**Presentation Notes:**

1. What are the two main parts of a computer architecture?
   1. CPU Chip
   2. RAM Memory
2. Google “basic Python commands” and list four commands.
   1. M 8++while
   2. input
   3. output
   4. print
3. Identify the two *syntax errors* in the following command: **Print("This command prints messages)**
   1. Capital P
   2. Missing quotation mark
4. Summarize the cause and effect of a *syntax error*.

Syntax errors occur when typos are made and they help fix the errors.

1. Explain what happens if you use a variable before it is defined.

Use of an undefined variable results in a **Run-Time** program error

1. Summarize the cause and effect of a *run-time* error.

The program will not run as the program does not understand what the variable means

1. Write a Python statement to assign the value of 24 to the variable classSize.

classSize = 24

1. Create a valid Python variable name to store a student exam mark and that follows the “mixedCase” style guidelines.

**examMarksOfStudents**

1. Create a valid Python variable name to store a student exam mark and that DOES NOT follow the “mixedCase” style guidelines.

exammarksOfStudents

1. Write a mathematical expression that assigns a value of 62 to the variable myAnswer.
   1. myAnswer = 30 \* 2 + 2

1. Write a mathematical expression that uses the variable aNumber and assigns a value of 77 to the variable myAnswer.
   1. aNumber = 35
   2. myAnswer = aNumber \* 2 + 7
   3. print("The answer is:",myAnswer)
2. Change the program on the last slide of the presentation to calculate and print out the cube (power 3) of an input number.

**Student Questions:**

A resource for Python Style guidelines mal be found here:

[https://www.python.org/dev/peps/pep-0008/#naming-conventions](https://www.python.org/dev/peps/pep-0008/)

1. Identify which of the following are valid Python variable names (even if they do not follow the mixedCase style guidelines).

|  |  |
| --- | --- |
|  | True / False |
| StudentNumber | True |
| 5thRow | False |
| else | True |
| break | True |
| Row\_5 | True |

1. Identify which of the following are valid Python variable names that also follow the mixedCase style guidelines.

|  |  |
| --- | --- |
|  | True / False |
| StudentNumber | false |
| studentNumber | True |
| Row | True |
| row5 | True |
| Row5 | True |

1. Summarize the difference between a *syntax error* and a *run-time* error.

The run-time error is when a program will not run as the program does not understand what the variable means and syntax errors occur when typos are made and they help fix the errors.

1. Write an expression that calculates the cost of 6 slices of pizza at 2 dollars a slice assigns the result to a variable in RAM memory. Use proper style and meaningful names for your variables.

pizzaCost = 6 \* 2

print ("Sir, that would cost" , pizzaCost)

1. Write an expression that calculates the cost of a variable number slices of pizza at 2 dollars a slice assigns the result to a variable in RAM memory. Use proper style and meaningful names for your variables.

pizzaSlice = int(input ("How many pizza slices will you like to order?"))

totalCost = pizzaSlices \* 2

print ("Sir, that would cost" , totalCost)

1. Write a program that gets the number of slices from the console input, uses your expression in #5 above, and prints out the result to the console output. Use proper style and meaningful names for your variables and meaningful messages for your input and print commands.

pizzaSlice = int(input ("How many pizza slices will you like to order?"))

totalCost = pizzaSlices \* 2

print ("Sir, that would cost" , totalCost)

1. Extend your program in #6 above to also calculate and print out the number of boxes of pizza if each box contains 8 slices.

pizzaSlices = int(input ("How many pizza slices will you like to order?"))

totalCost = pizzaSlices \* 2

amountOfBoxes = int(input ("How many pizza boxes will you like?"))

print ("Sir, that would cost" , totalCost , "dollars for" , amountOfBoxes, "pizza boxes")