) Function return value
- 2. One or more instructions are included in the body of a function. What is another name for this series of instructions?
- a) A function return type
- b) A function header
- c) An algorithm
- d) A function parameter











## QUIZ FEEDBACK

- 1. Which of the following is *not* part of a function definition?
- a) Function name
- b) Function trailer
- c) Function parameters
- d) Function return value
- 2. One or more instructions are included in the body of a function. What is another name for this series of instructions?
- a) A function return type
- b) A function header
- c) An algorithm
- d) A function parameter









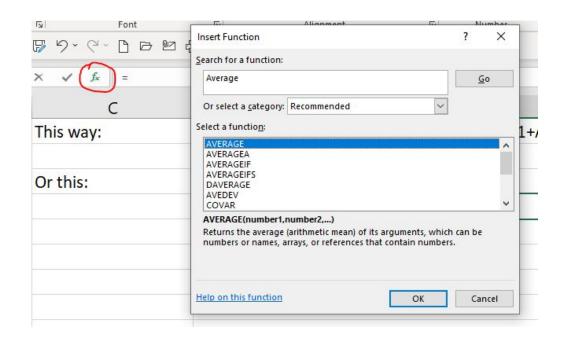




# **EXCEL FUNCTIONS**

In Excel, each function has its own name and performs a unique task.

You can see all the functions Excel has to offer by clicking the **fx** button.













## **ADDITION**

We can add a series of numbers in D1 using the addition (+) operator

OR

Use the SUM function to add the range of numbers.

D4	04 * : X * fx						
4	Α	В	С	D			
1	12		This way:	=A1+A2+A3+A4+A5+A6+A7+A8+A9+A10+A11+A12+A13+A14+A15			
2	16						
3	12		Or this:	=SUM(A1:A15)			
4	10						
5	15						
6	30						
7	29						
8	39						
9	29						
10	16						
11	8						
12	7						
13	9						
14	12						
15	15						











## **INPUTS**

To use a function, you can go into a cell and press = and type the function name (e.g., SUM) with an open parenthesis.

Functions accept inputs.

For example, the SUM function accepts numbers as inputs or a range of cells.

Or this:	=SUM(A1:A15)
	=SUM(
	SUM(number1, [number2],)











## DISCUSSION TIME

Review the following functions and write a description of what you would expect as an output.

What do you think they do?

- COUNT()
- AVERAGE()
- MIN()







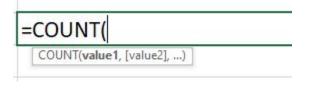


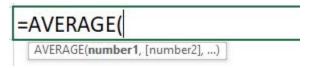




# DISCUSSION FEEDBACK.

- COUNT() gives the count of the number of cells in a range that contains numbers.
- AVERAGE() gives the arithmetic mean of a list or range of numbers.
- MIN() gives the minimum value of a list or range of numbers.

















## BREAKOUT ROOMS

You will need the 1.3.2 Activity document to get started.





















**True or False:** A function is reusable.











## QUIZ FEEDBACK.

**True:** A function is reusable.

Once you write a function, you can use it in different parts of your code.

This allows us to write more efficient code by instituting a function any time we need it, as opposed to manually writing out a process every single time.





























## BREAK TIME























## LESSON OUTLINE

- Objects
- Computing apps
- Databases











#### **Scenario**

There are six variables for a coffee shop on the right.

What happens if we add more items to the list? How would we keep track?

price\_latte = 4

price\_mocha = 4.25

price\_coffee = 2

num\_latte = 10

num\_mocha = 7

num\_coffee = 12

















## WHAT ARE THE GOALS?

- Define objects and classes.
- Explain how objects store and classify data.
- Organize object attributes into a database.

## WHY ARE THEY IMPORTANT?

Objects organize variables into attributes, giving them better context. We can look at the relationship between objects to provide context for our analysis.











## WHAT IS AN OBJECT?















## WHAT IS A CLASS?

#### **INSTANCES**

#### CLASS

#### **Pets**

name: legs: color: age:



#### Dog

name: Poppy

legs: 4

color: black age: 1 year

#### Rabbit

name: Thumper

legs: 4

color: gray age: 2 years

#### Fish

name: Goldie

legs: 0

color: yellow age: 2 months

#### Chicken

name: Little

legs: 2

color: brown age: 6 months











# WHAT'S IN MY POCKET?

Create an object and class from nearby everyday objects.

Class: Person	Object: JohnDoe
Name: Age: Height: Has_Tattoos:	Name: John Doe Age: 27 Height: 5'10" Has_Tattoos: Yes

Class: Fruit	Object: Orange
Color:	Color: Orange
Type:	Type: Citrus
Shape:	Shape: Round



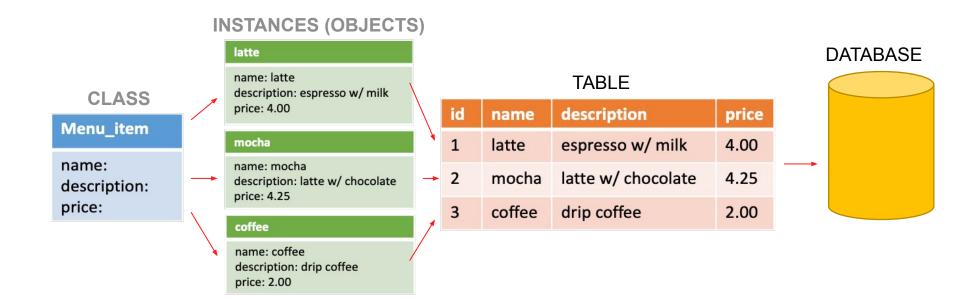








## **CONNECTING DATA**





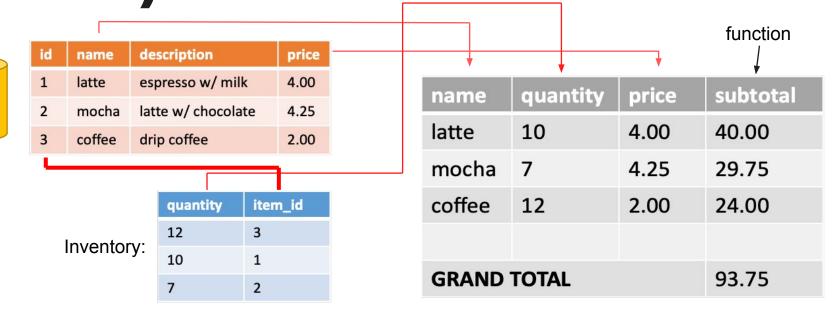








CONNECTING DATA (CONT.)













## WHAT IS A GUI?



From: Twitter

#### tweet

user: @TheEllenShow

user\_name: Ellen Degeneres

tweet: "If only Bradley's arm was longer. Best photo

ever. #oscars"

photo\_link: s3://someurl/oscars.jpg

hasthtags: ["oscars"]

timestamp: 11:06 AM 03-Mar-2014

likes: 2180282

retweets: 3327463











## DISCUSSION TIME

Explore the graphical user interface of a social media application to see interactions between objects and make assumptions about how these objects are organized and stored.



















#### Which of the following is true?

- a) An object is a blueprint for instances. An instance made from this object is called a class.
- b) An object is a blueprint for classes. A class made from this object is called an instance.
- c) A class is a blueprint for objects. An object made from this class is called an instance.
- d) A class is a blueprint for instances. An instance made from this class is called an object.











## QUIZ FEEDBACK.

#### Which of the following is true?

- An object is a blueprint for instances. An instance made from this object is called a class.
- b) An object is a blueprint for classes. A class made from this object is called an instance.
- c) A class is a blueprint for objects. An object made from this class is called an instance.
- d) A class is a blueprint for instances. An instance made from this class is called an object.













# REVIEW AND WRAP-UP

#### Today, you learned about:

- Variables and how we can use them to store and manipulate data
- The components that make up a function and how they behave
- The definition of objects and classes and how they relate to each other







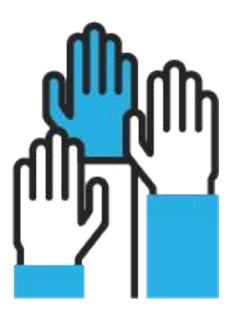






## YOUR TAKE

- Reflect on what you have learned so far.
- Share key takeaways.

























## **NEXT STEPS**



**Assigned Activities** 



Reminders

