CTEC 298-101 Spring 2021 1st 8w February11, 2021

Benjamin Dragenice

**Define the terms below.**

Jupyter Notebook – a powerful tool for interactively developing and presenting data science projects.

Cell – a container for text to be displayed in the notebook.

Kernel – computational engine that executes the code contained in a notebook document.

Ipynb File – text file that describes the contents of your notebook.

JSON – JavaScript Object Notation; open standard file format and data interchange format

Metadata – set of data that describes your notebook.

Code Cell – code to be executed in the kernel.

Markdown Cell – text formatted using Markdown and displays it output in-place.

Matplotlib Plots – lets you plot graphs using the Matplot library.

1. Save the result of (42 - 11) \* 22 to result.

**Result = (42-11) \* 22**

1. Print result.

**Print(result)**

1. Store the value 15 in a variable named a\_value.
2. Store the result of (25 - 7) \* 17 to a variable named a\_result.
3. Using the print() command, display the following:
   * The value stored in the a\_value variable.
   * The result of adding 12 to the variable a\_result.
   * The result of adding a\_value to a\_resul

**A\_value = 15**

**A\_result = ((25-7) \* 17**

**Print(a\_value)**

**Print(a\_result + 12**

**Print(a\_value + a\_result)**

1. In the code editor we already stored 34000 in a variable named old-income, and 40000 in a variable named new income — both these variable names cause syntax errors.
2. Change the variable name old-income to old\_income to prevent a syntax error.
3. Change the variable name new income to new\_income to prevent a syntax error.
4. Run the code after you changed both names.

**Old\_income = 34000**

**New\_income = 40000**

1. Update the variable income by adding 6000 to its current value. The variable income is already shown in the code editor on the right. Income = 34000
2. Print income.

**Income = income + 6000**

**Print(income)**

1. Assign a value of 20 to a variable named variable\_1.
2. Assign a value of 20 to a variable named variable\_2.
3. Update the value of variable\_2 by adding 10 to its current value. You can take advantage of the += operator.
4. Update the value of variable\_1 by multiplying its current value by 4. You can take advantage of the \*= operator.
5. Display variable\_1 and variable\_2 using print().

**Variable\_1 = 20**

**Variable\_2 = 20**

**Variable\_2 += 10**

**Variable\_1 \*= 4**

**Print(variable\_1)**

**Print(variable\_2)**

**Work with this table for the next exercise:**

|  | **track\_name** | **price** | **currency** | **rating\_count\_tot** | **user\_rating** |
| --- | --- | --- | --- | --- | --- |
| **0** | Facebook | 0.0 | USD | 2974676 | 3.5 |
| **1** | Instagram | 0.0 | USD | 2161558 | 4.5 |
| **2** | Clash of Clans | 0.0 | USD | 2130805 | 4.5 |
| **3** | Temple Run | 0.0 | USD | 1724546 | 4.5 |
| **4** | Pandora - Music & Radio | 0.0 | USD | 1126879 | 4.0 |

Data source: [Mobile App Store data set (Ramanathan Perumal)](https://www.kaggle.com/ramamet4/app-store-apple-data-set-10k-apps)

1. Store the second row ('Instagram', 0.0, 'USD', 2161558, 4.5) as a list in a variable named row\_2.
2. Store the third row ('Clash of Clans', 0.0, 'USD', 2130805, 4.5) as a list in a variable named row\_3.

**List for Tasks A - C below**

**row\_1 = ['Facebook', 0.0, 'USD', 2974676, 3.5]**

**row\_2 = ['Instagram', 0.0, 'USD', 2161558, 4.5]**

**row\_3 = ['Clash of Clans', 0.0, 'USD', 2130805, 4.5]**

**LIST A**

1. Assign the fourth element from the list row\_1 to a variable named ratings\_1. Don't forget that the indexing starts at 0.
2. Assign the fourth element from the list row\_2 to a variable named ratings\_2.
3. Assign the fourth element from the list row\_3 to a variable named ratings\_3.
4. Add the three numbers retrieved together and save the sum to a variable named total.
5. Divide the sum (now saved in the variable total) by 3 to get the average number of ratings for the first three rows. Assign the result to a variable named average.

**ratings\_1 = row\_1[3]**

**ratings\_2 = row\_2[3]**

**ratings\_3 = row\_3[3]**

**total = ratings\_1 + ratings\_2 + ratings\_3**

**average = total / 3**

**LIST B**

The last element in each list shows the average rating of each application.

Retrieve the ratings for the first three rows, and then find the average value of all the ratings retrieved.

1. Assign the last element from the list row\_1 to a variable named rating\_1. Try to take advantage of negative indexing.
2. Assign the last element from the list row\_2 to a variable named rating\_2.
3. Assign the last element from the list row\_3 to a variable named rating\_3.
4. Add the three ratings together and save the sum to a variable named total\_rating.
5. Divide the total by 3 to get the average rating. Assign the result to a variable named average\_rating.

**rating\_1 = row\_1[4]**

**rating\_2 = row\_2[4]**

**rating\_3 = row\_3[4]**

**total\_rating = rating\_1 + rating\_2 + rating\_3**

**average\_rating = total\_rating / 3**