# 1 Python

## 标准数据类型

图1



### 1.1.1 列表（序列）

列表：列表是有序的元素的集合。

图1



#### 切片

切片的结果包含切片的第一个元素，不包含第二个元素。

图1



#### None

None表示什么都没有。

图1



#### in（成员资格）

in判断一个元素是否在列表中。

图1



### 1.1.2 元组

元组：

图2



（注意：列表是可以修改的，元组是不能修改的）

### 1.1.3 字典

字典：字段类似于Java的Map，是key/value的结构。

图3

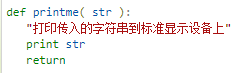


## 函数

图1



图2



### 匿名函数

python中的匿名函数用lamba来创建。

图1



图2



### 无返回值的函数

函数如果无返回值，则返回值是None。

图1



图2



## 可变和不可变对象

在python中，strings，tuples和numbers是不可变的对象，list和dict是可变的对象。

图1



### 可变对象和不可变对象的参数传递

不管是可变对象和不可变对象，传递的都是对象的地址。但当尝试去修改一个不可变的对象时，因为该对象不可变，所以只能创建一个新的对象，然后让引用指向这个新对象。

图1



图2



## 全局变量和局部变量

函数内是局部变量，函数外是全局变量。

图1



## 模块（Module）

在python中，一个模块就是一个Python文件（\*.py文件）。

图1



### 引入模块

模块定义好后，可以用import引入模块。

图1



图2



### 引入模块的一部分函数

模块定义好后，可以用from...import...引入模块的一部分函数。

图1（从模块math中导入sqrt函数）



### 使用模块中的函数

使用import导入模块后，使用module.function的方式调用模块的函数。

图1



## #! /usr/bin/env python

如果不加#! /usr/bin/env python，则必须指定python解释器来执行\*.py脚本，比如c:\python hello.py。

但UNIX支持在py脚本的第一行用#! /usr/bin/env python指定python的绝对路径后，不管python库在什么地方，都不需要再指定python解释器，直接hello.py运行py脚本。

当同一台就机器安装了多个python版本时，使用python3/python2代替python。