

PYTHON

INTERESTING FACTS

Nomenclature fact of Python

This name 'Python' is extracted from a British comedy series, "Monty Python's Flying Circus". It is **not named** a **snake**. It is said that this was the **favourite series** of its **inventor Guido Van Rossum**. He named it Python because it is **short, mysterious** and **unique**.

Python follows chain comparison

In Python, there can be **multiple comparisons** at once. It is able to **check multiple conditions** at the **same time**.

While in **other programming languages**, you cannot program a **chain of comparison operators**. The **comparison operators** can be **chained randomly**. It **does not** have to follow a **particular order of operators**.

Input:

```
>>> 1<2>1.6
```

Output:

```
True
```

Python does not need compiler

Python relies on an **interpreter**. Unlike other programming languages, it **does not need** a **compiler**. The **code** is **stored** in a **.pyc file**. This **file** acts as a **dynamic engine** for **Python** **eliminating** the **need** of any **compiler**.

Functions in Python can return multiple values

Python offers a feature to **return multiple values** using **function**. It returns the **value as a tuple**. While it is **not possible** with other languages such as **Java, C**, etc.

Input:

```
def x ():  
    name = "John"  
    age = 18  
    return name, age  
  
name, age = x ()  
print(name)  
print(age)
```

Output:

```
John  
18
```

Python does not enhance the use of pointer

In Python, **every program** is done by **reference**. It **does not** support **pointer**.

Python supports function unpacking

It another interesting fact about Python. It allows you to **easily unpack** a **list** or **dictionary** of all the **functions** you have **used** in your **program**.

You can **unpack** a **list** by using ***** and **dictionary** by using ******

Python can implement the 'else' clause within 'for' loop

Unlike other languages, Python is the only language that can use **else within** for a **loop**. This will be true only when the **loop exists naturally** and **do not break in between**.

```
while True:  
    pass  
else:  
    pass
```

The Zen of Python

Tim Peters, a major contributor to the Python community, wrote this poem to highlight the philosophies of Python. If you type in "import this" in your Python IDLE, you'll find this poem:

```
>>> import this
The Zen of Python, by Tim Peters

Beautiful is better than ugly.
Explicit is better than implicit.
Simple is better than complex.
Complex is better than complicated.
Flat is better than nested.
Sparse is better than dense.
Readability counts.
Special cases aren't special enough to break the rules.
Although practicality beats purity.
Errors should never pass silently.
Unless explicitly silenced.
In the face of ambiguity, refuse the temptation to guess.
There should be one-- and preferably only one --obvious way to do it.
Although that way may not be obvious at first unless you're Dutch.
Now is better than never.
Although never is often better than *right* now.
If the implementation is hard to explain, it's a bad idea.
If the implementation is easy to explain, it may be a good idea.
Namespaces are one honking great idea -- let's do more of those!
>>>
```

No braces

Unlike Java and C++, Python does not use braces to delimit code. Indentation is mandatory with Python. If you choose to import it from the `__future__` package, it gives you a witty error.

```
>>> from __future__ import this
SyntaxError: not a chance
```

Flavours of Python

Python ships in various flavours:

- **CPython**- Written in C, most common implementation of Python
- **Jython**- Written in Java, compiles to bytecode
- **IronPython**- Implemented in C#, an extensibility layer to frameworks written in .NET
- **Brython**- Browser Python, runs in the browser
- **RubyPython**- Bridge between Python and Ruby interpreters
- **PyPy**- Implemented in Python
- **MicroPython**- Runs on a microcontroller

Python supports multiple assignments in one statement

Python will let you assign the same value to multiple variables in one statement. It will also let you assign values to multiple variables at once.

This also means swapping in Python is quicker and can be done in only 1 line of code:

```
>>> a, b = 'Hello', 'World'
>>> print(a,b)
Hello World
```

With slicing, it's easier to reverse a list

If we slice a list of values from starting to end but with a step of -1, we get the list right to left (reversed).

```
>>> 'dlroW olleH'[::-1]
Hello World
```

String literals concatenate together

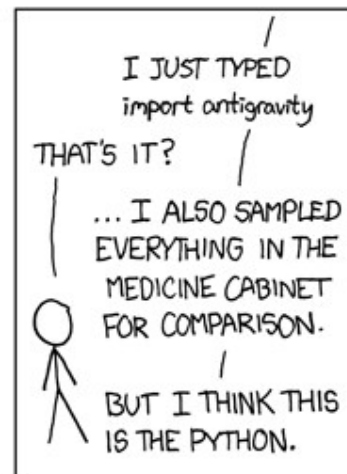
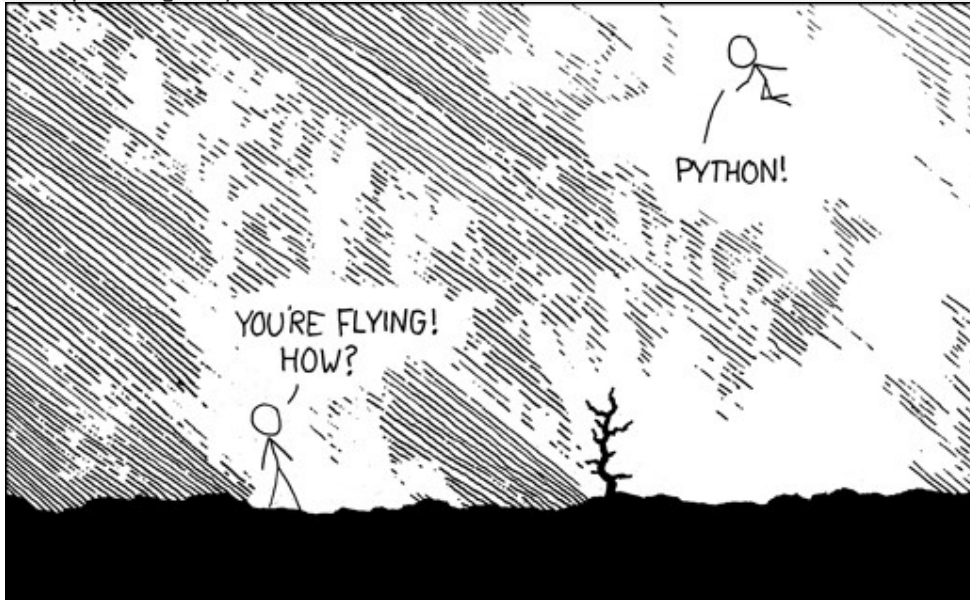
If you type in string literals separated by a space, Python concatenates them together.

```
>>> 'Hello' 'World'
Hello World
```

Antigravity!

If you get to the IDLE and type in `import antigravity`, it opens up a [webpage](#) with a comic about the antigravity module.

```
>>> import antigravity
```



Python influenced JavaScript

Python is one of the 9 languages that influenced the design of JavaScript. Others include AWK, C, HyperTalk, Java, Lua, Perl, Scheme, and Self.

Python Popularity

Python is the fastest-growing programming language and is the second-most preferred language for developers. Python also ranked at the #1 position in December 2020 on the PYPL index. **3rd most profitable programming language** in the world.

Python Overtook French

One of the most widely learned foreign languages in the world after English, French is losing its panache to an equally fancy language Python.

Believe it or not, In a survey conducted by the United Kingdom in 2015, Python overtook French to be one of the most popular languages that are taught in primary schools. The results came as a shock as 6 out of 10 parents wanted their children to learn Python instead of French.

Python is Older Than Java

The programming language of Python is very high in demand over the past decade. Many people think that Python is a new language that has attracted and strengthened its listeners.

But here's the interesting truth: Python is an old programming language and much older than Java. Python was first released in 1991 whereas Java was released in 1995. Today, Python is 30 years old but still stands strong for the future.

Python is Slower than Other Languages

Python is indeed a little bit slower than compared to other programming languages like C, C++, Java, etc...because they are compiled languages whereas Python is interpreted. Thus, this gives a slightly slower speed.

Python Can Define *Infinities*

One of the facts that we have been told from the start of our proper education is that infinities are not defined. Well, that's not actually true for Python.

Input:

```
# Positive Infinity
p_infinity = float('Inf')

if 9999999999999999 > p_infinity:
    print("The number is greater than Infinity!")
else:
    print("Infinity is greatest")

# Negative Infinity
n_infinity = float('-Inf')
if -9999999999999999 < n_infinity:
    print("The number is lesser than Negative Infinity!")
else:
    print("Negative Infinity is least")
```

Output:

```
Infinity is greatest
Negative Infinity is least
```