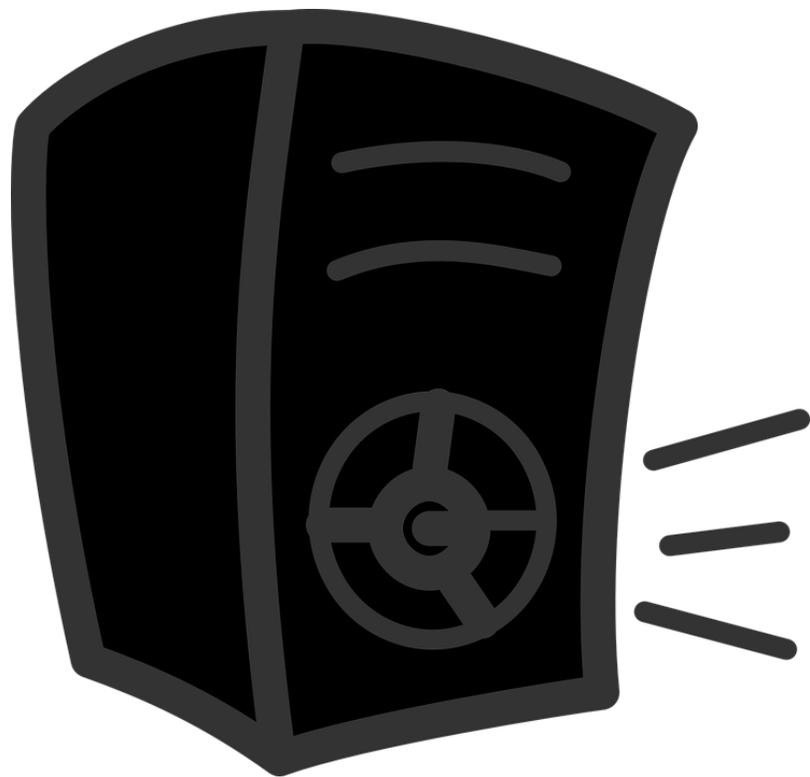


Python

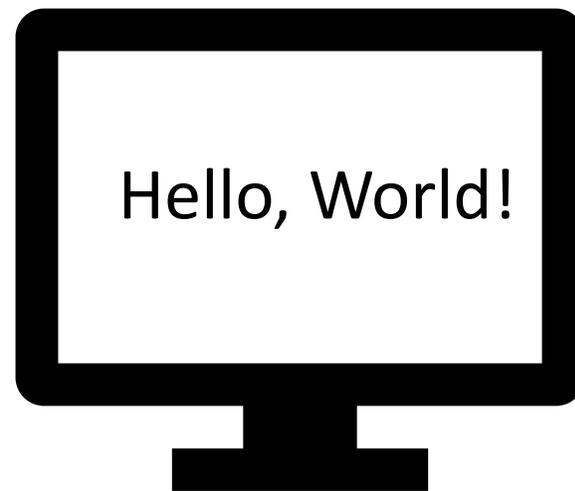
輸出語句

輸出的目的之一：在螢幕輸出文字

讓電腦用喇叭發出聲音



在螢幕輸出文字

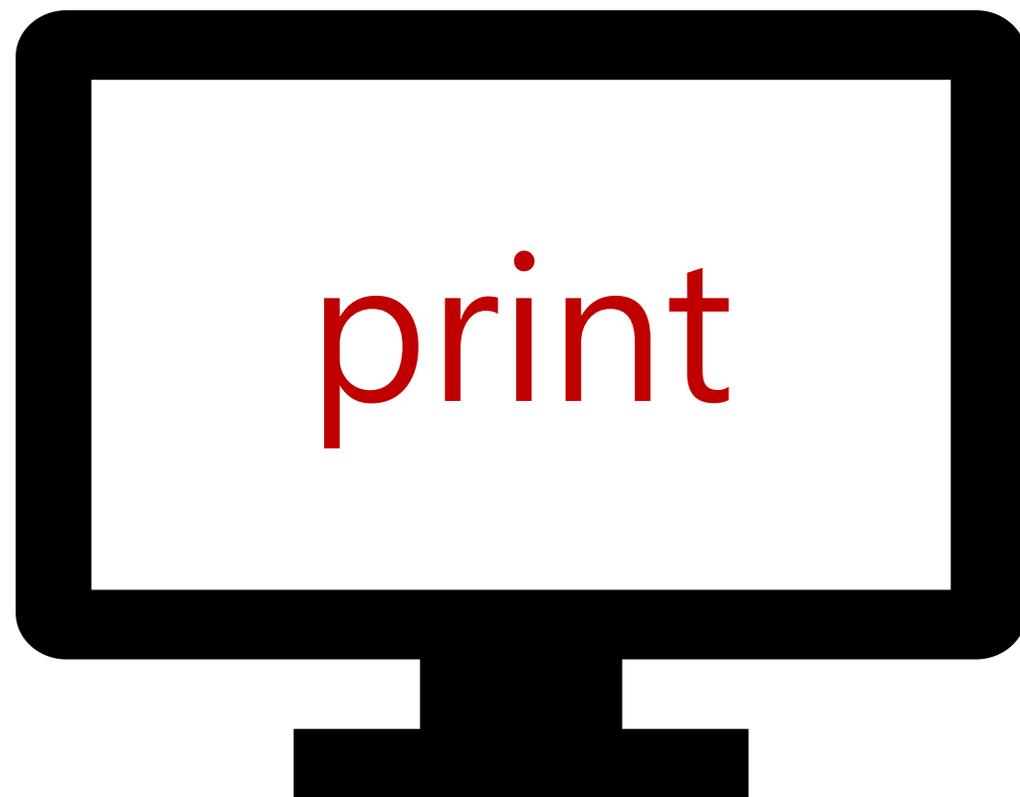


在螢幕上印出，
Hello, World!

問題



在螢幕輸出文字的指令



print指令的用法：
以輸出Hello, World!為例

```
print ("Hello, World!")
```



你好!

一些注意事項

- 紅色的字，會被印出來
 - `print("Hello, World!")`
- 如果印你好該怎麼辦？
 - `print("你好")`

電腦看不懂的字：註解

註解的寫法有二：分成單行與多行

註解是給人看的，
不是給電腦看的！

註解
就像閱讀者黏貼在書中內
頁的便條紙筆記一樣，不
會影響原書的表達，但可
以讓閱讀者更快速的了解
該書要傳達的意思。

單行註解

```
print("Hello, Python!") # 這是單行註解: 這行會印出
```

多行註解

```
"""
```

```
這是多行註解，這裡面的程式不會被執行
```

```
print( "Hello, World!" )
```

```
"""
```

```
print("Hello, Python!") # 這是單行註解: 這行會印出
```

包含註解的程式執行結果

```
print("Hello, Dice!") #這是單行註解: 這行的輸出結果會  
印出  
"""
```

這是多行註解，這裡面的程式不會被執行

```
print( "Hello, World!" )  
"""
```

```
print("Hello, Python!") # 這是單行註解: 這行會印出
```



註解有兩種:

- 多行註解: 使用三個雙引號包圍 (包在三個雙引號之間的所有文字都是註解)
- 單行註解: 使用 # 開頭 (# 後面的文字都是註解)

執行結果

```
Hello, Dice!  
Hello, Python!
```

PYTHON

延伸的概念

換行與不換行

自動換行

- 每一個 `print()` 印完都會自動換行，不需要其他的指令
- 以下範例參考

```
#自動換行程式碼範例  
print("2-3=5")  
print("2-3=5")
```

執行結果

```
2-3=5  
2-3=5
```

拒絕換行

- 可以加上參數控制：end是控制換行的參數
- end參數用法如下：
 - print後面需要加上, end= ""
- 以下程式範例參考

```
#拒絕換行程式碼範例  
print("2-3=5", end="")  
print("2-3=5")
```

執行結果=

2-3=5 2-3=5

拒絕換行，但是兩個句子中間有空格

- 可以加上參數控制：end是控制換行的參數
- end參數用法如下：
 - print後面需要加上, end= ""
 - "" 雙引號中留個空
- 以下程式範例參考

```
#拒絕換行程式碼範例  
print("2-3=5", end=" ")  
print("2-3=5")
```

執行結果

2-3=5 2-3=5

多空一行

- `\n`，表示換行
- 以下範例參考

```
#多空一行程式碼範例  
print("*\n")  
print("**\n")  
print("***\n")
```

執行結果

```
*  
**  
***
```

一次多空幾行

- 可以加上參數控制：end是控制換行的參數
- end參數用法如下：
 - `print(1, end= "\n\n")`
- 以下程式範例參考

```
#一次多空幾行程式碼範例  
print("*", end="\n\n\n")  
print("**\n")  
print("***")  
print("****")
```

執行結果

```
*  
  
  
**  
***  
****
```