Excel basics

Expressions

Expression	excel
A+b	=A+B
A-b	=A-B
-A	=-A
A*B	=A*B
A/B	=A/B
A% of B	=A%*B
A^B	=A^B

Order or operation

Operation	operator
Parentheses	()
negative	-
percent	%
exponentiation	^
Multiplication division	* Or /
Add and subtract	+ or -
compare	<,>,<=,>=

Labeling Data

Use descriptive names
Include units

Cell reference

- column (letter)
- Row(number)

Reference value in A3 =A3
Relative- as you copy the formula both the column and row have the ability to change (ex A3)
Absolute-as the formula is copies the neither the row or column of the reference cell changes (ex A3)
Mixed- as the formula is copied only one of the values (row or column) is allowed to change (ex A$3 or $A3)
place dollar sign inform of the one the needs to be locked

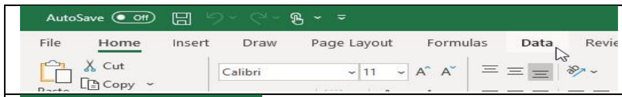
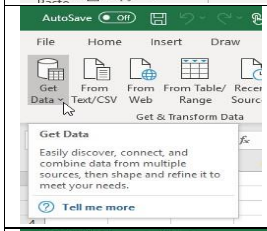
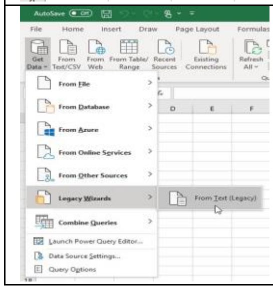
Functions

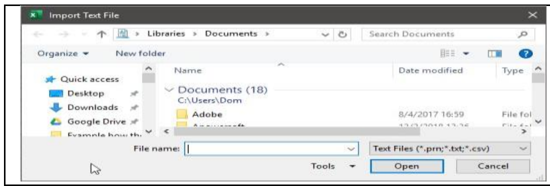
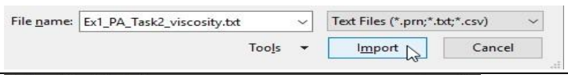
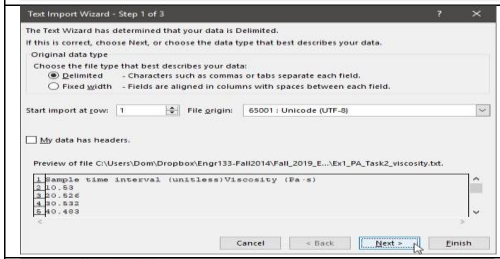
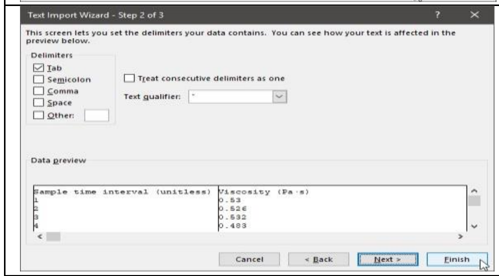
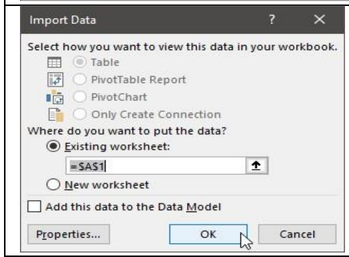
Sin cos and tan are radian values
Natural log is LN

Import data

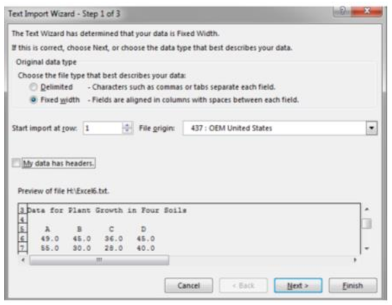
Delimited- separated by character (must specify what its separated by
Fixed- separated by a fixed distance
Specify which cell you start at

1.	Delimited file- .txt
2.	Data file- .dat
3.	Comma separated file- .csv

 AutoSave Off File Home Insert Draw Page Layout Formulas Data Revie Cut Copy Calibri 11	Click on "Data" in the ribbon on top.
 AutoSave Off File Home Insert Draw Get Data From Text/CSV From Web From Table/ Range Recen Sourc Get & Transform Data Get Data Tell me more Easily discover, connect, and combine data from multiple sources, then shape and refine it to meet your needs.	Click on "Get Data" under the Data tab.
 From File From Database From Azure From Online Services From Other Sources Legacy Wizards From Text (Legacy) Combine Queries Launch Power Query Editor... Data Source Settings... Query Options	Click on "From Text (Legacy)" under "Legacy Wizards".

 Import Text File Libraries Documents Search Documents Organize New folder Quick access Desktop Downloads Google Drive Name Date modified Type Documents (18) C:\Users\Dom Adobe 8/4/2017 16:59 File name: Text Files (*.prn;*.txt;*.csv) Tools Open Cancel	Select your data file.
 File name: Ex1_PA_Task2_viscosity.txt; Text Files (*.prn;*.txt;*.csv)	Once you have the data file, select "Import".
 Text Import Wizard - Step 1 of 3. The Text Wizard has determined that your data is Delimited. If this is correct, choose Next, or choose the data type that best describes your data. Original data type. Choose the file type that best describes your data: [✓ Delimited] - Characters such as commas or tabs separate each field. Fixed width - Fields are aligned in columns with spaces between each field. Start import at row: 1 File origin: 65001 : Unicode (UTF-8). My data has headers. Preview of file C:\Users\Dom\Dropbox\Engr133-Fall2014\Fall_2019_E...\Ex1_PA_Task2_viscosity.txt. `1 Sample time interval (unitless)Viscosity (Pa·s)` `2 10.53` `3 20.526` `4 30.532` `5 40.483`	Select the options you want – "Delimited" is the one you usually want. Then click "Next".
 Text Import Wizard - Step 2 of 3. This screen lets you set the delimiters your data contains. You can see how your text is affected in the preview below. Delimiters: [✓ Tab] Semicolon Comma Space Other: Treat consecutive delimiters as one. Text qualifier: ". Data preview: `Sample time interval (unitless) Viscosity (Pa·s)` `1 0.53` `2 0.526` `3 0.532` `4 0.483`	Select the appropriate delimiter. Then hit "Finish".
 Import Data. Select how you want to view this data in your workbook. [✓ Table] PivotTable Report PivotChart Only Create Connection. Where do you want to put the data? [✓ Existing worksheet:] =A1 New worksheet. Add this data to the Data Model	Tell Excel where it should paste your data. Click "OK".

If you want to import the following data with no headers (numbers only), the field following "Start input at row" should contain what number?-6


Text Import Wizard - Step 1 of 3

The Text Wizard has determined that your data is Fixed Width.

If this is correct, choose Next, or choose the data type that best describes your data.

Original data type

Choose the file type that best describes your data:

( ) Delimited - Characters such as commas or tabs separate each field.

[✓ Fixed width] - Fields are aligned in columns with spaces between each field.

Start import at row: 1 File origin: 437 : OEM United States

My data has headers.

Preview of file H:\Excel6.txt.

```
3 Data for Plant Growth in Four Soils
4
5    A      B      C      D
6   49.0   45.0   36.0   45.0
7   55.0   30.0   28.0   40.0
```

Charts

Dependent variable – y axis

Independent variable x axis

Scatter- experimental data, mathematical phenomena, relationships among data sets

Line-evenly spaced or sample data, time series data taken at even intervals, categorical data

Chart needs title, labels for x and y axis, x and y titles (with labels) if legend isn't needed delete it

Descriptive Statistics

Mean median mode- Which of the following are measures of central tendency?

Average- arithmetic mean of data set

Media- middle number

Mode- number that occurs most frequently

Variance-average of squared difference from mean (unit ^2)

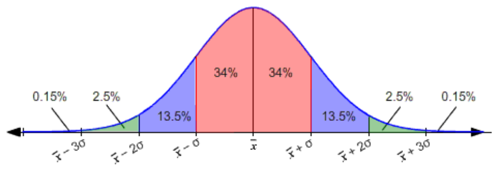
Stdev.s()- returns standard deviation of set of values (variance square root

Max

Min

Range- max-min

Count (unitless)



Histogram- cant be done on mac

type of chart that uses vertical columns to show distribution of data

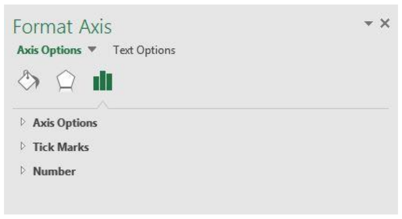
Columns of histograms are bins

Numbers on x axis are the bins upper value

Number of bins = round(sqrt(# ` data points)) rounded to nearest whole number

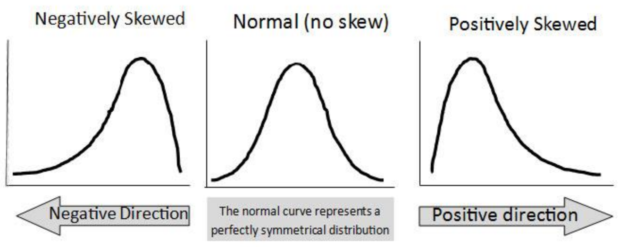
Bin width=data range/# of bins

Which part of the Format Axis dialog box below you should click to set custom bin ranges on your histogram?- axis options

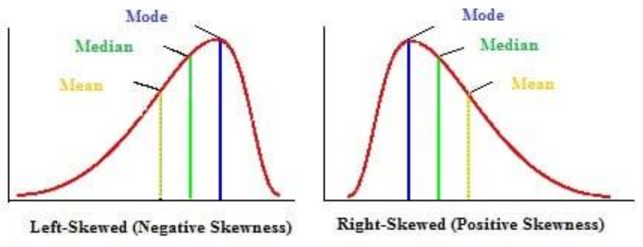


Specify your own bins #
Start at min value and continue going up by the width

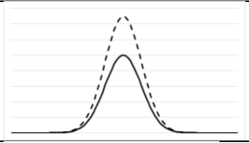
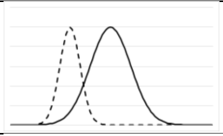
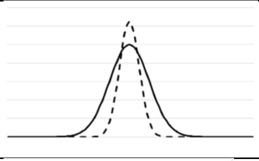
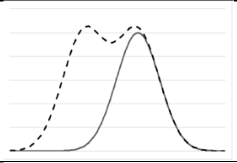
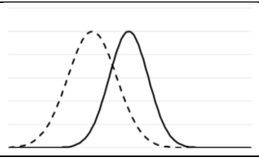
Title: Histogram of x axis
Y Axis: frequency
Excel skew function

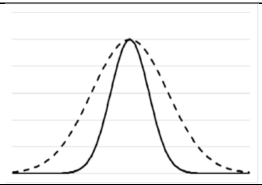


Normal distribution- mean and median are the same


Mode
Median
Mean
Mode
Median
Mean
Left-Skewed (Negative Skewness)
Right-Skewed (Positive Skewness)

As a Biomedical Engineer, your team is working on the delivery of a new cold vaccine. With FDA approval, 150 participants test the drug. A side effect of the vaccine is that they begin to crave fruit. The baseline samples provided data on the time until patient reported feeling side effect [in hours] that is distributed as shown in the solid line in the graphs below. Some modifications were made to the vaccine or its delivery, which caused a shift in the distribution, resulting in a modified distribution (a dashed line). For each scenario, identify the graph that best represents the modified curve that would result.

	You conduct an extended trial using the same delivery system as in the first test (injection) expanding the number of participants in the study from 150 to 225.
	The initial vaccine delivery was by injection. A new delivery method, in the form of a dermal patch. The result is a decrease in the time until the hunger cravings begin for all participants with less variation among patients.
	You combine another drug with the vaccine to decrease hunger cravings. The drug delays early cravings for some people, but an interaction effect causes cravings to occur sooner in people who before experienced late cravings. The result is a decrease in the variability of the time of onset of the side effect
	A new trial uses the same drug delivery system as in the first test but includes the 150 original participants plus 150 new participants who tend to exhibit the onset of cravings much sooner.
	A more potent dose is given to the same group of 150 participants. The result is a decrease in the time until the hunger cravings begin for all participants with similar variation.

	A new trial uses the same drug delivery system as in the first test but uses a generic manufacturer which has the same average response but increases the variability in the reactions.

Teaming

Adjourning	Team members are happy they are finished but sad to see the team break up.
Performing	Team members are highly motivated and work through problems.
Storming	Team members argue about roles and do not get much work done
Norming	Team members are friendly and trusting, increasing productivity and decreasing conflict.
Forming	Team members are very polite and talk about project objectives.

Code of Cooperation	Some team members often arrive late for meetings while two others sometimes have side conversations.
Action Items	The project is getting behind schedule because no one really knows who's responsible for certain things.
Agenda	In the last two meetings, unimportant items were discussed but items that needed to be completed were not.
Issues Bin	Often during group discussions, team members bring up important issues that are then forgotten.

Timekeeper and coordinator- A team has implemented all of the tools they can find to be more successful but are still falling behind in their project. They often end the meeting without finishing the last few items on the agenda and argue over where and when the next meeting will be.
True- A professional engineer is likely to use teaming skills often.
False- If you are your team's Coordinator, it is your job to be the boss of the team.
A team of engineers is designing a new type of long-life cell phone battery. Which of the following are likely to occur if the team does not make evidence-based decisions during the course of the project?

- Time and money will be wasted
- The solution or designed product may not be solved
- The engineers reputation or standing in the workplace may be damaged

Python

String

In “” the character “ cannot be used
In ‘’ the character ‘ cannot be used
// divide and floor

\' or \"	This escape character allows you to manually push apostrophes and double quotes through without accidentally signaling the end of the string
\t	Tab (inserts a few spaces into the string)
\n	New line (like pressing enter in a word document)

the letter 'r' followed by the 'string' Example: print(r'home\user\newDoc')	This negates certain whitespace characters like \n and \t, printing everything contained in the quotes as it is written

Lists

Assuming a list has been created with the name "things," match each of the list methods or functions with their purpose

len(things)	Returns the number of items in the list
things.index(x)	Returns the location of the first item with the value “x”
things.clear(x)	Produces an error (specifically, a “TypeError”)
things.insert(I, x)	Adds item “x” at index location “I” and shifts the following items up in index locations
things.reverse()	Reverses the order of items in the list
things.append(x)	Adds item “x” to the end of the list
things.sort(reverse=True)	Arranges the list either numerically or alphabetically in reverse order
things.clear()	Deletes all items from the list
things.pop(i)	Removes and returns from list the item at index “i.” If no index is provided, it removes and returns the last item in the list.
things.remove(x)	Deletes from list the first item with value “x,” but returns an error if no items have value “x”
things.count(x)	Returns the number of times that the value “x” appears in the list

Importing

Import (name) as USABLE NAME
Import name
Import name as *

Math

False- import math (ceil, floor)
True- from math import ceil, floor
True- from math import floor, ceil
False- import math (floor, ceil)

math.cos(x)	Return the cosine of x radians.
math.radians(x)	Convert angle x from degrees to radians.
math.sqrt(x)	Return the square root of x.
math.modf(x)	Return the fractional and integer parts of x. Both results carry the sign of x and are floats.
math.ceil(x)	Return the ceiling of x, the smallest integer greater than or equal to x.
math.isclose(a, b, *, rel_tol=1e-09, abs_tol=0.0)	Return True if the values a and b are close to each other and False otherwise. Whether or not two values are considered close is determined according to given absolute and relative tolerances.
math.exp(x)	Return e raised to the power x, where e = 2.718281… is the base of natural logarithms. This is usually more accurate than math.e ** x or pow(math.e,x).
math.isinf(x)	Return True if x is a positive or negative infinity, and False otherwise.
math.atan(x)	Return the arc tangent of x, in radians.
math.acos(x)	Return the arc cosine of x, in radians.
math.pow(x, y)	Return x raised to the power y.
math.e	The mathematical constant e = 2.718281…, to available precision.

math.isnan(x)	Return True if x is a NaN (not a number), and False otherwise.
math.degrees(x)	Convert angle x from radians to degrees.
math.fabs(x)	Return the absolute value of x.
math.log10(x)	Return the base-10 logarithm of x. This is usually more accurate than log(x,10).
math.log(x[, base])	With one argument, return the natural logarithm of x (to base e). With two arguments, return the logarithm of x to the given base, calculated as log(x)/log(base).
math.copysign(x, y)	Return a float with the magnitude (absolute value) of x but the sign of y.
math.gcd(a, b)	Return the greatest common divisor of the integers a and b. If either a or b is nonzero, then the value of gcd(a, b) is the largest positive integer that divides both a and b. gcd(0, 0) returns 0.
math.asin(x)	Return the arc sine of x, in radians.
math.tan(x)	Return the tangent of x radians.
math.factorial(x)	Return x factorial as an integer.
math.sin(x)	Return the sine of x radians.
math.isfinite(x)	Return True if x is neither an infinity nor a NaN, and False otherwise. (Note that 0.0 is considered finite.)
math.hypot(x, y)	Return the Euclidean norm, sqrt(x*x + y*y). This is the length of the vector from the origin to point (x, y).
math.pi	The mathematical constant π = 3.141592..., to available precision.
math.floor(x)	Return the floor of x, the largest integer less than or equal to x.

User Input

Unless converted user input is a string

Var= input("prompt message")

Functions

def name():

__6__

```python
def keywords(a = 1, b = 2, c = 3):
    return a + b + c

print(keywords(6, 9))
```

__8__

```python
def keywords(a = 1, b = 2, c = 3):
    return a + b + c

print(keywords(3, 6, 9, 12))
```

__2__

```python
def keywords(a = 1, b = 2, c = 3):
    return a + b + c

print(keywords(b = 5))
```

__3__

```python
def keywords(a = 1, b = 2, c = 3):
    return a + b + c

print(keywords(6))
```

__5__

```python
def keywords(a = 1, b = 2, c = 3):
    return a + b + c

print(keywords(c = 6, a = 9))
```

__1__

```python
def keywords(a = 1, b = 2, c = 3):
    return a + b + c

print(keywords())
```

__8__

```python
def keywords(a = 1, b = 2, c = 3):
    return a + b + c

print(keywords(a = 15, d = 27))
```

1. 6
2. 9
3. 11
4. 12
5. 17
6. 18
7. 19
8. Error

What is the difference between the standard "range()" function and NumPy's "arange()" function?- NumPy's "arange()" function is able to use steps smaller than
True- The main different between "continue" and "pass" is that continue skips the rest of the code in the current loop iteration whereas pass skips the remaining code in the current conditional.
False- The "break" command exits **all** currently running loops rather than simply the deepest loop currently running. In other words, if the break command is executed while in the third level of a nested loop, the code will not go exit to the second level and continue, but rather, it will exit the entire nested loop system currently engaged.

File IO

file.writelines(text)	If "text" is a simple string, the contents of "text" are written to "file." If "text" is a sequence of strings, then each element of "text" is written to "file" one after the other.
file = open('name.txt', 'r+')	Opens "name.txt" as a file object capable of reading and writing and assigns it to the variable name "file"
text = file.read(-x)	Regardless of the value of "x," this line of code reads all remaining characters from "file" and assigns them to "text"
file = open('name.txt', 'r')	Opens "name.txt" as a read-only file object and assigns it to the variable name "file"
file = open('name.txt', 'wb')	Opens "name.txt" as a write-only file object in binary mode and assigns it to the variable name "file"
text = file.readline(-x)	Regardless of the value of "x," this line of code reads the remaining characters from the current line and assigns them to "text"
text = file.read(x)	Reads the next "x" number of characters from "file" and assigns them to "text"
text = file.read()	Reads all remaining characters from "file" and assigns them to "text"
file.write(text)	Writes the contents of "text" to "file"
text = file.readlines()	Reads "file" and creates a list where each element of the list is a line from "file" in the same order that they appeared in "file." This list is assigned to the variable "text"
text = file.readline(x)	Reads the next "x" number of characters from the current line and assigns them to "text." If the value of "x" exceeds the number of remaining characters on that line, it stops reading at the end of that line and does not continue to the next line.
text = file.readline()	Reads the next line from "file" **or** the remaining characters from the current line and assigns it/them to "text"
file = open('name.txt', 'w')	Opens "name.txt" as a write-only file object and assigns it to the variable name "file"
text = file.readlines(x)	The function returnes a list of strings which gets assigned to the variable `text`. Each entry of the list is a line of the file `file`. The function returns as many lines as needed to return at least `x+1` characters total. It returns a line starting where the cursor is at the moment and it always returns from the cursor to the end of the line (cursor default position is at the beginning of a line). Note: It is recommended that you experiment with this using the `Dream.txt` file provided.

```
with open('file_name.txt', 'w') as variable_name:
```

Is that it will automatically close the file when done instead of requiring you to always finish with this line:

```
variable_name.close()
```