



# کارگزاری مفید

پیشرفت مداوم

تابستان ۱۴۰۱

علی اسفندیارپور

آموزش زبان برنامه نویسی پایتون

زبان برنامه‌نویسی و تاریخچه پایتون

۱

نحوه نصب و فضای برنامه‌نویسی پایتون

۲

انواع دیتاها در پایتون

۳

متغیرها در پایتون

۴

توابع در پایتون

۵

if در پایتون

۶

for در پایتون

۷

ماژول‌ها در پایتون

۸

Numpy در پایتون

۹

استثناها در پایتون

۱۰

## پایتون چه بود و از کجا آمد ؟

### Guido van Rossum



با تعداد خط کمتری کد بنویسیم!

۱۹۸۰

سالهای ۱۹۸۰ به عنوان یک ساید پراجکت شروع شد.

هدف پروژه جایگزینی برای زبان ABC بود.

۱۹۹۱

اولین نسخه با عنوان پایتون ۰.۹.۰ عرضه شد.

۲۰۰۰

پایتون ۲ جایگزین ورژن قبل شد.

۲۰۰۸

پایتون ۳ وارد شد و در نهایت پایتون ۲ کنار رفت.

۲۰۲۰

به عنوان محبوبترین زبان برنامه نویسی انتخاب شد.

C++: Can not compare float and int  
Python:



BEFORE AND AFTER CODING



java

python

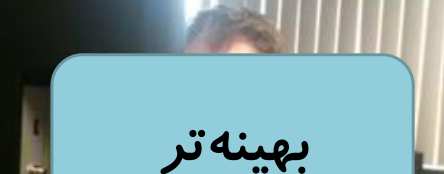
When you switch from C++ to Python



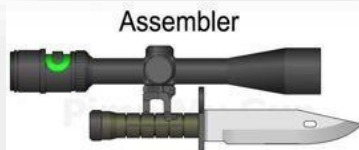
MEMES!



WRITES 10 LINE PYTHON SCRIPT



ELIMINATES THE NEED FOR 5 JOBS



When you try to define constants



When you replace a for loop with a vectorized numpy function and see the speed improvement



## نحوه نصب پایتون

Integrated Development Environment (IDE)

Code Editor

Compiler

Debugger

### Integrated Development Environment

#### Text Editor

Syntax Coloring

Autocomplete

Indexer

Doc Viewer

Code Templates



#### Builder

Compiler

Lexer

Parser

Assembler

Linker



#### Debugger GUI

Debugger



مراجعه به سایت Python.org و دانلود پایتون!

آشنایی با IDE و Compiler

مراجعه به سایت anaconda.com و دانلود آناکوندا!

### ANACONDA.NAVIGATOR

Home

Environments

Learning

Community

Applications on base (root)

Channels



CMD.exe Prompt

0.1.1

Run a cmd.exe terminal with your current environment from Navigator activated

Launch



Datalore

Online Data Analysis Tool with smart coding assistance by JetBrains. Edit and run your Python notebooks in the cloud and share them with your team.

Launch



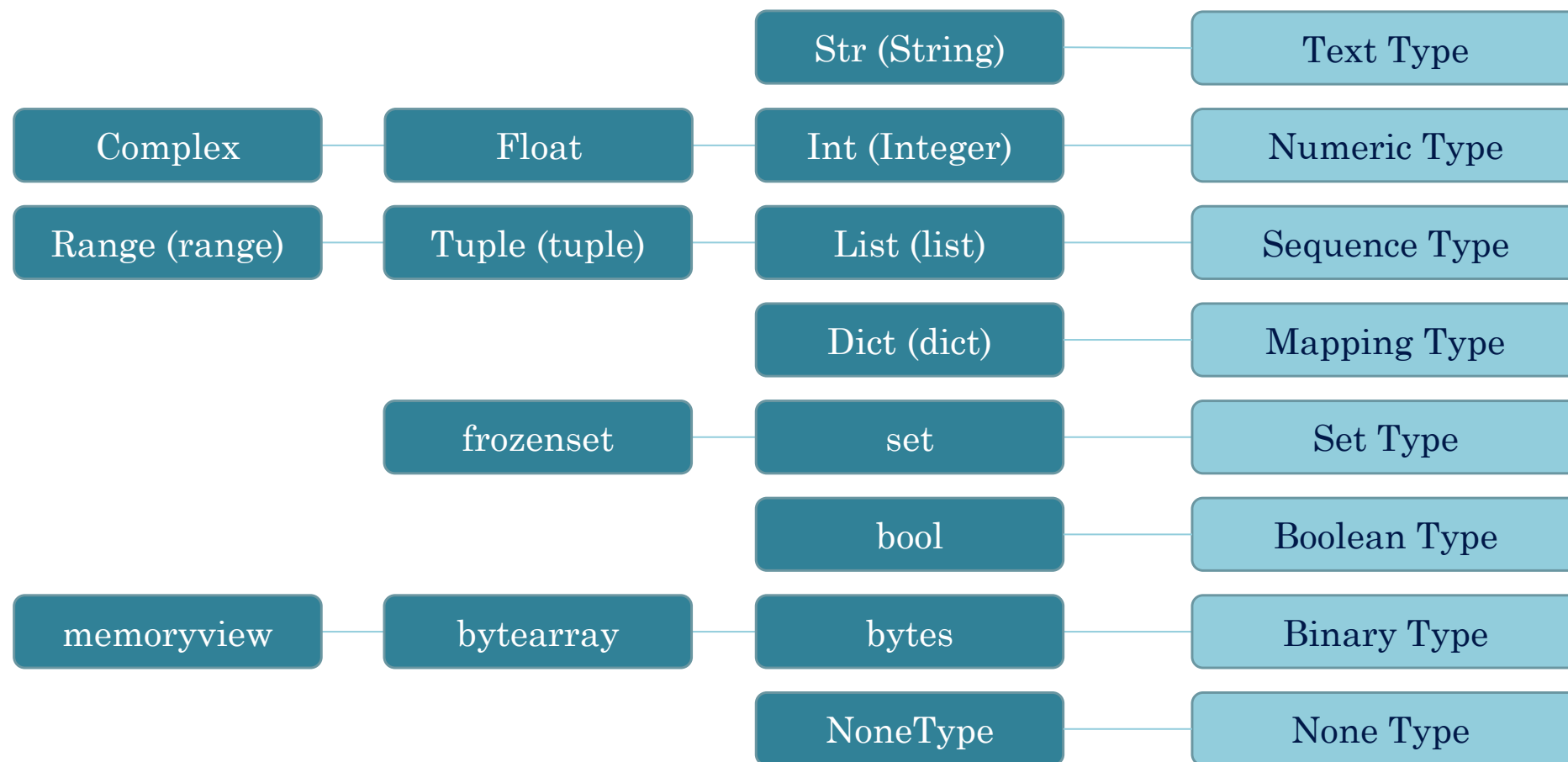
IBM Watson Studio Cloud

IBM Watson Studio Cloud provides you the tools to analyze and visualize data, to clean and shape data, to create and train machine learning models. Prepare data and build models, using open source data science tools or visual modeling.

Launch



## انواع دیتا در پایتون



## انواع دیتا در پایتون

Type	Example
"Mofid"	text
25	int
25.6	float
1j	complex
['a','b','c']	list
('a','b','c')	tuple
Range(5)	range
{'a':"2",'b':"3"}	dict
{'a','b','c'}	set
True	bool
None	NoneType

## خروجی در پایتون و اولین کد پایتون

```
Print ()  
Print ("Welcome to Mofid.")  
Print (4+2)  
Print (4,2)  
Print('Koche', 'Shomare', '2')
```

```
Print (12 ,24 ,2 ,sep=":")  
Print ("Welcome to Mofid.\n")  
Print ("Welcome to Mofid.\n" , end = ' ' )
```

برای بررسی متغیرها و خروجی‌های بخش‌های مختلف از دستور پرینت استفاده می‌شود. علاوه بر این دستور پرینت می‌تواند جهت دیباگ کردن کد هم استفاده شود.



## متغیرها در پایتون

```
a = 5  
Name = "Ali"  
Company = "Mofid"
```

```
a = str(3)  
b = int(3)  
c = float(3)
```

```
Print (Type(a))  
Print (Type(b))  
Print (Type(c))
```

```
a = 'Ali'  
a = "Ali"
```

## اعداد و متدهای اعداد در پایتون

عمل	دستور
جمع	+
ضرب	*
منها	-
تقسیم	/
توان	**
باقیمانده	%
قدر مطلق	abs()

a = "Ali"

دستور	عمل
capitalize()	بزرگ کردن حرف اول
casefold()	کوچک کردن حرف اول
center()	وسط چین کردن حروف
count()	شمارش رشته مشخص در حروف
encode()	مشخص کردن حروف
find()	پیدا کردن ایندکس حروف
replace()	جایگزینی کارکترها
index()	پیدا کردن بر اساس ایندکس
join()	اتصال حروف با کارکتر مشخص
lower()	کوچک کردن حروف
split()	جدا کردن حروف از هم
strip()	حذف کردن فاصله‌ها از حروف

## رشته‌ها و متدهای رشته‌ها در پایتون

دستور	عمل
title()	بزرگ کردن حرف اول همه رشته‌ها
upper()	بزرگ کردن حروف
zfill()	صفر گذاشتن پشت کارکترها
len()	تعداد کارکترهای رشته

format

```
print("Text ... {Name}.format(Name = 'Ali'))
```

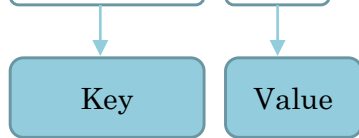
## لیست و متدهای لیست در پایتون

```
my_list = ["a", "b", "c", "d"]
```

عمل	دستور
المنتهای یک لیست (یا اعداد و رشته) را به انتهای المنتهای یک لیست دیگر اضافه می‌کند و یک لیست خروجی می‌دهد.	append()
متغیرهای لیست را پاک می‌کند و یک لیست خالی تحویل می‌دهد.	clear()
یک کپی از لیست می‌سازد و یک لیست تحویل می‌دهد.	copy()
تعداد المنتهای موجود در لیست را نمایش می‌دهد.	count()
یک لیست را به انتهای لیست دیگر اضافه می‌کند.	extend()
پیدا کردن ایندکس المنتهای لیست	index()
اضافه کردن المنت به لیست با ایندکس مشخص	insert()
حذف بر اساس ایندکس	pop()
حذف بر اساس متغیر	remove()
برعکس کردن ترتیب المنتهای لیست	reverse()
مرتب کردن لیست	sort()

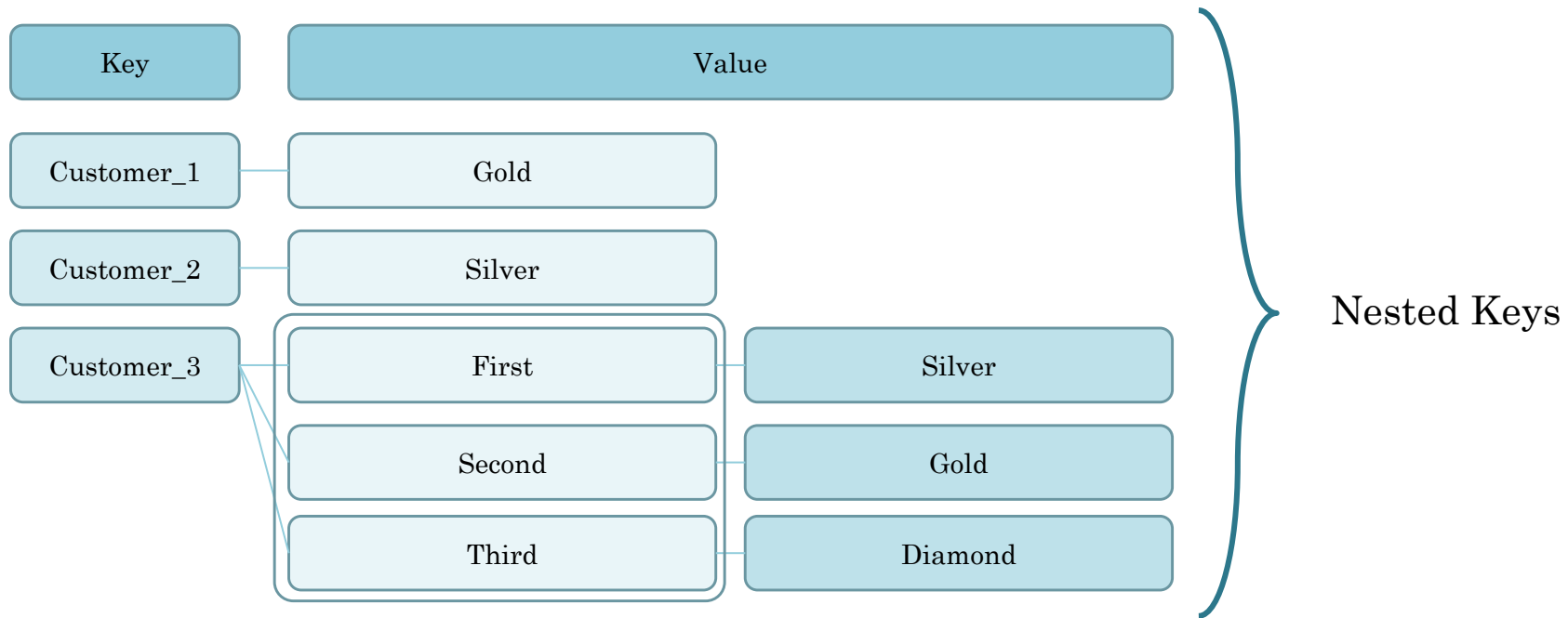
# dic و متدهای dic در پایتون

```
Customers = {'Customer_1': 'Gold', 'Customer_2': 'Ordinary', 'Customer_3': 'Silver'}
```



```
print(Customers["Customer_1"])
```

```
Customers = {'Customer_1': 'Gold', 'Customer_2': 'Silver',  
             'Customer_3': {'first': 'Silver', 'second': 'Gold', 'third': 'Diamond'}}
```



## عملگرها در پایتون

عمل	دستور
آیا برابر است ؟	==
آیا برابر نیست ؟	!=
بزرگتر است ؟	>
کمتر است ؟	<
آیا بزرگتر مساویست ؟	>=
آیا کوچکتر مساویست ؟	<=
آیا هست ؟	in
آیا نیست ؟	not in
آیا هست ؟	is in
و	and
یا	or

## توابع در پایتون

```
Def first_function ():  
    print ("Shomare Eshtebah!")  
first_function()  
  
Def second_function (name):  
    print ("Salam" + name)  
  
Def third_function (name,shahr):  
    print ("Salam" + name + "az" + shahr)  
  
Def fourth_function (adad):  
    return 10*adad
```

Pass

```
Def pass_function ():  
    pass
```

## if در پایتون

```
if condition :  
    ...
```

```
if condition1 :  
    ...  
elif condition2 :  
    ...  
elif condition3 :  
    ...  
.  
.  
.
```

```
if condition :  
    ...  
  
if condition1 :  
    ...  
elif condition2 :  
    ...  
elif condition3 :  
    ...  
.  
.  
.  
  
else :  
    ...
```



## تمرین ۱

تابعی بنویسید که یک لیست دریافت کند. اگر "majid" در لیست بود تایید کند و جایگاه مجید را هم به ما بگوید. اگر مجید در لیست نبود. بگوید مجید نیست!

## تمرین ۲

تابعی بنویسید که یک لیست دریافت کند. اگر "majid" در لیست بود و دقیقا در کنار آن "ali" بود بگوید مجید با علی آمده. اگر مجید بود اما علی یافت نشد بگوید مجید آمده اما علی نیست! اگر مجید در لیست نبود. بگوید مجید نیست!

## for در پایتون

```
for i in range(0,6):  
    print(i)
```

```
for i in range(0,6):  
    if i == 5:  
        continue  
    print(i)
```

```
for x in range(6):  
    print(x)  
else:  
    print("Tamammmmmm!")
```

```
for i in range(0,6):  
    print(i)  
    if i == 5:  
        break
```

```
range(0,10)  
range(4,10)  
range(2, 30, 3)
```

```
count = 0  
while (count < 5):  
    count = count + 1  
    print("kamtare az 5")
```

## تمرین ۳

لیست ۵ سهم اول از ۱۰۰۰ مشتری را داریم. می‌خواهیم بدانیم :

- چه تعداد از این ۱۰۰۰ مشتری سهم فولاد را در میان ۵ سهم اول خود دارند؟
- چه تعداد از این ۱۰۰۰ مشتری سهم کوروش را در میان ۵ سهم اول خود دارند؟
- کدام سهم از میان سهام، بیشترین تکرار را در پرتفوی این ۱۰۰۰ مشتری داشته است؟
- کدام سهم از میان سهام، کمترین تکرار را در پرتفوی این ۱۰۰۰ مشتری داشته است؟

## ماژول‌ها در پایتون

شامل دستورات و تعاریف مشخصی در فضای پایتون است که برای استفاده در قسمت‌های مختلف کد مورد استفاده قرار می‌گیرد. تفاوت آن با تابع در این است که در ماژول می‌توان به مجموعه‌ای از توابع دسترسی داشت و این توابع را استفاده کرد.

```
import module_name
```

```
from module_name import function
```

# Numpy در پایتون

## استثناها در پایتون

### تمرین ۳

با استفاده از تابع رندوم ۱۰۰۰۰ کد ۱۰ رقمی ایجاد کنید. سپس ببینید از میان ۱۰۰۰۰ کد ایجاد شده چند کد استاندارد کد ملی را دارا هستند و می‌توانند به عنوان کد ملی در نظر گرفته شوند.