**Presentation Notes:**

1. What are the two main parts of a computer architecture?
   1. CPU Processor
   2. RAM Memory
2. Google “basic Python commands” and list four commands.
   1. print
   2. if
   3. elif
   4. else
3. Identify the two *syntax errors* in the following command: **Print("This command prints messages)**
   1. Capital P
   2. Only one quotation mark
4. Summarize the cause and effect of a *syntax error*.

The cause of a syntax error is a problem in the formatting of the commands

It prevents a program from running and the error is displayed in the editor window

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

You get a Run-Time program error

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

A runtime Error is caused by a variable being run before it is defined

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.

examMark1

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

studentexammark

1. Write a mathematical expression that assigns a value of 62 to the variable myAnswer.
   1. myAnswer = (3 \*2) + 3

1. Write a mathematical expression that uses the variable aNumber and assigns a value of 77 to the variable myAnswer.
   1. aNumber = 154
   2. myAnswer = aNumber / 2
2. Change the program on the last slide of the presentation to calculate and print out the cube (power 3) of an input number.

value = int(input("Enter a number:"))

value2 = value \*\* 3

print("The cube of %d is %d" % (value,value2))

**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 | False |
| break | False |
| 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 | False |

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

A runtime error is when a variable is run before it is defined whereas a syntax error is an error in the formatting of your code.

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.

numberOfSlices = 6

costPerSlice = 2

totalCost = numberOfSlices \* costPerSlice

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.

numberOfSlices = 5

costPerSlice = 2

numberOfSlices \* costPerSlice = totalCost

print (“That will be $”,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.
2. numberOfSlices = int(input("How many slices would you like?"))
3. costPerSlice = 2
4. totalCost = numberOfSlices \* costPerSlice
5. print("That will be $%d" % (totalCost))
6. Extend your program in #6 above to also calculate and print out the number of boxes of pizza if each box contains 8 slices.
7. numberOfSlices = int(input("How many slices would you like?"))
8. costPerSlice = 2
9. totalCost = numberOfSlices \* costPerSlice
11. numberOfBoxes = numberOfSlices / 8
12. if (numberOfSlices % 8 == 0):
13. if (numberOfSlices >= 16):
14. print ("That's %d boxes." % (numberOfBoxes))
15. if (numberOfSlices == 8):
16. print ("That's 1 box." )
17. slicesModEight = numberOfSlices % 8
18. if (numberOfSlices % 8 != 0 and numberOfSlices >= 8):
19. if (numberOfSlices >= 16):
20. if (slicesModEight == 1):
21. print ("That's %d boxes and %d slice." % (numberOfBoxes, slicesModEight))
22. if (slicesModEight > 1):
23. print ("That's %d box and %d slices." % (numberOfBoxes, slicesModEight))

26. if (numberOfSlices >= 8):
27. if (numberOfSlices <=15):
28. if (slicesModEight == 1):
29. print ("That's 1 box and %d slice." % (slicesModEight))
30. if (slicesModEight > 1):
31. print ("That's 1 box and %d slices." % (slicesModEight))
32. print("That will be $%d" % (totalCost))
33. print(slicesModEight)