|  |
| --- |
| 格式化输出 |
| **作者：李昂**  **归档：python之路day1**  **日期：2018/1/29** |
| **快捷键：**  Ctrl + 1 标题1  Ctrl + 2 标题2  Ctrl + 3 标题3  Ctrl + 4 实例  Ctrl + 5 程序代码  Ctrl + 6 正文  Ctrl + q 正文注释 |
| **格式说明：**  蓝色字体：注释  黄色背景：了解  绿色背景：重要  红色背景：特别重要 |

目 录

[格式化输出 I](#_Toc504542468)

[第1章 传参 1-2](#_Toc504542469)

[第2章 格式化输出 2-2](#_Toc504542470)

[2.1 字符串拼接 2-2](#_Toc504542471)

[2.2 直接引用 2-3](#_Toc504542472)

[2.2.1 强制类型转换 2-4](#_Toc504542473)

[2.2.2 修改字符类型 2-4](#_Toc504542474)

[2.3 修改变量类型为字符型---str 2-6](#_Toc504542475)

[2.4 直接引用变量-- 指定参数 2-7](#_Toc504542476)

[2.5 使用顺序 获取参数 2-8](#_Toc504542477)

# 传参

python传参可以使用input 从标准输入获取参数

*Author：LiAng*

*#encoding=utf8*

username = input(**"username:"**)

password = input(**"password:"**)

print(**"usernameis"**,username,**"，passwordis"**,password,**',,,,,,,,'**)

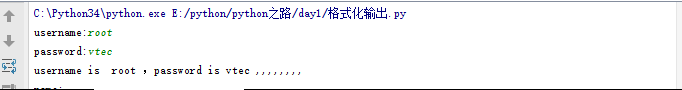
*#print 输出时 常量用引号引起来，变量直接写变量名,变量除外的都要用引号标识出来，引号内username：是要输出的东西，填写的参数会直接赋值给username*

*#input标识从标准输入读取数据*

变量前后要有逗号

对于常量和符号在print中要用“”，变量可以直接写。

input： 从标准输入获取参数



input就是指定这个变量是来自标准输入。

# 格式化输出

## 字符串拼接

name = input(**"name:"**)

age = input(**"age:"**)

job = input(**"job:"**)

salary = input(**"salary:"**)

info = **'''**

**--------------infoof'''**+name+**'''------------**

**Name:'''**+name+**'''**

**Age:'''**+age+**'''**

**Job:'''**+job+**'''**

**'''**

print(info)

使用'''+ +''' 来标识变量

info Of Li 
Name : Li 
Age: 23 
Job: aaa 

+ 拼接会使用多块内存， 效率低。尽量不要使用。

## 直接引用

name = input(**"name:"**)

age = input(**"age:"**)

job = input(**"job:"**)

salary = input(**"salary:"**)

info = **'''**

**----------info of %s----------**

**Name:%s**

**Age:%s**

**Job：%s**

**Salary:%s**

**'''**%(name,name,age,job,salary)

print(info)

%s 代表字符型， 这里的%s 可以理解为占位符同时表示变量类型， 表示这是一个变量， 顺序和后面%（）， 内的变量名对应。%（） 内的变量名是前面定义的。

ane: Li 
e:23 
Job; aaa 
Salary: 400000 

## 常用输出方式

*# Author：Li Ang  
# encoding=utf8*name **=** input("Name:")  
age **=** int(input("Age:"))  
job **=** input("Job:")  
salary **=** input("Salary:")  
info **=** '''  
-------------- info of {\_Name} -----------------  
Name: {\_Name}  
Age : {\_Age}  
Job : {\_Job}  
Salary : {\_Salary}  
'''.format(\_Name**=**name,  
 \_Age**=**age,  
 \_Job**=**job,  
 \_Salary**=**salary)  
print(info)



#字符类型

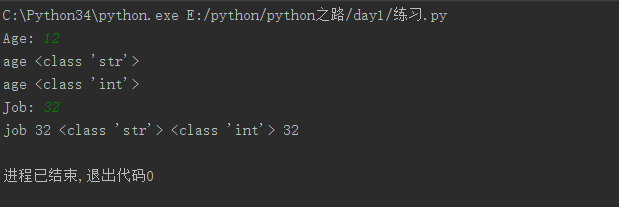
**%s字符型**

**%d整型**

**%f浮点型**

### 强制类型转换

age = input("Age: ") #标准输入一个变量，默认数据类型为字符串  
print ("age",type(age)) # 打印age 及age的数据类型  
age = int(age) #将age以整形方式赋值给age  
print ("age",type(age))  
  
job = str(input("Job: ")) #定义一个字符串类型的变量  
print ("job",job,type(job),type(int(job)),job) #将字符串job转换为整形。



### 修改字符类型

*#Author：LiAng*

*#encoding=utf8*

name = input(**"name:"**)

age = input(**"age:"**)

job = input(**"job:"**)

salary = input(**"salary:"**)

info = **'''**

**--------------info%s---------------**

**name:%s**

**age:%d**

**job:%s**

**salary:%s**

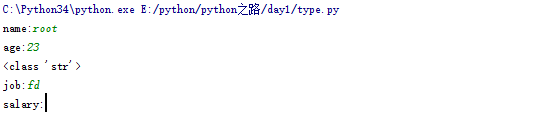
**'''**%(name,name,age,job,salary)

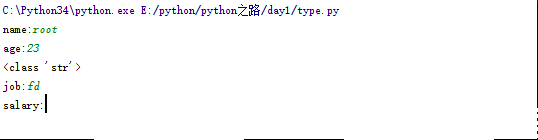
job: 
F il. 14. 
n.e, job, 

age 定义的为整形？

查看变量类型

print (type(age))





age的实际字符类型为字符型。默认定义的变量都是字符型。

修改age类型为整型-- int

*#Author：LiAng*

*#encoding=utf8*

name = input(**"name:"**)

age = int(input(**"age:"**))

print(type(age))

job = input(**"job:"**)

salary = input(**"salary:"**)

info = **'''**

**--------------info%s---------------**

**name:%s**

**age:%d**

**job:%s**

**salary:%s**

**'''**%(name,name,age,job,salary)



输出正常。此时的age为整形，如果填入字符串则会报错。

int() ---------定义变量为整形。int 代表integer 整形。

## 修改变量类型为字符型---str

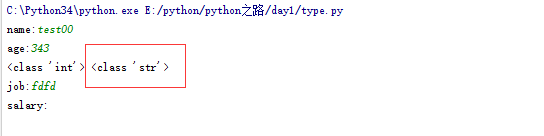
*#Author：LiAng*

*#encoding=utf8*

name = input(**"name:"**)

age = int(input(**"age:"**))

print(type(age),type(str(age))) #str(age) 设置age为字符型，type 输出age类型



*#Author：LiAng*

*#encoding=utf8*

name = input(**"name:"**)

age = int(input(**"age:"**))

print(type(age),type(str(age)))

print(type(age))

job = input(**"job:"**)

salary = input(**"salary:"**)

info = **'''**

**--------------info%s---------------**

**name:%s**

**age:%d**

**job:%s**

**salary:%s**

**'''**%(name,name,age,job,salary)

print(info)

0 № ： 鉍 Ip № “ · E ；/“：№n/”t№a之谘／血y丷0}@ “ 
int' 》 〈 el 飞 
其 实 age 还 是 整 型 

## 直接引用变量-- 指定参数

*#Author：LiAng*

*#encoding=utf8*

name = input(**"Name:"**)

age = int(input(**"Age:"**))

job = input(**"Job:"**)

salary = input(**"Salary:"**)

info = **'''**

**--------------infoof{\_Name}-----------------**

**Name:{\_Name}**

**Age:{\_Age}**

**Job:{\_Job}**

**Salary:{\_Salary}**

**'''**.format(\_Name=name,

\_Age=age,

\_Job=job,

\_Salary=salary)

print(info)

#这里可以{} 可以间接的表示一个变量， .format 使{} 内的名称关联到前面定义的变量。

E /pythWvytiwn2Wa•yt/typ•, vy 
S.l«y fdf 

## 使用顺序 获取参数

*#Author：LiAng*

*#encoding=utf8*

name = input(**"Name:"**)

age = int(input(**"Age:"**))

print(type(age))

job = input(**"Job:"**)

salary = input(**"Salary:"**)

info2 = **'''**

**---------------infoto{0}--------------**

**Name:{0}**

**Age:{1}**

**Job:{2}**

**Salary:{3}**

**'''**.format(name,age,job,salary)

#使用{0} 对应.format() 中的变量名称。

print(info2)

E 'i nf.2 PY 
cl„z ' ;nt') 
S.l«y 