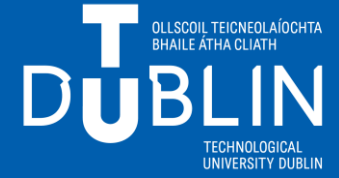




Trinity College Dublin  
Coláiste na Tríonóide, Baile Átha Cliath  
The University of Dublin

pytch\_



## Lesson 7

Finishing off the “Catch the apple” game



Developed by:

pytch.team

<https://pytch.org/>

<https://pytch.scss.tcd.ie/>

# Python while loops

You can use a `while` loop to keep executing some Python commands/block of code until a condition is no longer true. So, if we have `while True` the code will be executed forever because the condition is the constant Boolean `True` itself.

## Example in Pytch :

```
while self.x_position < 0:
```

```
    self.say("I'm at the bottom of the stage")
```

```
    self.glide_to_xy(random_x, random_y, 1)
```

These two lines of code will be executed so long as the condition self.x\_position < 0 is true

```
while score < 50:
```

```
    self.glide_to_xy(10, 10, 1)
```

```
    self.glide_to_xy(-10, -10, 1)
```

These two lines of code will be executed so long as the condition score < 50 is true



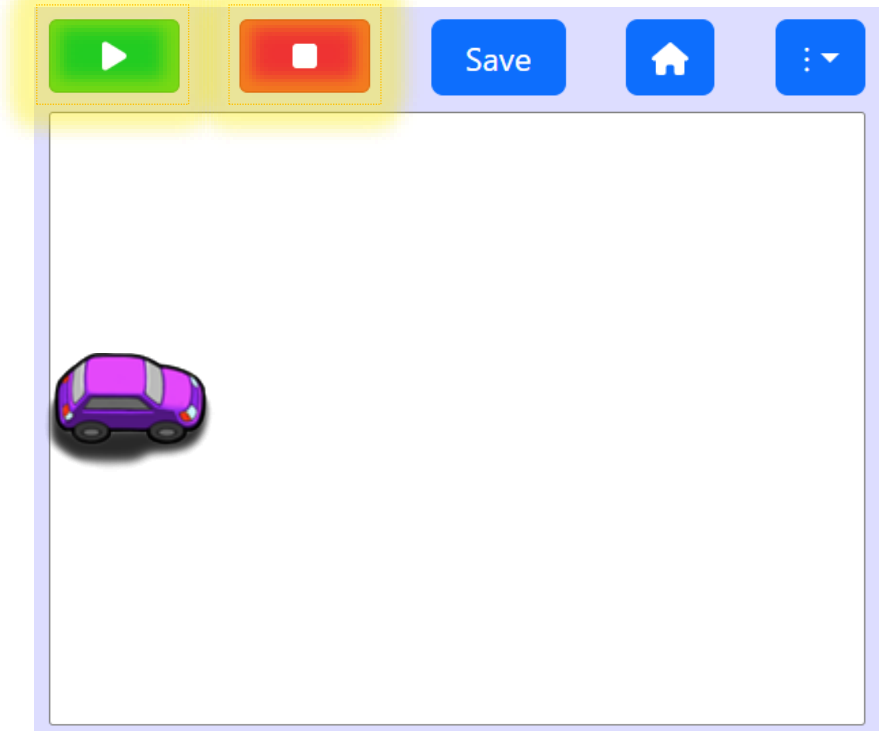
# Python nested while loops

Just like if statements, you can also have nested while loops.

**Example in Pytch :**

```
when green flag clicked

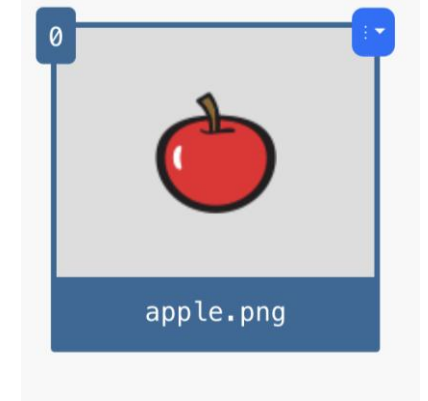
1 while True:
2     self.go_to_xy(-200, 0)
3     self.wait_seconds(2)
4     while self.x_position < 200:
5         self.change_x(3)
```



# Pytch switch costume method

To change a Pytch Sprite costume you can use: `self.switch_costume(name_or_number)`  
using the costume's **name** (e.g. `"apple.png"`)  
or its **number** = the position (e.g. `0` if it is the first costume).

Similarly, if you want to change the Pytch Stage backdrop you can use: `self.switch_backdrop(name_or_number)`



## Example in Pytch with position and backdrop name:



```
self.switch_backdrop(0)           ----->
self.switch_backdrop("Jungle.png") --->
self.switch_backdrop(2)           ----->
```

Switch to first backdrop: **"Forest.png"**  
Switch to Jungle backdrop: **"Jungle.png"**  
Switch to third backdrop: **"Wetland.png"**



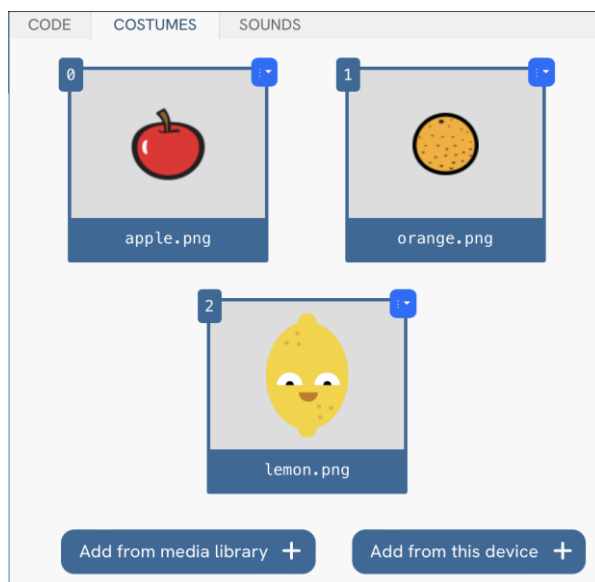


5 minutes

# Worksheet 1

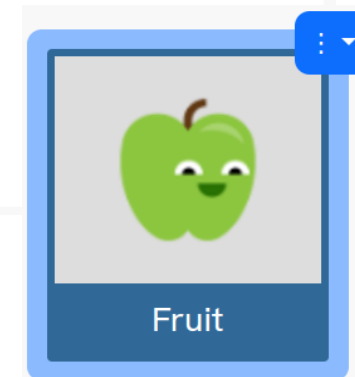
## Now work in pairs:

- This sprite has three costumes (there is a picture of the costumes tab below)
- What does this new code do?
- Write your answers on worksheet 1



when green flag clicked

```
1 self.score = 0
2
3 while True:
4     self.switch_costume(1)
5     self.go_to_xy(random.randint(-190, 190), 200)
6     self.show()
7     while self.y_position > -140:
8         self.change_y(-3)
9         self.say(self.score)
10        if self.touching(Bowl):
11            self.score = self.score + 5
12            self.hide()
```



Trinity College Dublin  
Coláiste na Tríonóide, Baile Átha Cliath  
The University of Dublin

pytch\_

# Try it out

- Follow this link to get a Pytch project that you can run
- Run the program
- Does the Fruit Sprite do *exactly* what you thought it would do?
- If not:
  - Look at the differences
  - Correct your answer on worksheet 1

<https://pytch.org/app/lesson/sbys/7>



# Questions to do in pairs – Worksheet 2

1. We changed the sprite name to Fruit. Did that change anything in how the program works? If not, why do you think we did it?
2. Why does the sprite look like an orange and not an apple? Experiment with the number in `switch_costume` being 1 or 0. How does this work?
3. If you move `switch_costume` as shown on the right:
  - Does it still work?
  - How many times is the command run now, vs before?
  - Is this better or worse (or the same)?
4. What happen if you change the number in `switch_costume(1)` into a 2? Does it work? Why or why not?
5. The bowl can move only left and right to collect the fruits. Do you think the game would be better if the bowl could also move up and down?

when green flag clicked

```
[...]  
self.switch_costume(1)  
while True:  
    self.switch_costume(1)  
    self.go_to_xy ...  
[...]
```

<https://pytch.org/app/lesson/sbys/7>



# Tasks – Worksheet 3

Work in pairs on these activities:

1. **Make the Fruit give you 10 points instead of 5.**
2. **Have the Fruit choose its costume at random each time it starts falling, so that it's sometimes an apple and sometimes an orange.**



To select the costume for the fruit, we use a number, so you could use `random.randint()` here to make a random choice between the two. You could also use the `next_costume` Pytch method — find it in the help panel on the left and learn more about it.

You can personalise the game as you like,  
You can create other objects to catch or  
change the appearance of the bowl,  
adding other pictures to your project to  
use as costumes.



## Extension

Finished early? Challenge: (1) Make oranges worth more than apples — give 20 points each time an orange is collected.

(2) Now the orange gives you more points than the apple, can you make the orange appear less often?



- 1) You can use `self.costume_number` to obtain the current costume number.
- 2) You can use `random.randint()` but then, instead of using the result directly as the costume number, make a decision based on the value. You might find an `if/else` statement useful — look this up in the help. This is not easy!





# Recap

Today we have

1. Learned about nesting while loops and about the Pytch *switch\_costume()* command.
2. Learned to apply these ideas in extending the fruit-catching game.

More than that!

- We have learned about lots of coding concepts and ideas — commands, scripts, loops, variables. Really, pretty much everything you need to make your own projects.

We hope you had some fun while doing it!

