


Coding with Python



A close-up photograph of a person's face in profile, showing their eye and forehead. They are looking towards the right side of the frame. In the foreground, a portion of a laptop screen is visible, displaying a green-bordered callout box with text about Python's popularity and applications.

Python is one of the most popular, and widely used programming languages in the world. It's currently used by NASA scientists, at the Large Hadron Collider, as well as by banks and the Stock Market. Python code helps sort through huge amounts of data, helping users pick out what's important, and it helps form bonds and essential links between other programming languages.

It's therefore safe to say that learning Python should be a top priority to anyone who wants to start coding. With a good understanding of Python, you'll be able to create everything from your own text adventures to Big Data analytics and even Artificial Intelligence.

Read on, and let's start seeing how powerful Python is.

-
- 22** Why Python?
- 24** What can you do with Python?
- 26** Python in Numbers
- 28** How to Set Up Python in Windows
- 30** How to Set Up Python in Linux
- 32** Python on the Pi
- 34** Getting to Know Python

Why Python?

There are many different programming languages available for the modern computer, and some still available for older 8 and 16-bit computers too. Some of these languages are designed for scientific work, others for mobile platforms and such. So why choose Python out of all the rest?

PYTHON POWER

Ever since the earliest home computers were available, enthusiasts, users and professionals have toiled away until the wee hours, slaving over an overheating heap of circuitry to create something akin to magic.

These pioneers of programming carved their way into a new frontier, forging small routines that enabled the letter 'A' to scroll across the screen. It may not sound terribly exciting to a generation that's used to ultra high-definition graphics and open world, multi-player online gaming. However, forty-something years ago it was blindingly brilliant.

Naturally these bedroom coders helped form the foundations for every piece of digital technology we use today. Some went on to become chief developers for top software companies, whereas others pushed the available hardware to its limits and founded the billion pound gaming empire that continually amazes us.

Regardless of whether you use an Android device, iOS device, PC, Mac, Linux, Smart TV, games console, MP3 player, GPS device built-in to a car, set-top box or a thousand other connected and 'smart' appliances, behind them all is programming.

All those aforementioned digital devices need instructions to tell them what to do, and allow them to be interacted with. These instructions form the programming core of the device and that core can be built using a variety of programming languages.

The languages in use today differ depending on the situation, the platform, the device's use and how the device will interact with its

```

Bombs - GUI - TheIDE - [d:\uppsrc\CtrlLib\ArrayCtrl.cpp windows-1252] { examples }

File Edit Macro Project Build Debug Assist Setup
GUI MSC71cdb Debug
Ln 639, Col 45

Bombs
CtrlLib
CtrlCore
RichText
PdfDraw
Draw
Core

EditCtrl.h
EditField.cpp
TextEdit.h
Text.cpp
LineEdit.cpp
DocEdit.cpp
ScrollBar.h
ScrollBar.cpp
HeaderCtrl.h
HeaderCtrl.cpp
ArrayCtrl.h
ArrayCtrl.cpp
DropChoice.h
DropBox.cpp
DropList.cpp
DropPusher.cpp
DropChoice.cpp
StaticCtrl.h
Static.cpp
Splitter.h
Splitter.cpp
FrameSplitter.cpp
SliderCtrl.h
SliderCtrl.cpp
ColumnList.h
ColumnList.cpp
Progress.h
Progress.cpp
AKeys.h

AKeys.cpp
RichText.h
RichText.cpp
Prompt.cpp
Help.cpp
DateTimeCtrl.h
DateTimeCtrl.cpp
Bar.h
Bar.cpp
MenuBar.cpp
ToolBar.cpp
ToolTip.cpp
StatusBar.h
StatusBar.cpp
TabCtrl.h
TabCtrl.cpp
TreeCtrl.h
TreeCtrl.cpp
DlgColor.h
DlgColor.cpp
ColorPopup.cpp
ColorPusher.cpp
FileSel.h
FileSel.cpp
FileDialog.cpp

SetCursor(p.y);
Ctrl::ChildGotFocus();

void ArrayCtrl::ChildLostFocus()
{
    if(cursor >= 0)
        RefreshRow(cursor);
    Ctrl::ChildLostFocus();
}

void ArrayCtrl::Paint(Draw& w) {
    LTIMING("Paint");
    Size size = GetSize();
    Rect r;
    r.bottom = 0;
    bool hasfocus = HasFocusDeep();
    int i = GetLineAt(sb);
    int xs = -header.GetScroll();
    int js;
    for(js = 0; js < column.GetCount(); js++) {
        int cw = header.GetTabWidth(js);
        if( (xs + cw - vertgrid + (js == column.GetCount() - 1)) >= 0)
            break;
        xs += cw;
    }
    Color fc = Blend(SColorDisabled, SColorPaper);
    if(!IsNull(i))
        while(i < GetCount()) {
            r.top = GetLineY(i) - sb;
            if(r.top > size.cy) break;
            r.bottom = r.top + GetLineCy(i);
            int x = xs;
            for(int j = js; j < column.GetCount(); j++) {
                int cw = header.GetTabWidth(j);
                int cm = column[j].margin;
                if(cm < 0)
                    cm = header.Tab(j).GetMargin();
                if(x > size.cx) break;
                r.left = x;
            }
        }
}

```



environment or users. Operating systems, such as Windows, macOS and such are usually a combination of C++, C#, assembly and some form of visual-based language. Games generally use C++ whilst web pages can use a plethora of available languages such as HTML, Java, Python and so on.

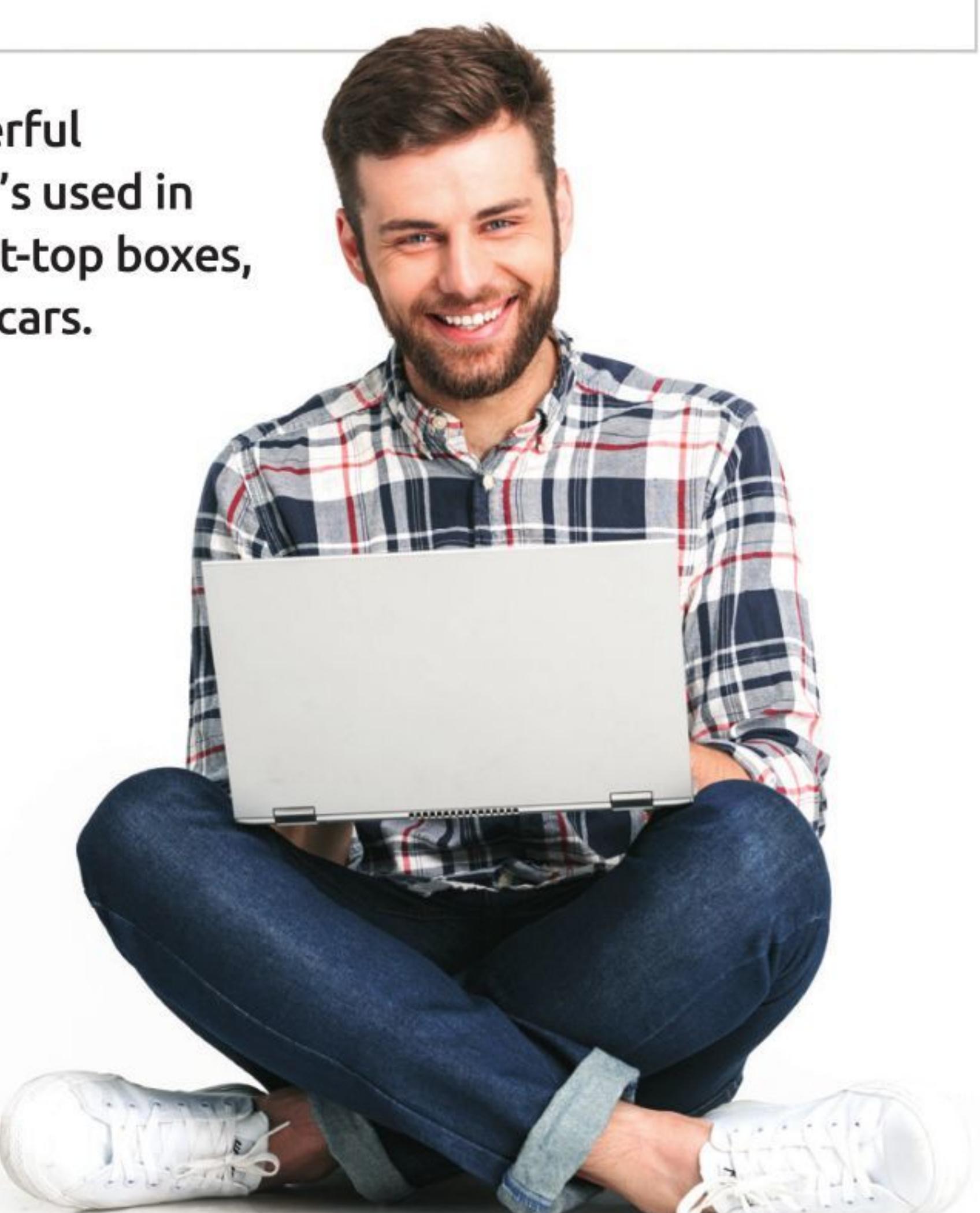
More general-purpose programming is used to create programs, apps, software or whatever else you want to call them. They're widely used across all hardware platforms and suit virtually every conceivable application. Some operate faster than others and some are easier to learn and use than others. Python is one such general-purpose language.

Python is what's known as a High-Level Language, in that it 'talks' to the hardware and operating system using a variety of arrays, variables, objects, arithmetic, subroutines, loops and countless more interactions. Whilst it's not as streamlined as a Low-Level Language, which can deal directly with memory addresses, call stacks and registers, its benefit is that it's universally accessible and easy to learn.

```
1 //file: Invoke.java
2 import java.lang.reflect.*;
3
4 class Invoke {
5     public static void main( String [] args ) {
6         try {
7             Class c = Class.forName( args[0] );
8             Method m = c.getMethod( args[1], new Class []
9                 [ ] { } );
10            Object ret = m.invoke( null, null );
11            System.out.println(
12                "Invoked static method: " + args[1]
13                + " of class: " + args[0]
14                + " with no args\nResults: " + ret );
15        } catch ( ClassNotFoundException e ) {
16            // Class.forName( ) can't find the class
17        } catch ( NoSuchMethodException e2 ) {
18            // that method doesn't exist
19        } catch ( IllegalAccessException e3 ) {
20            // we don't have permission to invoke that
21            method
22        } catch ( InvocationTargetException e4 ) {
23            // an exception occurred while invoking that
24            method
25            System.out.println(
26                "Method threw an: " + e4.
27                getTargetException( ) );
28        }
29    }
30 }
```



Java is a powerful language that's used in web pages, set-top boxes, TVs and even cars.



Python was created over twenty six years ago and has evolved to become an ideal beginner's language for learning how to program a computer. It's perfect for the hobbyist, enthusiast, student, teacher and those who simply need to create their own unique interaction between either themselves or a piece of external hardware and the computer itself.

Python is free to download, install and use and is available for Linux, Windows, macOS, MS-DOS, OS/2, BeOS, IBM i-series machines, and even RISC OS. It has been voted one of the top five programming languages in the world and is continually evolving ahead of the hardware and Internet development curve.

So to answer the question: why Python? Simply put, it's free, easy to learn, exceptionally powerful, universally accepted, effective and a superb learning and educational tool.

```
40 LET PY=15
70 FOR w=1 TO 10
71 CLS
75 LET BY=INT (RND*28)
80 LET BX=0
90 FOR d=1 TO 20
100 PRINT AT PX,PY;" U "
110 PRINT AT BX,BY;" o"
120 IF INKEY$="P" THEN LET PY=PY+1
130 IF INKEY$="O" THEN LET PY=PY-1
135 FOR n=1 TO 100: NEXT n
140 IF PY<2 THEN LET PY=2
150 IF PY>27 THEN LET PY=27
180 LET BX=BX+1
185 PRINT AT BX-1,BY;" "
190 NEXT d
200 IF (BY-1)=PY THEN LET S=S+1
210 PRINT AT 10,10;"score=";S
220 FOR V=1 TO 1000: NEXT V
300 NEXT W
```



BASIC was once the starter language that early 8-bit home computer users learned.

```
print(HANGMAN[0])
attempts = len(HANGMAN) - 1

while (attempts != 0 and "-" in word_guessed):
    print("\nYou have {} attempts remaining".format(attempts))
    joined_word = "".join(word_guessed)
    print(joined_word)

try:
    player_guess = str(input("\nPlease select a letter between A-Z" + "\n> "))
except: # check valid input
    print("That is not valid input. Please try again.")
    continue
else:
    if not player_guess.isalpha(): # check the input is a letter. Also checks a
        print("That is not a letter. Please try again.")
        continue
    elif len(player_guess) > 1: # check the input is only one letter
        print("That is more than one letter. Please try again.")
        continue
    elif player_guess in guessed_letters: # check it letter hasn't been guessed
        print("You have already guessed that letter. Please try again.")
        continue
    else:
        pass

guessed_letters.append(player_guess)

for letter in range(len(chosen_word)):
    if player_guess == chosen_word[letter]:
        word_guessed[letter] = player_guess # replace all letters in the chosen

if player_guess not in chosen_word:
```



Python is a more modern take on BASIC, it's easy to learn and makes for an ideal beginner's programming language.



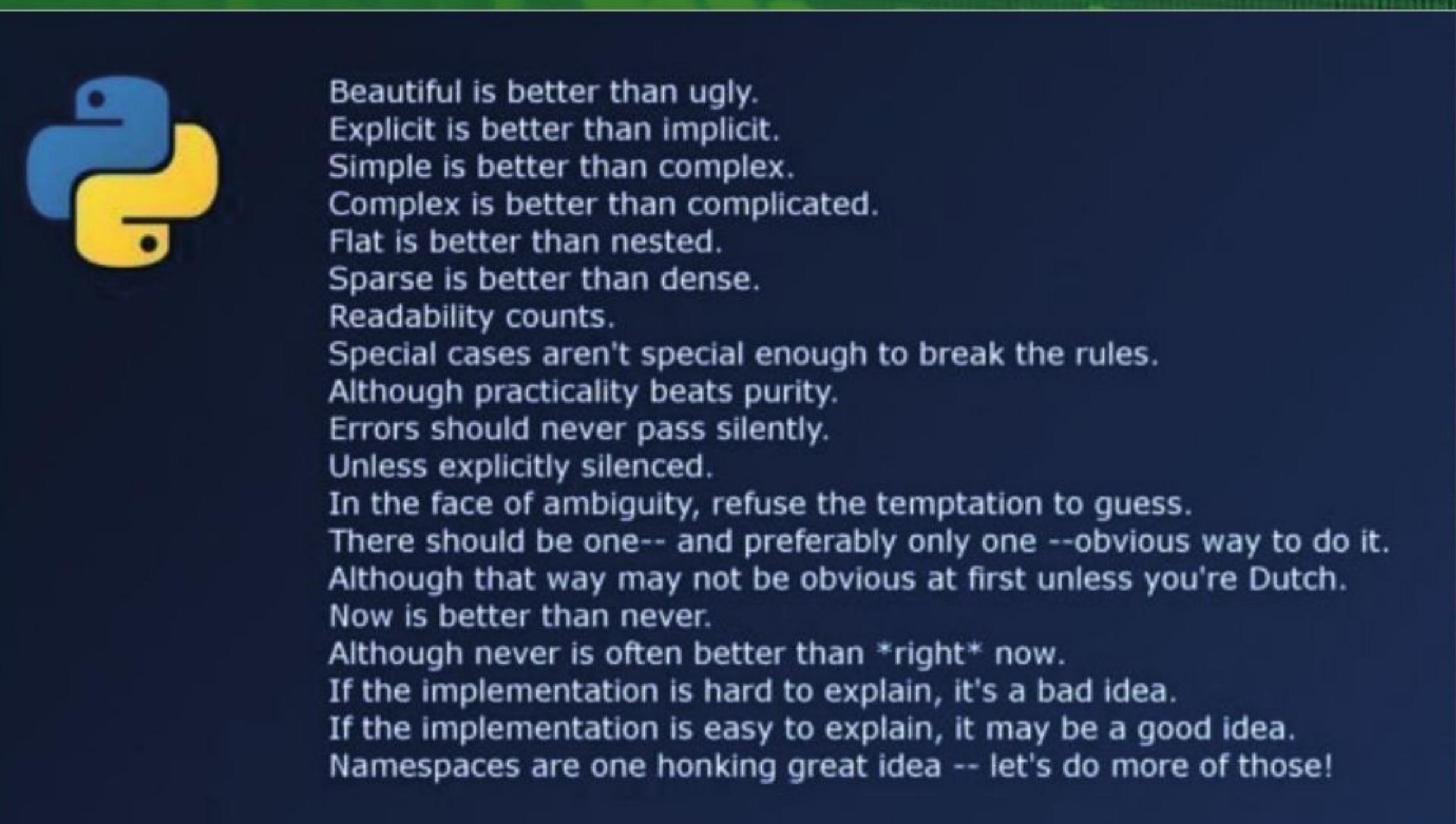
What can you do with Python?

Python is an open-source, object-oriented programming language that's simple to understand and write with, yet also powerful and extremely malleable. It's these characteristics that help make it such an important language to learn.

Python's ability to create highly readable code within a small set of instructions has a considerable impact on our modern digital world. From the ideal, first programmers' choice to its ability to create interactive stories and games; from scientific applications to artificial Intelligence and web-based applications, the only limit to Python is the imagination of the person coding in it.

It's Python's malleable design that makes it an ideal language for many different situations and roles. Even certain aspects of the coding world, that require more efficient code, still use Python. For example, NASA utilises Python both as a stand-alone language and as a bridge between other programming languages. This way, NASA scientists and engineers are able to get to the data they need without having to cross multiple language barriers; Python fills the gaps and provides the means to get the job done.

You'll find lots of examples of this, where Python is acting behind the scenes. This is why it's such an important language to learn.

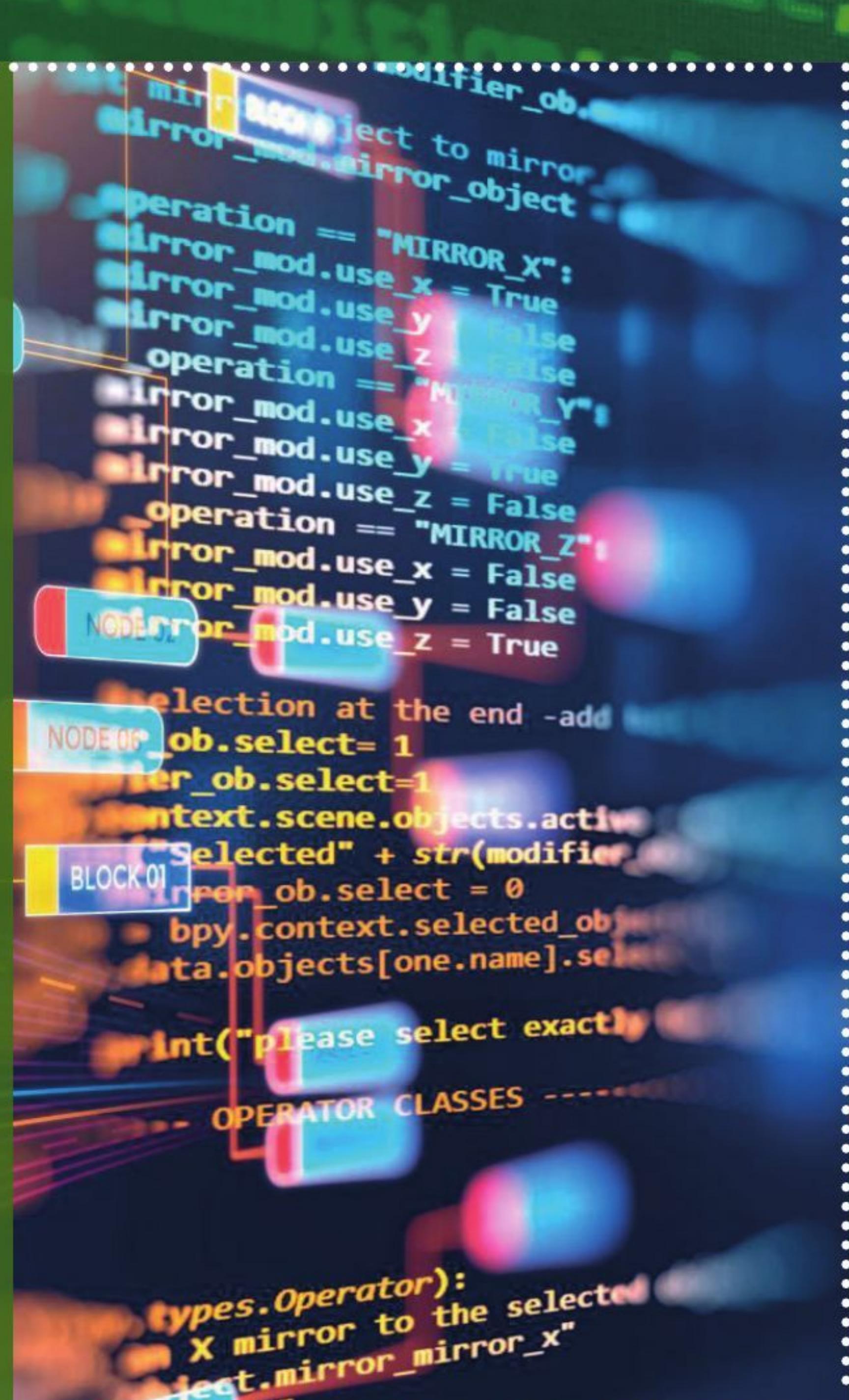


BIG DATA

Big data is a buzzword you're likely to have come across in the last couple of years. Basically, it means extremely large data sets that are available for analysis to reveal patterns, trends and interactions between humans, society and technology. Of course, it's not just limited to those areas, big data is currently being used in a variety of industries, from social media to health and welfare, engineering to space exploration and beyond.

Python plays a substantial role in the world of big data. It's extensively used to analyse huge chunks of the available big data and extract specific information based on what the user/company requires from the wealth of numbers present. Thanks to an impressive set of data processing libraries, Python makes the act of getting to the data, in amongst the numbers, that counts and presenting it in a fashion that's readable and useable for humans.

There are countless libraries and freely available modules that enable fast, secure and more importantly, accurate processing of data from the likes of supercomputing clusters. For example, CERN uses a custom Python module to help analyse the 600 million collisions per second that the LHC produces. A different language handles the raw data, but Python is present to help sift through the data so scientists can get to the content they want without the need to learn a far more complex programming language.



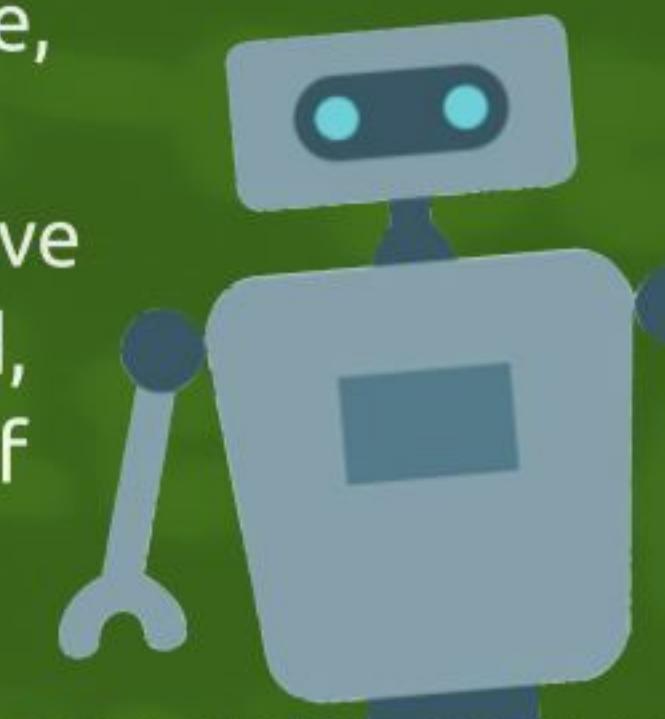
ARTIFICIAL INTELLIGENCE

Artificial Intelligence and Machine Learning are two of the most groundbreaking aspects of modern computing. AI is the umbrella term used for any computing process wherein the machine is doing something intelligent, working and reacting in similar ways to humans. Machine Learning is a subset of AI and provides the overall AI system with the ability to learn from its experiences.

However, AI isn't simply the creation of autonomous robots intent on wiping out human civilisation. Indeed, AI can be found in a variety of day-to-day computing applications where the 'machine', or more accurately the code, needs to learn from the actions of some form of input and anticipate what the input is likely to require, or do, next.

This model can be applied to Facebook, Google, Twitter, Instagram and so on. Have you ever looked up a celebrity on Instagram and then discovered that your searches within other social media platforms are now specifically targeted toward similar celebrities? This is a prime example of using AI in targeted advertising and behind the code and algorithms that predict what you're looking for, is Python.

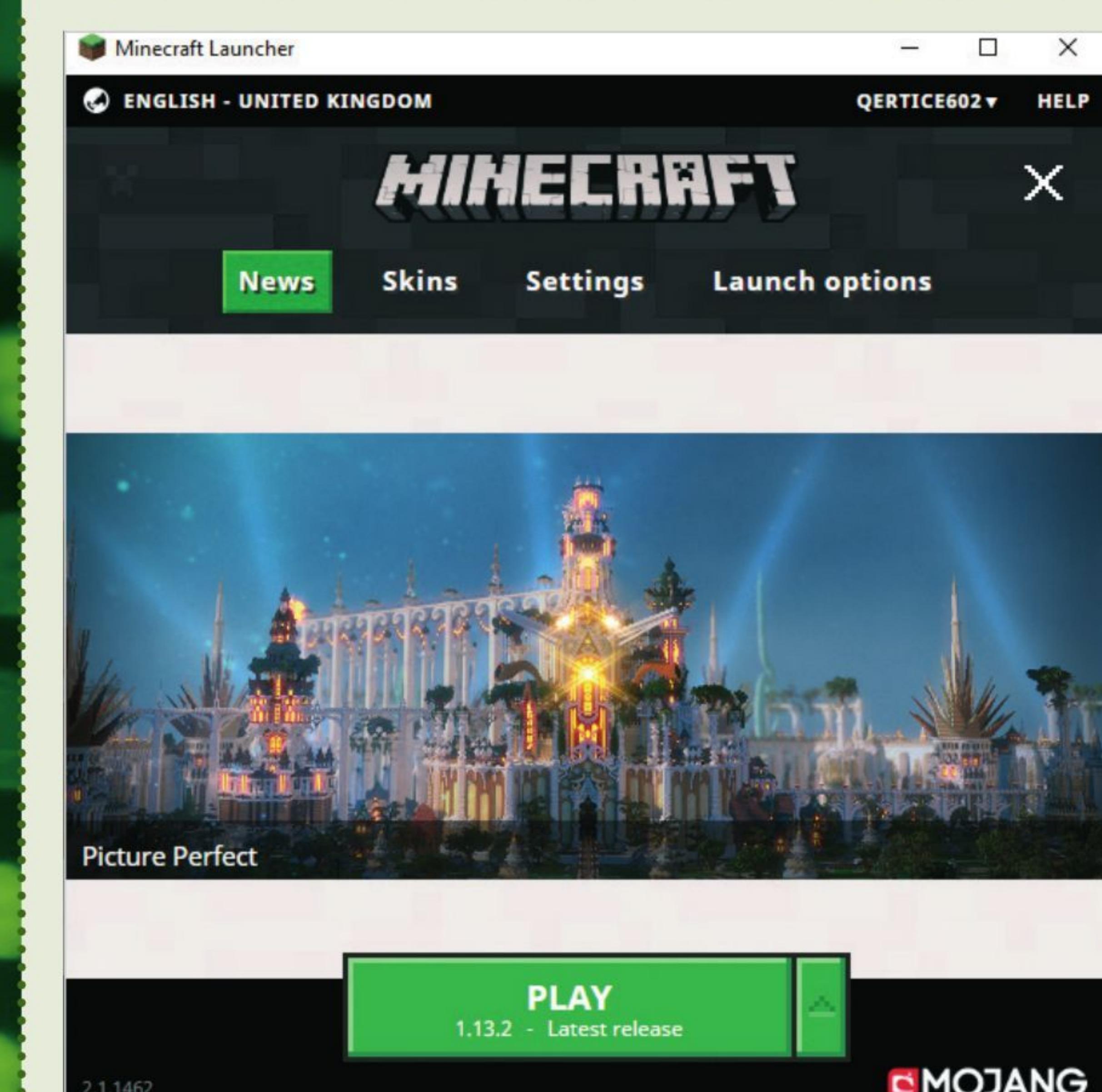
Spotify, for example, uses Python based code, among other things, to analyse your musical habits and offer playlists based on what you've listened to in the past. It's all clever stuff and, moving forward, Python is at the forefront of the way the Internet will work in the future.



WEB DEVELOPMENT

Web development has moved on considerably since the early days of HTML scripting in a limited text editor. The many frameworks and web management services available now means that building a page has become increasingly complex.

With Python, the web developer has the ability to create dynamic and highly secure web apps, enabling interaction with other web services and apps such as Instagram and Pinterest. Python also allows the collection of data from other websites and even apps built within other websites.



GAMING

Although you won't find too many triple-A rated games coded using Python, you may be surprised to learn that Python is used as an extra on many of the high-ranking modern games.

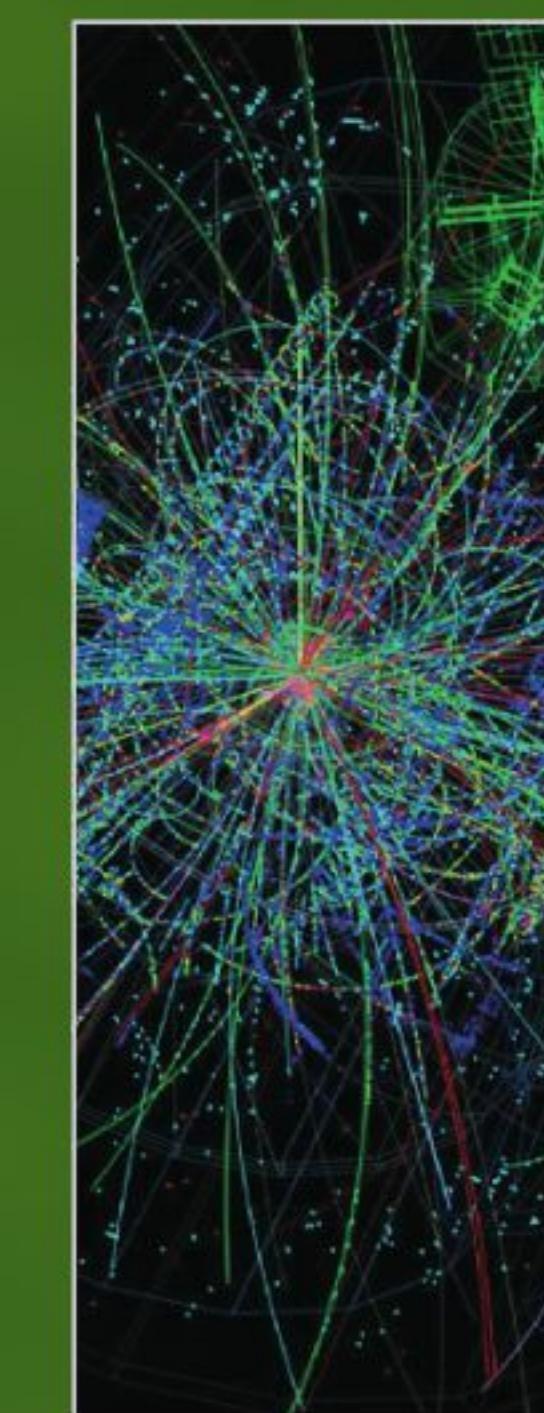
The main use of Python in gaming comes in the form of scripting, where a Python script can add customisations to the core game engine. Many map editors are Python compatible and you will also come across it if you build any mods for games, such as The Sims.

A lot of the online, MMORPGs (Massively Multiplayer Online Role-Playing Games) available utilise Python as a companion language for the server-side elements. These include: code to search for potential cheating, load balancing across the game's servers, player skill matchmaking and to check whether the player's client-side game matches the server's versions. There's also a Python module that can be included in a Minecraft server, enabling the server admin to add blocks, send messages and automate a lot of the background complexities of the game.

PYTHON EVERYWHERE

As you can see, Python is quite a versatile programming language. By learning Python, you are creating a well-rounded skillset that's able to take you into the next generation of computing, either professionally or simply as a hobbyist.

Whatever route you decide to take on your coding journey, you will do well to have Python in your corner.





PYTHON in NUMBERS

There's a lot to like about Python, but don't just take our word for it. Here are some amazing facts and figures surrounding one of the most popular programming languages of recent years.

Python creator Guido Van Rossum named Python after reading scripts from Monty Python's Flying Circus.



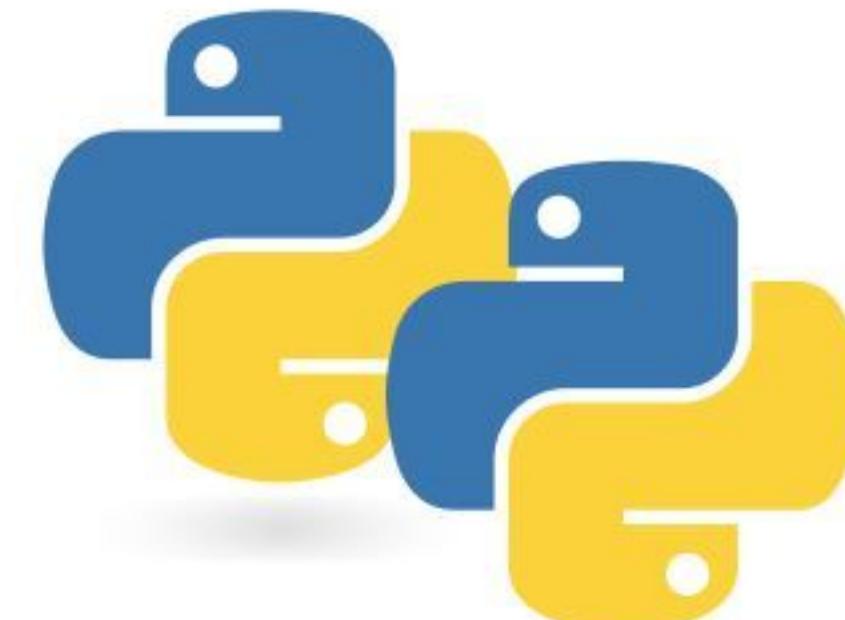
Alexa, Amazon's Virtual Personal Assistant, uses Python to help with speech recognition.



As of the end of 2018, Python was the most discussed language on the Internet.



.....
PYTHON AND LINUX SKILLS ARE THE THIRD MOST POPULAR I.T. SKILLS IN THE UK.



Data analysis and Machine Learning are the two most used Python examples.



Disney Pixar uses Python in its Renderman software to operate between other graphics packages.



OVER 75% OF RECOMMENDED CONTENT FROM NETFLIX IS GENERATED FROM MACHINE LEARNING – CODED BY PYTHON.

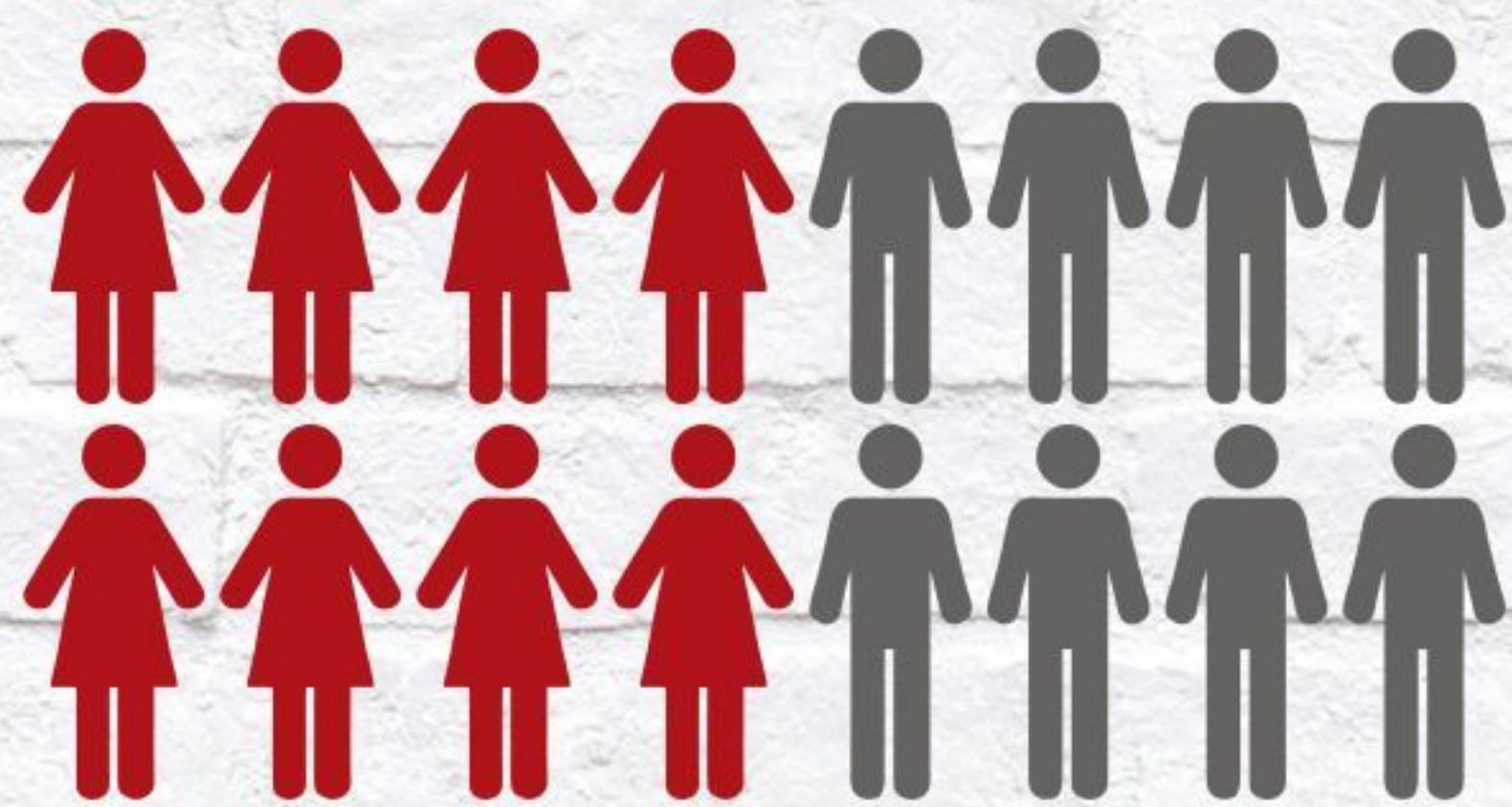


90% OF ALL FACEBOOK POSTS ARE FILTERED THROUGH PYTHON-CODED MACHINE LEARNING.



IT'S ESTIMATED THAT OVER 75% OF NASA'S WORKFLOW AUTOMATION SYSTEMS ON-BOARD THE I.S.S. USE PYTHON.

16,000



There are over 16,000 Python jobs posted every six months in the UK.

PYTHON SKILL-BASED POSITIONS ARE THE

16th

MOST SOUGHT-AFTER JOBS IN THE UK.



Python Data Science is thought to become the most sought-after job in the coming years.



Google is the top company for hiring Python developers, closely followed by Microsoft.



Data Science, Blockchain and Machine Learning are the fastest growing Python coding skills.



New York and San Francisco are the top Python developer cities in the world.



Python developers enjoy an average salary of

£60,000



95% OF ALL BEGINNER CODERS START WITH AND STILL USE, PYTHON AS THEIR PRIMARY OR SECONDARY LANGUAGE.



75% OF ALL PYTHON DEVELOPERS USE PYTHON 3, WHEREAS 25% STILL USE THE OUTDATED PYTHON 2 VERSION.



79% OF ALL PROGRAMMERS USE PYTHON REGULARLY, 21% USE IT AS A SECONDARY LANGUAGE.



49% OF WINDOWS 10 DEVELOPERS USE PYTHON 3 AS THEIR MAIN PROGRAMMING LANGUAGE.