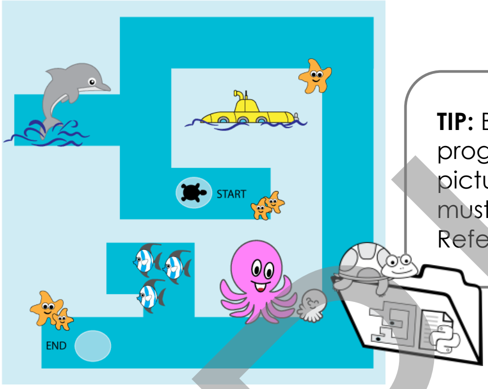
Assignment 9 Move Through the Maze

Programmers have a goal when they design a program. They know what they want the outcome to be.

You are a programmer! Your goal is to move the Turtle through the maze. You must write instructions to go from the start to the end point. The Turtle must stay inside the path!

Help the Turtle swim home.



This image is in the maze folder.

TIP: Before you start to program, the background picture and Python program must be in the SAME FOLDER! Refer to Step 1 and 2.

Copy the Maze Folder

The maze is a picture. You will add it to the background using `bgpic("picture.gif" )`. To have it display, the picture MUST BE in the same place as the maze Python file.

1. ▷ Open the *Turtle* folder. (Ask your teacher where this is located.)

   ▷ Copy the *maze* folder and paste it where you save your work.

Save a Python File into the Maze Folder

2. ▷ Open IDLE (Python).

   ▷ From the File menu, select *New File*.

   ▷ From the File menu, select *Save*.

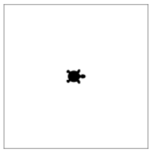
   ▷ Go the *maze folder* you copied in Step 1.
   Type **swim** as the file name. Click *Save*.

Start the Program

3. ▷ Import the Turtle library. Set the speed to slow. Make the symbol look like a Turtle:

```
from turtle import *
speed("slowest")
shape("turtle")
```

   ▷ Apply your skills to run the program.
   A Turtle should be on the canvas.

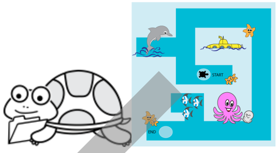


Apply a Background Picture to the Canvas

4. ▷ Add the maze image to the background using the command `bgpic("maze.gif")`:

```
from turtle import *
speed("slowest")
shape("turtle")
bgpic("maze.gif")
```

If you cannot see the maze, the Python file and picture are not in the same folder.



▷ Apply your skills to run the program.
The Turtle will be at the start of the maze.

Use Trial and Error as a Debugging Strategy

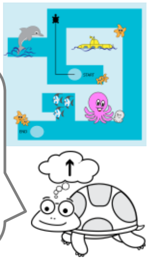
5. When you run a program, the result might not be what you want. One method to fix the problem is *Trial and Error*.

Trial and Error means that you test different ideas until you find one that works. After each *trial* you study the *error* you get. This lets you come up with a better idea to try next.

Start to move the Turtle through the maze.
How much do you need to move the Turtle forward? Use Trial and Error to find out!

```
from turtle import *
speed("slowest")
shape("turtle")
bgpic("maze.gif")
backward(80)
left(90)
forward(?)
```

If the turtle moves too far use a lower number. If it moves too little use a higher number. Keep trying numbers until the turtle moves forward to the top of the path.



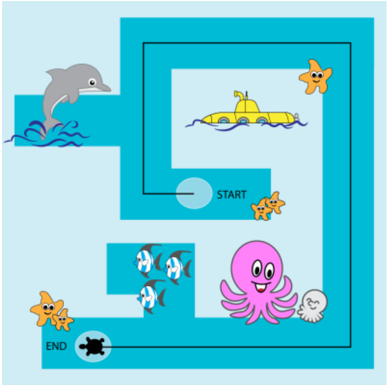
Program the Turtle to Move to the End of the Maze

6. Apply your skills to move the Turtle to the end of the maze. Use the following commands:

```
forward( )      backward( )      left( )    right( )
```

HINTS:

- The maze is 600 pixels wide by 600 pixels tall.
- The number you pick should always be less than 600.
- Compare the path to a part you have already done. Is the path shorter or longer? This will help you pick a number to use.
- Pretend your body is the turtle. Point it in the direction the turtle faces. Do you need to turn right or left to move along the path?
- Write one line of code at a time – then test it. This will allow you to focus on each movement.



Close the Program