

The Turtle Prize - Lesson 3 - Conditionals



This work is licensed under a [Creative Commons Attribution-ShareAlike 4.0 International License](https://creativecommons.org/licenses/by-sa/4.0/).

Contents

Introduction	1
Learning objectives	1
Resources	1
Lesson Summary	1
Sabotage! : 10 minutes	2
Conditionals : 15 minutes	2
Work on their Turtle Prize entry : 25 minutes	2
Extension activity	3
Plenary : 10 minutes	3
Enter the competition!	3
Outcome	3

Introduction

In the 2nd lesson, students learnt more about Python, including how to implement loops and use variables.

Learning objectives

- Improve resilience to syntax errors.
- Students will learn some more advanced graphical techniques.
- Students will learn about conditionals.
- Students will learn to find documentation on libraries using the internet.

Resources

- 15 copies of [lesson 3 handout](#)

Lesson Summary

- Sabotage!
- Introduce conditionals.
- Work on competition entries, make use of a conditional.
- Introduce new turtle techniques.

Sabotage! : 10 minutes

Explain they're going to play a game called Sabotage. This game was invented by [Alan O'Donohoe](#) to teach resilience in dealing with syntax errors.

Each pair loads their most recent program and saves it to a new file called 'sabotaged.py' using `file->save as`. They then make 5 deliberate syntax errors. Students will already have come across syntax errors as they are easily caused by:

- missing a `:` at the end of a line,
- not indenting properly after a `while` loop,
- getting case wrong - Python is case sensitive,
- not matching (and) brackets.

After making 5 errors, one person from each team swaps to a different team. The person arriving has to try and fix the syntax errors, and if they get stuck the person who made them can give them clues.

This game is really good fun and worth using a few times until your students are really quick at spotting and fixing errors.

Conditionals : 15 minutes

In our programs so far we have only covered sequences and looped sequences. When we want our program to do something different depending on its environment, we need to use conditionals.

Ask the students for examples of computer games they play. Then ask them what conditionals there are in the games. For example:

- In a driving game, if the speed is too high, the car might skid.
- In a platform game, once a character's life points are too low, they die.
- In a shoot-em-up, once an enemy has been hit it dies.

Ask the students to look at the *conditional statement* in their handout.

```

1  if loops == 3:
2      pencolor("red")
3  elif loops > 10:
4      pencolor("green")
5  else:
6      pencolor("blue")

```

A conditional statement always has to start with the `if` keyword. It can any number of `elif`s - even 0. And it can have an optional `else` that only happens if all the other conditions fail.

Each `if` or `elif` has a *condition* that is evaluated to True or False. If the condition is True, the following indented codes are run.

Ask them what colour the turtle will draw with the variable `loops` set to 5. Ask what `loops` would need to be to change the colour to red or green?

Work on their Turtle Prize entry : 25 minutes

- Ask the students to work on their competition entry, and to make use of a conditional statement to change the colour, or another graphical parameter (such as line width).
- Encourage students to use the new function they chose from last lesson's homework.
- Encourage students to experiment with the *fill* and *stamping* options detailed in the handout.
- Encourage students to use the internet to search for other turtle techniques, starting with the link in the handout.

Extension activity

Ask students to start creating their own colours using `pencolor()`. Up till now, we've only used colours written in English, like "red", "blue". We can also make our own custom colours by passing an argument for red, green and blue:

```
pencolor(0.5,0.1,0.3)
```

The values have to be between 0 and 1. Students can also look at using a variable instead of typing the number in, then the colour can change as the drawing progresses.

Plenary : 10 minutes

- Remind students that they've learnt the fundamentals of computer programming, recap on the meanings of:
 - sequences,
 - loops,
 - data,
 - conditionals,
 - operators.
- That programs are made of sequences of instructions that can be looped.
- That different sequences can be run using conditional statements.
- That they can use their creativity with computers and computer programming.

Enter the competition!

Ask the students to add their names, class and school and your email to the top of the code using comments (shown in the handout) Then ask them to submit their code to us at entries@turtleprize.com.

Outcome

All students:

- Understand that a conditional allows a computer program to change its behaviour.
- Used either stamp or fill in their code.

Most students

- Have used a conditional in their competition entry .

Some students

- Have used a conditional to change a different graphical parameter.
- Have researched the stamp shapes using the internet.
- Experimented with creating their own colours.