

Python是什么？

正式地进入Python语法学习之前，我们需要搞清楚，什么Python是什么。

康康Python的官网是如何描述Python的吧

Python is a clear and powerful object-oriented programming language, comparable to Perl, Ruby, Scheme, or Java. [Reference](#)

Python是一门简洁强大的面向对象的编程语言，可比与Perl, Ruby, Scheme和Java

好家伙，这一句话里就设计了好几个初学者闻所未闻的东西

啥叫面向对象啊？啥叫简洁强大啊？

真是伤脑筋，再继续介绍何为Python之前我们需要先弄清楚啥叫面向对象

什么是面向对象？

对象的英语其实是object，说成人话就是，物件。一个柜子，一台电脑，一个手机，都是一个物件。

抛开科学的严谨的定义，用一个例子来简单的阐述面向对象是个什么东西。

现在你有一个任务，你要做蛋糕。

如何做一个蛋糕呢？当然是找一个食谱，跟着食谱走，第一步放什么第二步放什么，烤箱烤多久，是水浴法还是什么别的方法，不论过程如何，你都是在跟着食谱走。这就是面向过程，你面对问题时，注重于用一个流程来解决问题

但是除了找一个食谱还有别的办法吗？

当然有，当然你可以直接买一个自动蛋糕机，市面上肯定有只需要你放入鸡蛋和面粉就能烤出蛋糕的机器，不需要你关心如何烘焙，不需要你掌握水浴法之类的高级烘焙技巧，你只需要这一套机器，机器里怎么运作的你不用关心，机器里的具体流程你也不需要关心，ALL you need to do，只是把鸡蛋和面粉放进去

当然你也可以自己做一个全自动蛋糕机，自己去设计这样的一个机器，只要鸡蛋面粉和电源，这样在以后需要做蛋糕的时候就不需要一步一个的去做，不用考虑什么水浴法什么烘焙温度，把这个复杂的流程“集合”在一起，用的时候一键启动。

你观察到区别了吗？一个是需要你关注流程，下一步怎么走，一个完全对你封闭了流程，你只需要专注于食材的投入

这就是面向过程和面向对象。

面向对象要求你专注于事物本身，面向过程要求你专注于过程。

那么假设都是从0开始做蛋糕，面向对象的编程就是在努力的生产一个蛋糕机，生产的过程似乎和做蛋糕没什么关系，但是通过生产蛋糕机这个对象，你解决了做蛋糕的问题，你专注的是设计蛋糕机这个对象。

面向过程的编程，要求你专注于做蛋糕而不是转头做蛋糕机。

但是事实上面向对象和面向过程**并不是**完全对立的两个模式

比如说你做这个蛋糕是为了朋友的生日派对，蛋糕只是整个派对流程的一个环节，准备整个派对也有自己的过程

气球的选购，蛋糕的制作，制作邀请函

各个环节组成了整个准备过程，每个环节中有不同的人不同的机器来完成

邀请函可以用打印机打印，你不需要关心打印机是如何把字写在纸上的

蛋糕可以用蛋糕机，你不需要关注烤箱要多少度

过程是永远逃脱不掉的，只是引入工具，比如蛋糕机，打印机帮助你简化了封装了重复的工作

打印机这次用完了下次可以继续用

蛋糕机也不是用完就丢了

这些“封装”好的功能不仅帮助你快速的解决问题，还可以允许你下次再继续用

这就是所谓的面向对象

面向对象的编程最后都需要一个使用对象的过程，比如你设计好的蛋糕机总要你插上电源，放入鸡蛋和面粉

面向过程的编程可以用一个个的对象也就是一个个的物品来简化其过程。有蛋糕机做蛋糕和空手做蛋糕比，蛋糕机的效率更高一些。

两个是相辅相成的，并不是独立的。

所以事实，作者认为，并不存在一个完全面向过程的语言，一个语言是否面向对象其实说的是他对面向对象的支持程度

Python java c++都有class关键字，帮助你快速的构建蛋糕机

C虽然没有class关键字但是并不影响你构建蛋糕机

说白了，只是两种不同的思维模式，和语言关系不大

一个语言说自己面向对象，应该是指该语言对面向对象的编程有着某种程度上的支持

java这种号称面向对象的语言最后还是需要有一个Main函数作为入口执行一定的流程

C这种“强调”main函数的语言也可以通过头文件来实现封装，你当然可以写一个 `蛋糕机.h` 和 `蛋糕机.c` 来实现封装

不论如何我希望读者你能明白，面向对象只是一种思想，一种解决问题时的思想

如何理解这种思想，需要你日积月累的学习，作者到这里的解释也只是浅薄的论述

更加科学更加严谨的面向对象可以参考[Wikipedia面向对象](#)

为什么说Python简洁强大？

Python的语法简单，很简单

举个例子，同样是使用 `for` 循环的方式在屏幕上打印出0 – 9十个数字

C++则需要你这样写：

```
1  #include<iostream>
2  int main(void)
3  {
4      for(int i = 0; i < 10; i++)
5          std::cout << "i = " << i << endl;
6      return 0;
7  }
```

`cout` 是 `iostream` 里的一个对象，一个对象就是一个物品，这个物品就像是电脑or手机一样，这个物品的功能就是往控制台上输出一些字或者简单的说，就是在你的屏幕上打印出一些字

```
1  #include "stdio.h"
2  int main(void)
3  {
4      for (int i = 0; i < 10; i++)
5          printf("%d\n",i);
6      return 0;
7  }
```

Python则需要你这样写

```
1  for i in range(10):
2      print(i)
```

代码长度上Python就短了许多，而且没有任何编程经历的人都可以看得出，Python这里写起来就像是一篇英语作文

没错，Python 的代码阅读起来像是英文的小说，易于阅读，没有过多的晦涩难懂的写法，所以说Python简洁易懂

有人就说了，简洁有什么用？

简洁即强大，简洁而不简单。

人生苦短，我用Python

好啦，到此为止我们就简单的介绍了Python是一个什么语言，Python的内容其实非常多，绝非一篇文章就能搞懂的，我们会在后续的教程中逐渐完善对Python的介绍，总之到此为止，你需要知道

- Python简洁而强大
- Python是一门面向对象的编程语言

阅读更多的Python介绍可以参考

[Python Beginners Guide](#)