Unit 3: Conditions, if and while — Application: FizzBuzz

In this Application you will write a program to illustrate the amazing 'FizzBuzz' phenomenon.

Objectives:

- Make a while loop that terminates in two ways
- Define a function that examines numbers for 'divisibility'

FizzBuzz

Write a program to print some of the natural numbers (1,2,3,…) but…

If the number is divisible by 3, print "Fizz" instead.
If the number is divisible by 5, print "Buzz" instead.
If the number is divisible by both 3 and 5, print "FizzBuzz" instead.
The program should end when the esc key is pressed.

'Divisibility' is tested using the % operator (mod or remainder).

There is one other form of the **if** statement you will find useful:

```
if <condition>:
    <true block1>
elif <another condition>:      there can be many elif blocks
    <true block2>              elif is short for 'else if'
<more elifs?>
else:
    <false block>      this block is processed when none of the others are true
```

1. Begin a new Python 'blank program' file.

   The **get_key()** is going to be used so add

   from ti_system import *

   found on **menu > More Modules > TI System**.

2. Set up a variable that represents the counting numbers starting with 1.

3. Use the **while get_key()!=0:** structure found on **menu > More Modules > TI System**.

4. Use the **if..elif..else** structure found on **menu > Built-ins > Control**.

Remember, it works like this:

if *<this is true>*:

<do this>

elif *<this is true>*: *(stands for 'else if…')*

<do this>

(there may be more **elif***s in here)*

else:

<do this> *(when none of the others are true)*

Caution: Remember to use ***==*** *when writing a condition, not* ***=****. Using the wrong symbol will result in a syntax error. if* ***x==5:****, not if* ***x=5:****.*

Using the ***ctrl+=*** *menu can help.*

5. Here is a peek at the **if..elif..else** structure but some information has been omitted.
The **if** and each **elif** has a *condition* to be met and a block to be processed when the condition is true. **else**: has no condition and its block will be processed when *none* of the others are true.

counter+=1 is shorthand for **counter = counter + 1**.

Hint: Recall that **A % B** gives the *remainder* when A is divided by B.

6. A sample run of the program might look like the image to the right.
   If your numbers go by too fast, use a **sleep()** function at the bottom to slow things down.
   sleep() is in the time module so be sure to include:

   from time import sleep

   What is the first number replaced by 'FizzBuzz'?