**Review**

This chapter will review all the materials that have been provided.

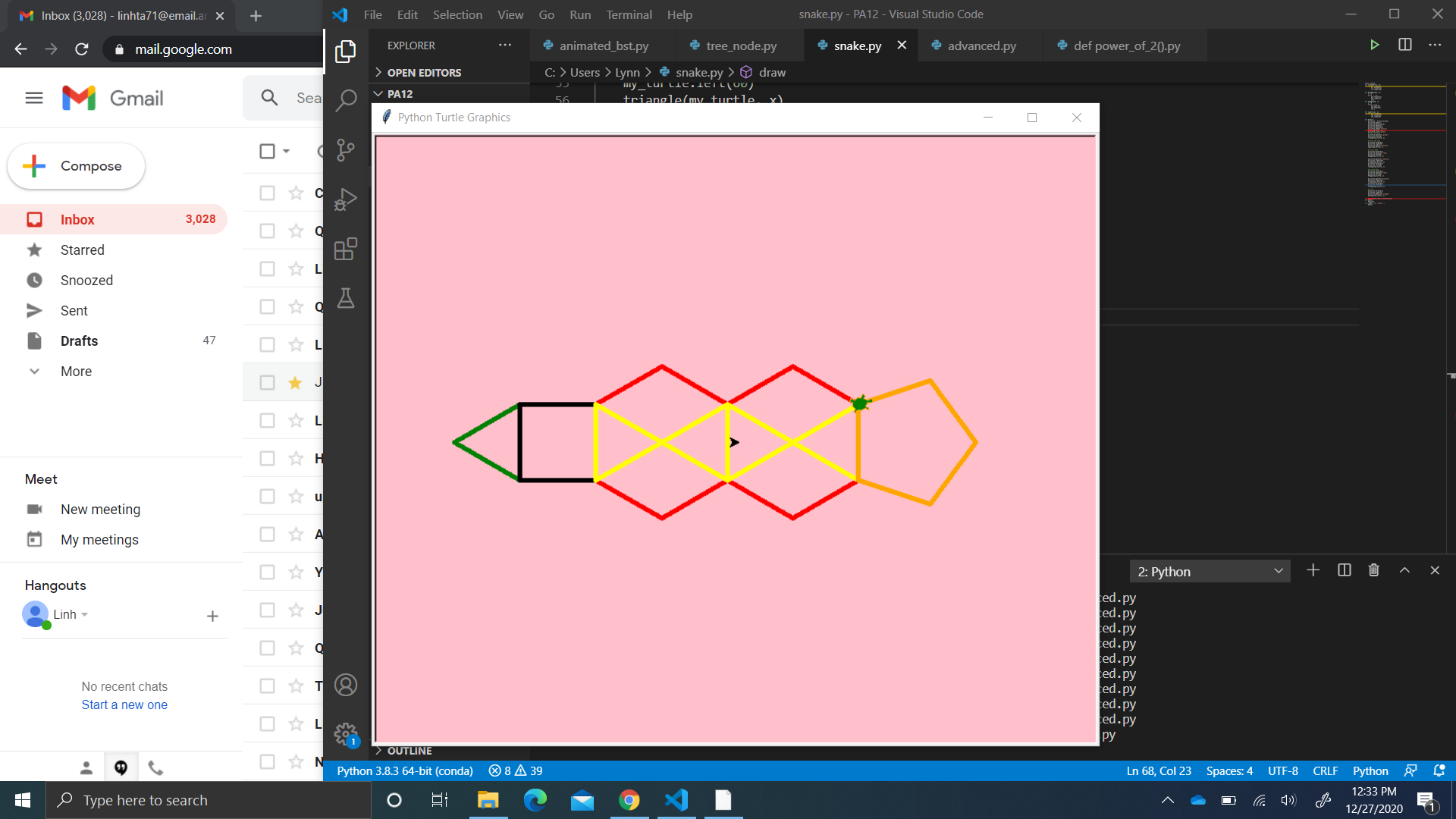
* Summary of what you have learnt.

A picture containing table

Description automatically generated

|  |  |
| --- | --- |
| **Purpose** | **Command** |
| Import Python turtle graphic | import turtle |
| Assign a name for your turtle / Create a turtle | name = turtle.Turtle() |
| Choosing a shape.  *if you skip the command choosing a shape for your turtle then your turtle shape will automatically be an arrow.* | name.shape(‘shape’)  *You have to replace ‘shape’ by one of these shapes: arrow, circle, square, triangle or turtle.* |
| Choosing a color  *if you skip the command choosing a color for your turtle then your turtle color will automatically be black.* | name.color(‘color’)  *You have to replace ‘color’ by the color that you like.* |
| Setting a default location to (0,0) | name.goto(0,0) |
| Setting a background  *If you don’t want to have background image for your turtle, please skip this part.* | |
| Call the screen function. | screen=turtle.Screen(). |
| Set background to the picture that you like, | screen.bgpic(“picture\_file”)  *Click to the image icon and look at 3 images that it contains. Replace ‘picture\_file’ with the image file that you like.* |
| Make the turtle goes forward in the current direction | turtle.forward(step)s |
| Turns the turtle to the left direction in number degrees | turtle.left(degrees) |
| Turns the turtle to the right direction in number degrees. | turtle.left(degrees) |
| Pointing a variable to an object. | variable = object  *The object can be in any type: integer, decimal, list, etc.* |
| Setting turtle pen to a color | turtle. pencolor('color')  *Replace ‘color’ by a color that you want.* |
| Make the turtle goes backward in the current direction | turtle.backward(steps)  *Replace “steps” by an integer.* |
| Draw a semi circle | turtle.circle(radius, 180)  *Replace radius by either a decimal or an integer. 180 is fixed for SEMI-CIRCLE only.* |
| Draw a circle | turtle.circle(radius)  *Replace radius by either a decimal or an integer.* |

In this chapter, you will have only one long task to do, and you also need to perform some calculations.

Task: draw a snake.

You will need to write:

* 2 while loops to draw the pentagon and the hexagon.
* 2 for loop to draw the square and the triangle.

**Instruction:**

* **Step 1: set up your turtle:**
  + Name your turtle.
  + Pen up your turtle.
  + Set the location of your turtle to -200 and 0.
  + Pen down your turtle.
  + Pick a length for all of your edges by using the similar syntax when you did for the square in the chapter 3. Notice that all the edges in the following shapes: the triangle, the pentagon, the hexagon and the square are the same. Please take the length of 50.
* **Step 2: draw triangle head.** 
  + Set your turtle pen color to green
  + Move your turtle to the left for 30 degrees.
  + Write the for loop to draw the triangle:
    - There are 3 edges in the triangle, therefore, “i” here should be 3
    - The code chunk should contain 2 lines of code.
      * Make your turtle goes forward by the length you have set up in step 1.
      * Turn your turtle to the right for 120 degrees.
* **Step 3: Draw the neck of the snake (the black square):**
  + Make your turtle goes forward by the length you have set up in step 1.
  + Turn your turtle to the right for 30 degrees.
  + Change your turtle pen color to black.
  + Write the for loop to draw the square:
    - There are 4 edges in the square, therefore, “i” here should be 4.
    - The code chunk should contain 2 lines of code.
      * Make your turtle goes forward by the length you have set up in step 1.
      * Turn your turtle to the right for 90 degrees.
* **Step 4: Draw the first hexagon along with triangle features:**
  + Make your turtle goes forward by the length you have set up in step 1.
  + Change your turtle pen color to red.
  + Turn your turtle to the left for 90 degrees.
  + Use while loop to draw hexagon:
    - “a” should start at 0.
    - “c” should be 6 since there are 6 edges in the hexagon.
    - There should be 3 lines of code below the while command:
      * Turn your turtle to the right for 60 degrees.
      * Make your turtle goes forward by the length you have set up in step 1.
      * Increase “n” by 1, which means “b” = 1.
  + Change your turtle pen color to yellow.
  + Turn your turtle to the right for 120 degrees.
  + *Repeat the for loop that draw the triangle in step 2.*
  + Make your turtle goes forward by the length you have set up in step 1.
  + Turn your turtle to the left for 60 degrees.
  + *Repeat the for loop that draw the triangle in step 2.*
* **Step 5: Draw the second hexagon along with triangle features:**
  + Make your turtle goes forward by the length you have set up in step 1.
  + Change your turtle pen color to red.
  + Turn your turtle to the left for 60 degrees.
  + *Repeat the while loop that draws the hexagon in step 4.*
  + Change your turtle pen color to yellow.
  + Turn your turtle to the right for 120 degrees.
  + *Repeat the for loop that draw the triangle in step 2.*
  + Make your turtle goes forward by the length you have set up in step 1.
  + Turn your turtle to the left for 60 degrees.
  + *Repeat the for loop that draw the triangle in step 2.*
* **Step 6: Draw the tail of the snake (which is the orange pentagon):**
  + Make your turtle goes forward by the length you have set up in step 1.
  + Turn your turtle to the right for 12 degrees.
  + Change your turtle pen color to orange.
  + Use while loop to draw the pentagon:
    - “a” should start at 0.
    - “c” should be 5 since there are 5 edges in the hexagon.
    - There should be 3 lines of code below the while command:
      * Make your turtle goes forward by the length you have set up in step 1.
      * Turn your turtle to the right for 72 degrees.
      * Increase “n” by 1, which means “b” = 1.