**Python Variables Tasks:**

* Define a variable of type bool :
* It can be either TRUE OR FALSE
* What Is the benefit of the type method
* It provides information about the variable nature. For example: type()-🡪 can be string, bool,list,…etc.
* Can we use semi colon ; with python
* Yes🡪 in the case of continuing the syntax in another line
* Python is interpreted or compiled
* Python is interpreted language
* What is the differences between low level & high level
* It refers to low- and high-level languages. The most important difference is that low level languages are coded to one machine and cannot be transferred easily to a different machine. On the other hand, high level languages (i.e. Python) are easily transferable and shared with others.

**Python Operators Tasks:**

* What is the differences between = , ==
* =: means equal
* ==: means check if the variable is equal to certain value
* What do we mean by using !=
* != means not equal to
* What is the operator precedence
* It is the way Python process calculations in order; for example, division and multiplication precedes addition and subtractions in calculation sequence

**Python Conditions Tasks:**

* In which cases we will use all
  + In the case of using multiple “and” in a syntax and its used for simplification
* What is the differences between all , and
  + Both can do the same function-all is preferable when there is a series of and’s
* What is the differences between any , or
  + Both can do the same function, yet any is more used when there is a series of OR’s
* If we need all the conditions to be true we will use ….
  + ALL()
* What is the differences between if , elif
* If we use “if” we don’t need else to execute the code but in another case, we use “if” and “elif” then we will have to complete the code with “else” to provide the default.
* What is the differences between elif else
* Can we use more than one elif
  + Yes-can use as many as we need for the syntax
* Can we use more than one else
  + No-only one can be used --🡪 the default is only one