Upper Key Stage 2 - Session 7
Increase the challenge - Creating new Python variables, incrementing variables.

Objectives

- Design and write programs independently in Python using **repetition** and **selection**: **for count in range (n)**:
- Debug their Python programs, demonstrating an understanding of the appropriate syntax
- Use indents correctly in Python
- Create and increment **variables**
- Use **comments** in Python to explain their programming.

Resources

- Interactive White Board (IWB)
- Levels 106 to 109 in Rapid Router
- Resource Sheets UKS2-S7-1, UKS2-S7-2
- UKS2 Levels Guide
- UKS2 Program Solutions Table
- UKS2 Assets Python Cards

Vocabulary

- Variable
- Function
- Increase
- Increment
- Spiral
- Algebra
- Mathematical operators +, - , *, /

Let's get started

Start with an unplugged activity to explain the concept of a **variable**.

Prepare a little open box to hold a number card. Label the box on the outside with the letter **n**.

Have numbered cards 1- 10 ready, which will fit neatly in the box.

Starting with the number 1 card in the box ask a child to be the delivery van driver and move forward n steps where n is the number they find in the box.

Repeat this exercise again, this time with another child selecting a random number card from the pack and placing it secretly in the box.

Explain that in Python you can create a **variable** which you name and can then change as you wish.

When a program needs to use that **variable** it 'looks in the box' to see what value/number is currently in there.

Use Resource Sheet UKS2-S7-1 if you wish to reinforce this concept. *[fig S7.1]*

Show Level 96 to illustrate this. *[fig S7.2]*

Do you remember how you used two repeat loops to solve this?

Let's see how creating a **variable** for the number of forward moves would work here.

Let's call the **variable n**. We can then write

for count in range (n):

but we need to tell Python what the value of **n** is to start with. **Who can suggest that?**

We need to start with setting **n= 2**

At what point in the program will we change that to n=3?

Support the children to come up with this solution and test it:

```
import van

v = van.Van()

n = 2
for count in range(n):
  v.move_forwards()

v.turn_left()

n = 3
for count in range(n):
  v.move_forwards()
```

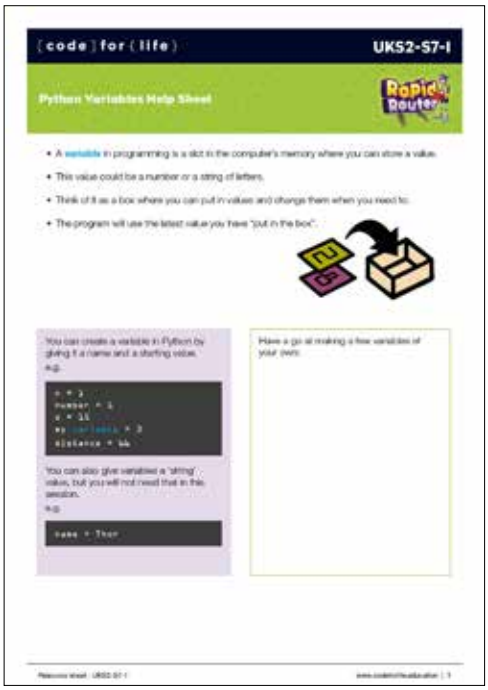

fig S7.1

fig S7.2

We looked at Level 96 to show how variables can work. In this case it made our program a bit longer than necessary. Let's now look at where a variable can really save lines of code.

Show Level 106 on the IWB. *[fig S7.3]*

fig S7.3

Who can describe the shape of this route using mathematical vocabulary?

What type of spiral is this? How does the size of the straight sections change?

Group the children into pairs and ask them to write the algorithm in English to drive the van around the spiral.

Share feedback and write the combined effort on a flipchart, it should look something like:

Can you see a pattern here? Can you spot some sort of repeat loop?

Discuss the pattern: a right turn followed by a move forward which gets bigger/ increases each time by one step.

If we can put this in a **repeat** loop, we could write a very neat program. **Can anyone suggest something which uses the idea of a variable?**

If we start by giving our **variable** a value of 1, we could code

n=1

but next time that value of **n** has to be 1 greater.

We can do that cleverly by coding

n = n +1

This says make **n**, 1 bigger that it was before.

So what would the value of n be?

Show the following code on Resource Sheet UKS2-S7-2, and ask the children to discuss and explain what is happening. *[fig S7.4]*

Ask a child to write the code in the app and test it together. Debug any input errors together.

```
import van

v = van.Van()

n = 1

while not v.at_destination():
  v.turn_right()
  for count in range(n):
    v.move_forwards()
  n = n + 1
```

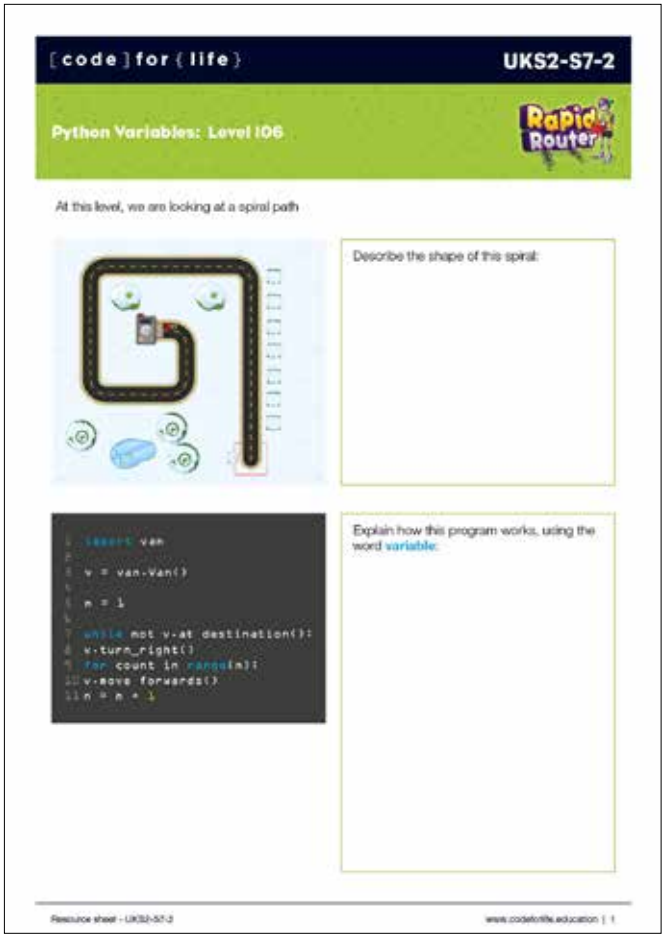

fig S7.4

Independent activity

You are now going to have a go at some different spirals, where the number of forward moves increases or decreases mathematically. It won't be just simple addition. You may have to use other mathematical operators. **Can anyone tell us what those might be?**

Are you ready for the challenge?

Children try Levels 106 to 109 in pairs or independently. Make sure you have the mathematical operator signs visible: **+**, **-** , ***** , **/** , on the board / code wall.

Share and review

Allow enough time for a different pair of children to go through a solution for Levels 107 to 109.

Explain that in the next session they will be creating their own road challenge, and they may wish to create a spiral to challenge their friends.

Extension activity

Remind the children about the use of **comments** introduced in Session 6.

To create a **comment**, you must put a hash symbol **#** in front of your text so that Python knows it's just information for the reader and not code that Python needs to action.

For homework, they could select a Level from 106 to 109 and explain their programming rationale using the **comment** facility. They could then screen grab this for their Computing diary/ portfolio on your school system, and perhaps to display on the classroom code wall.