A children's guide to Python programming

By Thales group

1. Printing text and creating variables

1. Open the Python app  and tap 

Press Menu and start a NEW program.
Type a name for your program and select a folder to save it in.

2. Type these commands into the 'script' window:

```
print("Hello world.")
print("\n")
print("I am learning Python.")
```

Press Run and watch the 'interpreter' window.

Program – A sequence of commands that are followed in order to carry out a task.
Run – Carrying out the commands in a program. Also known as execute.

What does the `print` command do?
What does printing \n do?
What happens if you make a mistake in your commands?

3. Press Menu and start a NEW program.
Type a name for your program and select a folder to save it in.

4. Type these commands in and then Run them:

```
Teacher name = input("What is your teacher name? ")
print("Hello" ,teacher name)
```

What does the `input` command do?
Does it matter if you type in text other than your name?

Variable – A value that can be stored and used in a program.

Edit and improve:

- Add a variable to store a surname. Then change the print command so it prints their full name.

2. Calculations and random numbers

1. Open the Python app  and tap 

Press Menu and start a NEW program.
Type a name for your program and select a folder to save it in.

2. Type these commands in and then Run them:

```
Print(20+20)
```

Is the calculation still solved if you use a negative number or a decimal number?

Edit and improve:

Change the commands to do a different calculation, such as a: take away -, multiplication * or division /.

Testing - Trying out a program to check if it works as expected.
Debugging - Finding and correcting mistakes in a program's source code.

3. Press Menu start a NEW program
Type a name for your program and select a folder to save it in.

4. Type these commands in and then Run them:

```
Import random
Number = random.randrage(5,10,15)
Print(number)
```

What does the **.randrange** command do?

Edit and improve:

- Change the number 5 to a smaller number and the number 15 to a bigger number to see what effect this has on the program.
- Add some commands to do the calculations with the random number. E.g. print(number+10)

3. Number variables and adding comments

1. Open the Python app  and NEW start program.

Type these commands in and then Run them:

```
number = int(input("Type a whole number: "))
answer = number * 8
print(number, "multiplied by 8 is" ,answer)
```

What happens if you type in a decimal number
instead of an integer (whole number)?

Edit and improve:

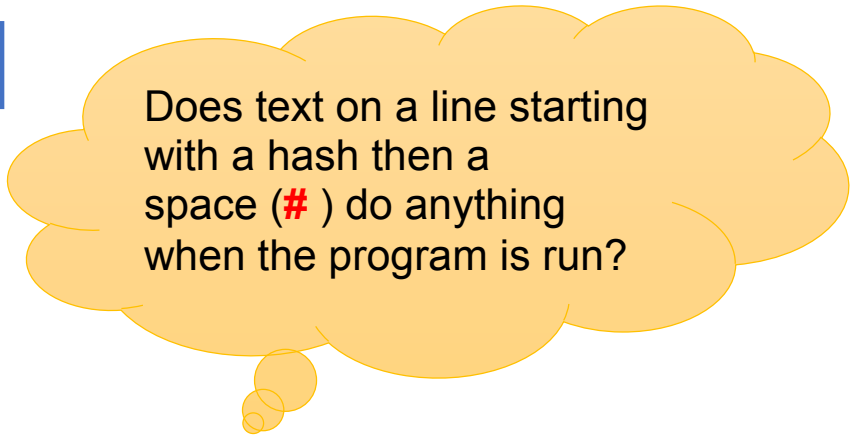
- Find out what changing **int** to **float** lets you do.
(Remember to change it back to **int** afterwards!)
- Add commands so the answer to an addition is printed as well. You will need to use another variable called **answer2**:

```
answer2 = number + 6
print(number,"add 6 is",answer2)
```

- Change the program so you have to type in two numbers at the start to use in each calculation. You will need to use another variable called **number2**. Remember to print it on the screen before you show the answer!

2. Add these commands to your program:

```
# This is a comment.
```



Comments - Notes in a program's code which explain what commands do to remind you. They are not run.

Edit and improve:

- Type some comments beside some commands to explain what they do.

4. If statements

1. Open the Python app  and start a program. NEW

Type these commands in and then Run them:

```
answer = input("Do horses bark? ")
if answer = = "no":
print("Correct")
else:
print("Wrong")
```

What does this program do? Why do you think two equals signs are used and not just one?

IF statement - Decides which commands to run depending on whether certain things (conditions) are true or false.

Edit and improve:

- Change the question being asked (and the answer too, if needed).

2.Start a **NEW** program.

Type these commands in and then Run them:

```
mark = int(input("Score: "))
if mark < 40:
print("Bad")
elif mark > 40:
print("Great")
else:
print("Good")
```

What does this program do?
What does the **elif** command let you do?

Edit and improve:

- Add another **elif** command in the middle so that a score of more than 60 is rated as **"Super"**.

Programming challenge:

Create a program that asks a maths calculation and prints if the user answers it right or wrong. Can you change one of the numbers in it to a random number?

5. Arrays

1. Open the Python app  and start a **NEW** program.

Type these commands in and then **Run** them:

```
import random
vegarray = ["broccoli","beans"]
veg = random.choice(vegarray)
print("Mashed",veg," on the ceiling.")
veg = random.choice(vegarray)
print("Boiled",veg," on the floor.")
veg = random.choice(vegarray)
print("Stewed",veg," in the corner.")
veg = random.choice(vegarray)
print(veg," upon the door.")
```

Use copy and paste to help you quickly copy this!

Copy **Paste**

What does this program do?
What is the purpose of the **vegarray**?

Array - A list of values.

2. Start a **NEW** program, type these commands in **Run** them:

```
vehicles = ["bus","car","train"]
print(vehicles[0])
print(vehicles[1])
print(vehicles[2])
vehicles.append("plane")
print(vehicles)
vehicles.pop(2)
vehicles.insert(2,"boat")
print(vehicles)
vehicles.remove("car")
print(vehicles)
```

Can you see what the:
.append, **.pop**,
.insert and **.remove**
commands do?

Programming challenge:

Create an array to store a list of names. Add commands to: .append, .pop,
.insert and .remove names. Find out what the .sort() command does.

6. Functions

1. Open the Python app  and NEW start a program.
Type these commands in and then Run them:

Function - A sub-program which is placed at the start of a bigger program and can be called (run) later using its name.

```
import random
def cointoss():
options = ["heads","tails"]
result = random.choice(options)
print(result)
cointoss()
cointoss()
cointoss()
cointoss()
cointoss()
```

What does this program do?
Why is better to call the function five times than to copy all of its commands five
times?

Edit and improve:

- Change the program so it shows the results of rolling a six-sided dice instead. You don't need to put "" around the options because they are numbers.

Programming challenge:

Create a program that tells a user's fortune by calling (running) a function two times which randomly picks a prediction from an array:
e.g. You will be given money.
You will become famous.
You will see an alien.
You will find a lost item.
You will score well in a test.
Can you ask the user to **input** their name so that it is included in the predictions (e.g. Tom will be given money)?

7. Iteration (looping)

1. Open the Python app  and start a NEW program. Type these commands in and then Run

```
for i in range(4):
print("Hello world")
```

What happens if you change **4** to a different number?

Iteration - A way of repeating or looping commands multiple times.

2. Start a NEW program, type these commands in and then Run them:

```
for i in range(1,10):
print(i*10)
```

What happens if you change **4** and 10 to different numbers?

3. Start a NEW program, type these commands in and then Run

```
password = "fish"
guess = ""
while (password != guess"):
guess = input("Enter password: ")
if password = = guess:
print("Correct")
else:
print("Try again")
```

If = = means 'equal to',
what does != mean?
What does a **while** loop do?

Programming challenge:

Create a program in which the computer sets the password as a random
integer from 1 to 100 and user has to correctly guess it.
Can you use: **if**, **elif** and **else** commands to give the user clues (e.g. **"Too high"** or **"Too low"**)? Can you add a variable which counts the number of
guesses (**count = count + 1**)?

8. Parameters and validation

1.Open the Python app  and start a **NEW** program.

Type these commands in and then **Run** them:

```
def timestable(multiple):
for i in range(1,11):
print(i,"* ",multiple," =",i*multiple)
multiple = int(input("Type a number: "))
print("\n")
timestable(multiple)
```

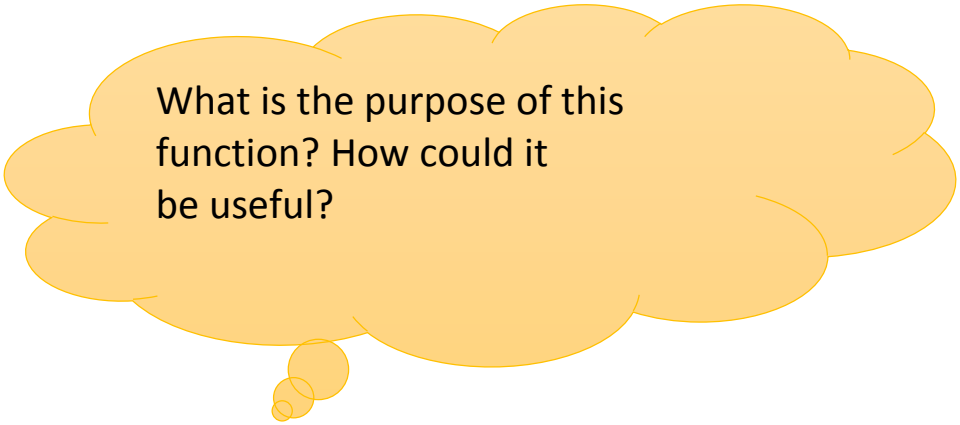
Parameter - A way of passing a value from the main program to a function when it is called (run).

Programming challenge:

Create a function that prints the biggest of two values.
In the main program below it, the user will input two **int**egers and they will
be passed to the function as parameters.

2. Start a **NEW** program, type these commands in and then **Run** them:

```
def validation():
number = 0
while True:
try:
number = int(input("Type a whole number: "))
except ValueError:
print("Not a whole number!)
else:
return(number)
x = validation()
```



Validation - Automatic checking by a computer to ensure that an entered value is sensible.

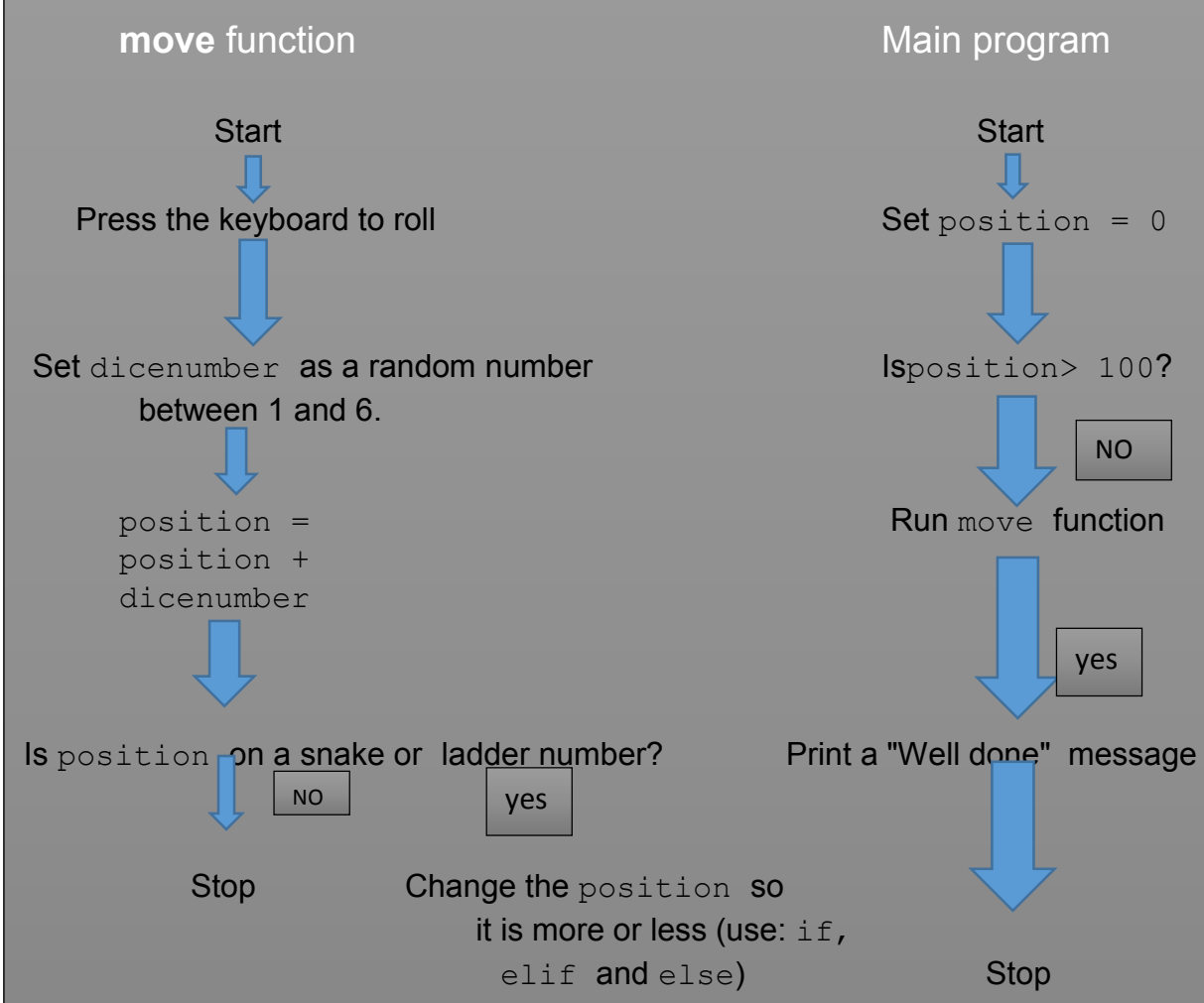
Programming challenge:

Create a function which prints whether a triangle is: equilateral, scalene or isosceles. In the main program below it, the user will input the triangle's three side lengths as **int**egers (checked with the validation function above) and they will be passed to the function as parameters.

```
if side1 == side2 == side3:
print("Equilateral")
elif side1 == side2:
print("Isosceles")
elif side2 == side3:
print("Isosceles")
elif side1 == side3:
else:
print("Scalene")
```

9. Algorithm

Create a simple version of a Snakes and Ladders game:



• Can you add more print commands to display what is happening on screen?
• Can you make the game print the player's name at the end?
• Can you add another player to the game whose position is stored in a variable
called *position2*? You will need to make the game let each player move in turns. You could create a variable called finished which is set to 0 at the start and changes to 1 when a player wins, forcing the game to stop.