Tiffany Tech   Smuffy   Dan Devices   Chew-Chew

Welcome, budding coders!

Say hi to our heroic adventurers: Dan Devices and Tiffany Tech. Together they are the best of friends, and an amazing team who love to explore and help each other whenever they can. Dan adores devices, Tiff loves tech, and both are on a constant quest to learn and make learning fun. The friends are joined by their pets, Smuffy and Chew-Chew, who help out in their own special way. Together... they are The QuestKids!

Jumping from a block-based coding language like Scratch to a text-based coding language like Python is a quite a leap.

Tiffany, Dan, Smuffy and Chew-Chew together will help you take the leap. They'll take you from basic Python commands to start with simple shapes, and then keep you fully engaged to master key Python code to create most impressive 3D and intricate images – all in easy steps.

Learning Python has never been so much fun!

About the author: Max Wainewright, author of four books in The QuestKids series, is an experienced computing teacher and has taught in the UK and Singapore. He also has significant experience of working in the EdTech industry, including working as an Educational Software consultant for major companies such as Discovery and LEGO. Currently, he teaches coding in a London primary school and runs an innovation group supporting other schools. Details of his other books are on the inside back cover.

THE QuestKids®

Coding with Python

Create Amazing Graphics

in easy steps
BOOKS FOR KIDS

turtle.goto(x,y)

turtle.dot(10)

turtle.fd(100)

Max Wainewright

Contents

Getting Started

In this book you will learn how to create some amazing graphics using Python. You will also become a great Python coder!

We will start by drawing some simple shapes and learning how to use loops to repeat code. We will discover how random numbers can create elaborate patterns and how to mix colours using variables. Later on in this book we will show you how to draw pictures with code and how to create your own commands by defining your own functions.

Develop your skills and create some amazing graphics!

1 Getting Python

You need to install the Python program on your computer.

www.python.org

Open your web browser and go to **www.python.org**

2 Click Downloads

Click the **Downloads** tab at the top of the page.

Downloads

You can use an iPad app but these instructions are for a desktop or laptop computer.

3 Download Python

Download the latest version of Python for your computer.

Download the latest version fo

Download Python

Looking for Python different OS? Python for Wind
macOS, Other

Wait while the file downloads.

4 Run the download

Double-click the downloaded file to start installing the software.

python-3.10.0-....pkg

5 Install the download

Python should now start to install on your computer.

Installing
Python

Follow any steps shown on screen to complete the installation.

*Python was designed by Guido van Rossum in the late 1980s.*

PYTHON In the real world, Python is used to make website searches work, create games, handle large amounts of data, and create 3D images in movies.

Saying Hello

1 Make your first Python file

Start up Python. This launches something called the **shell**. Click **File** > **New File** to start typing a short file of code.

Python

File   Edit
New File
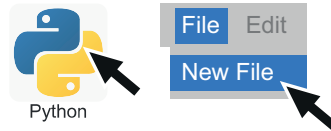
2 Start coding!

Carefully type in the following code:

```
untitled
print('Hello')
```

Hold down the **Shift** key...
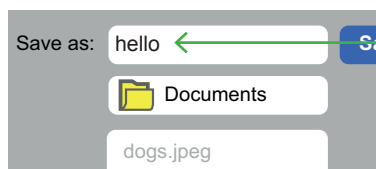
SHIFT

...and tap these keys to get the brackets.

(9   ) 0

The colours of your code change automatically to help you read your code.

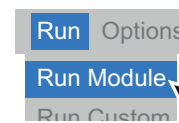## 3 Save your file

Click **File** > **Save**.

Save as: hello

📁 Documents

dogs.jpeg

Type in **hello** as the file name.

Browse to your **Documents** folder.

Save   Click **Save**.

4 Run the code

Click **Run** > **Run Module**.

Run   Options
Run Module
Run Custom

Don't worry if it isn't working yet.

! Check for errors

If your code doesn't work, go back and check it carefully.

( )

'

Make sure you typed the code exactly as shown. Check you have typed the () and ' symbols correctly.

Click **File** > **Save** and run your code again by repeating Step 4.

5 View your work

When you run some Python code, the results or output from the code appear in the IDLE shell.

```
IDLE shell
>>>
====== RESTART: hello.py ==
Hello
>>>
```

It should look like this.

USING PYTHON ONLINE

You can use Python online, without installing it on your computer. However, not all online versions let you create graphics. There are also some apps available for the iPad that run Python. If you are using an online Python website you should ignore some steps in the projects in this book — you won't need to save your work or switch on line numbers.

For more information on online Python, visit **maxw.com/pythonlinks**

Giant Circles

This first project will give you a chance to start coding with Python. We will start by loading the **turtle** commands so that we can do some drawing. This will allow us to type in commands to set the colour and draw some circles. By using different colours and different sized circles we will make some exciting patterns!

HOW THE CODE WORKS

We will start by drawing a very large green circle.
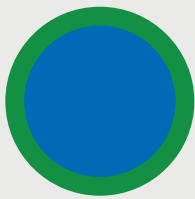
Next, we will draw a large blue circle.

Inside that we will put a medium-sized yellow circle.

A small red circle will be drawn next.

Finally, we will add a very small purple circle.

```
import turtle
```

This tells Python to load some extra commands that will let us draw on the screen. These commands are stored in something called the **turtle module** or **turtle library**.

```
turtle.color('red')
```

This command tells the turtle to draw with red. Make sure you type **color** not **colour**!

```
turtle.dot(500)
```

This will make the turtle draw a dot, 500 pixels wide. The bigger the number, the bigger the dot.
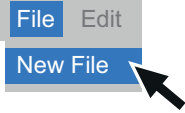
We'll need to type special commands to draw this pattern.

A pixel is one of millions of tiny dots on a computer screen that combine together to make images.
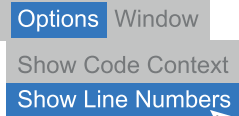
1 Make a new Python file

Start up **IDLE** or click **File** > **New File**.

File	Edit
New File	

2 Switch on the line numbers

Click **Options** > **Show Line Numbers**.

Options	Window
Show Code Context	
Show Line Numbers	

3 Start coding!

Type in your code.

```
1 import turtle
2
3 turtle.color('green')
4 turtle.dot(500)
```

Import the graphics library.
Leave a blank line — it will be clearer to read.
Set the green colour.
Draw a very large circle.

Press the Enter key at the end of each line.

4 Save your file

Click **File** > **Save**.

Save as: giant
📁 Documents
homework.doc
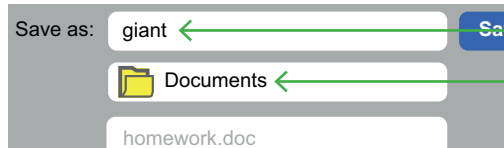
Type in **giant** as the file name.

Browse to your **Documents** folder.

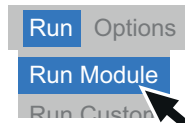**Save**   Click **Save**.

LINE NUMBERS

Line numbers are used in Python to show where an error is. They are also useful for keeping track of where you are when typing in a program.

Don't type in the line numbers — they will show up automatically.

5 Run the code

Click **Run** > **Run Module**.

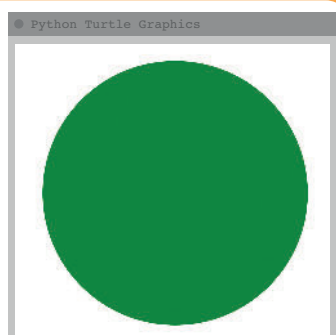| Run	Options
Run Module	
Run Custom	

F5

Or you can just press the F5 key on the keyboard to run your code.

6 View your work

Your code should make another window appear on your computer.

It should contain a large green circle, like this:

● Python Turtle Graphics

! Check for errors

Help!

If you don't get a green circle, go back and check your code carefully.

( ) **'** Make sure you typed the code exactly as shown. Check you have typed the () and ' symbols correctly.

Finally, click **File** > **Save** and run your code again by repeating Step 5.

➔

7 More code!

Find the IDLE window with your code. Add two new lines below your code.

```
1 import turtle
2
3 turtle.color('green')
4 turtle.dot(500)
5 turtle.color('blue')
6 turtle.dot(400)
```

Don't change these lines.
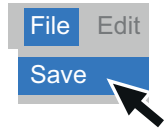
Now, pick the blue colour.

Draw a slightly smaller circle.

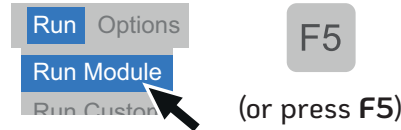*To make more space, click at the end of your code then Press Enter.*
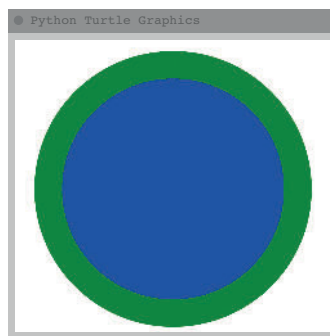
8 Save

Click **File** > **Save**.

File	Edit
Save	

9 Run all the code

Click **Run** > **Run Module**.

Run	Options
Run Module	
Run Custom	

F5

(or press **F5**)

The output means what your code makes happen. Here, it is some graphics.

10 View your work

Python Turtle Graphics

Another window showing the **output** from your code should appear.

This time, there should be two circles.

! Check for errors

If your code doesn't work, go back and check it all carefully. Make sure you haven't missed out one of the lines.

Then, click **File** > **Save** and run your code again by repeating Step 9.

FINDING ERRORS

When you start coding with Python it can be hard to spot your errors, but this is a really important skill to learn. The more you code, the easier it will get.

Keep looking up and down your code to look for patterns. IDLE (the Python editor) changes the colour of some code to make it easier for you to find errors.

Can you find the errors in each of these? Only one has no errors!

A
```
import turtle

turtle.color('green')
turtle.dot(500)
turtle.color(blue')
turtle.dot(400)
```

B
```
import turtle

turtle.color('green')
turtle.dot(500)
turtle.color('blue')
turtle.dot(400)
```

C
```
import turtle

turtle.color('green')
turtle.dot(500)
turtle.color'blue')
turtle.dot(400)
```

D
```
impor turtle

turtle.color('green')
turtle.dot(500)
turtle.color('blue')
turtle.dot(400)
```

See below for the answers!

Answers: A) Missing quote mark before blue. B) No errors! C) Missing left bracket after color. D) Import is spelt incorrectly.
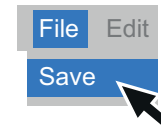
11 Another circle

Find the IDLE window with your code in. Add the lines of code marked in green (lines 7 to 12).

```
1 import turtle
2
3 turtle.color('green')          Don't change
4 turtle.dot(500)                this part.
5 turtle.color('blue')
6 turtle.dot(400)
7 turtle.color('yellow')    ← Pick yellow.
8 turtle.dot(300)           ← Draw a medium circle.
9 turtle.color('red')       ← Choose red.
10 turtle.dot(200)          ← Draw a smaller circle.
11 turtle.color('purple')   ← Pick purple.
12 turtle.dot(100)          ← Draw a small circle.
```

12 Save again

Click **File** > **Save**.

File | Edit
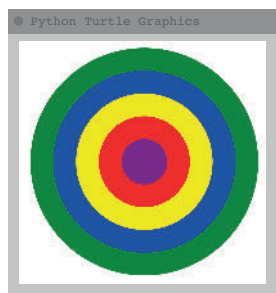Save

13 Run the code

Click **Run** > **Run Module**.

Run | Options
Run Module
Run Custom…

F5

(or press **F5**)
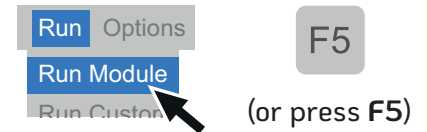
14 View your work

The turtle graphics window should appear with an image like this:

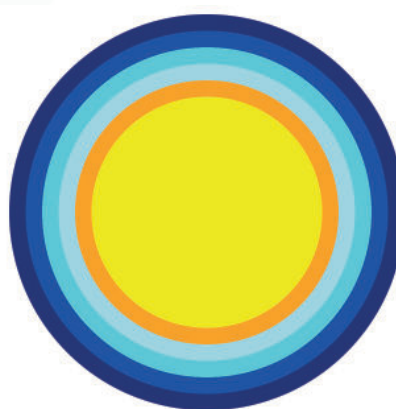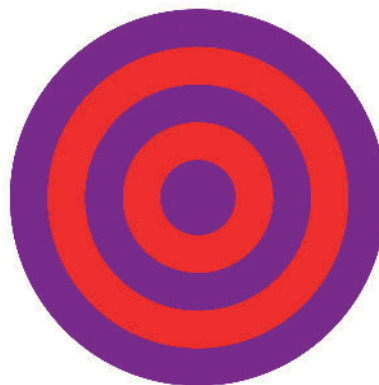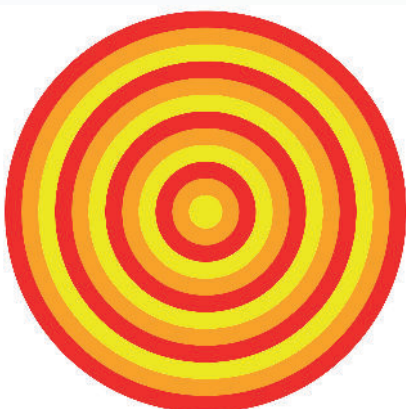Python Turtle Graphics

! Check for errors

Check your code carefully if it doesn't work properly. Make sure your follow the pattern of changing colour then drawing a dot on the next line.

After that, save and run your code again.

Challenges

Experiment with your code and make your own designs!

Can you make the following patterns?

The QuestKids® series is designed to make learning fun!
Let the two best friends along with their pets inspire
you to make learning fun.

Using the proven approach of learning In Easy Steps, this guide is:
• Easy to follow • Fully illustrated • Packed with tips

Python is a powerful, text-based programming language essential to grasp for serious coding but can be dull to learn. This book focuses on inspired learning. Step-by-step, it illustrates how to use Python code to create exciting and colourful graphics – making learning Python great fun!
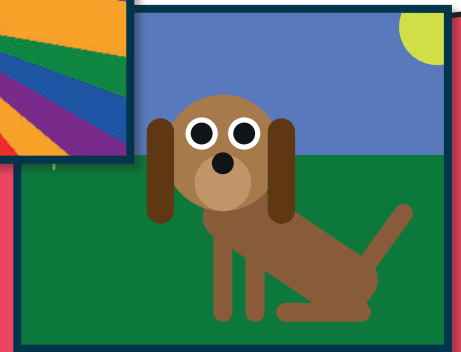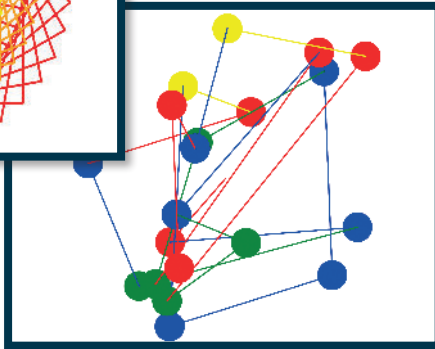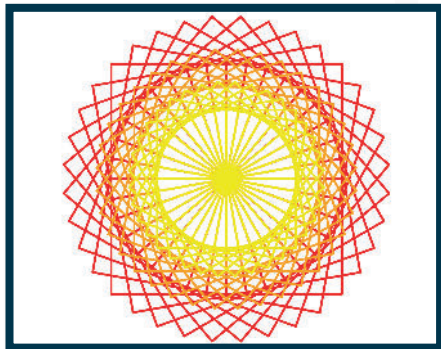
An engaging and easy introduction to Python.

Learn Python code to:

• Use random numbers to create unique artwork
• Mix colours together using variables to create amazing effects
• Use loops to repeat your code and create intricate patterns
• Code your own functions and build up your own designs

Supported by a designated website www.thequestkids.com
Learning couldn't be more fun!

CATEGORIES
Children's/Hobbies/Computers
Children's/Education/Technology/IT

www.thequestkids.com | www.ineasysteps.com