

**1 useage of QProgressBar and timer**

self.pro=**QProgressBar**()

self.MyTable.**setCellWidget**(0,2,self.pro)

self.**startTimer**(100) #set a timer

self.step=0

def **timerEvent**(**self**, QTimerEvent): # when sattled timer reach

if self.step>=100:

self.**killTimer**(0)

return

self.step+=1

self.pro.setValue(self.step)

return super().timerEvent(QTimerEvent)

**2 lambda expression**

Lambda [arg1,arg2,…]]: return vcalue. For example:

Lambda a,b:a+b

Lambda L: L[0]

**3 map(operate,object list)**

This function will implement operation operate specfied to each element in object list

**4 communcation among multiple threads implementing with signal slot**

Any type of customized signal can be used in pyqt, reference:

http://blog.csdn.net/a359680405/article/details/45196207

**5 regular expression**

.\*? : match a string of arbitrarily length with greedy mode

(.\*?): matched content will be return by a touple

abc|abc: match a arbitrarily expression in the left or right of ‘|’

re module:

re.compile(r”<div.\*?>(.\*?)</div>”,re.S)

6 **for key in info\_dict.keys(): print(key)**

automatic\_captions

creator

thumbnail

description

extractor

formats

subtitles

view\_count

categories

tags

title

is\_live

end\_time

average\_rating

webpage\_url\_basename

webpage\_url

annotations

id

uploader\_id

extractor\_key

upload\_date

duration

age\_limit

start\_time

alt\_title

like\_count

license

uploader\_url

uploader

dislike\_count

None

**7 the “get” of dict is better than [‘key’]**

If there no specified key in a dict while visit the dict with dict[‘key’], it will encount index error

8 enumerate 函数用于遍历序列中的元素以及它们的下标：

>>> for i,j in enumerate(('a','b','c')):  
 print i,j

0 a  
1 b  
2 c

**Python 字典(Dictionary) get()方法**

**描述**

Python 字典(Dictionary) get() 函数返回指定键的值，如果值不在字典中返回默认值。

**语法**

get()方法语法：

dict.get(key, default=None)

**参数**

* key -- 字典中要查找的键。
* default -- 如果指定键的值不存在时，返回该默认值值。

**返回值**

返回指定键的值，如果值不在字典中返回默认值None。

**实例**

以下实例展示了 get()函数的使用方法：

[download] Destination: 《三界传说之魔界圣僧》~2016最新电影-HYo1ufMpd\_c.webm

[download] Destination: 《三界传说之魔界圣僧》~2016最新电影-HYo1ufMpd\_c.webm

[download] 8.5% of 236.43MiB at 4.95KiB/s ETA 12:25:23

[download] 8.5% of 236.43MiB at 14.86KiB/s ETA 04:09:41

[download] 8.5% of 236.43MiB at 34.51KiB/s ETA 01:47:00

[download] 8.5% of 236.43MiB at 73.59KiB/s ETA 50:10

[download] 8.5% of 236.43MiB at 77.17KiB/s ETA 47:50

[download] 8.5% of 236.43MiB at 104.60KiB/s ETA 35:17 ----->

[download] 8.5% of 236.43MiB at 126.26KiB/s ETA 29:13

[download] 8.6% of 236.43MiB at 180.58KiB/s ETA 20:25 ----->

请按任意键继续. . .

Message=[download] 8.6% of 236.43MiB at 180.58KiB/s ETA 20:25

row=re.findall(re.compile(r".\*?([0-9]+\.?[0-9]\*)%.\*of(.\*)at(.\*)ETA(.\*)",re.S),message)

**10**  **pyinstaller -F -w E:\Myprog\Youtube\_get\Youtube\_get\Youtube\_get.py**

**When data files, .dll and .so files is needed to package to a exe file, follow command should be performed for generating a .spec file , in which some necessary command can be appended.**

**pyi-makespec E:\Myprog\Youtube\_get-pyCharm\Youtube\_get\Youtube\_get.py**

**11 QPushButton{**

**background-color: #0f0 ;**

**height:30px;**

**border-style: outset; #要不要边框？**

**border-width: 2px;**

**border-radius: 10px;**

**border-color: beige;**

**font: bold 14px;**

**min-width: 10em;**

**padding: 6px;**

**}**

self.setStyleSheet**( '''QPushButton{  
height:30px;  
border-style: outset;  
border-width: 2px;  
border-radius: 15px;  
border-color: beige;  
font: bold 14px;  
min-width: 30px;  
padding: 0px;}''')**

**12、 Bundle about pyinstaller**

**(1)我用upx加壳；**

**(2) pyinstaller -w \*.pyw**

**Transform a png file to ico file with png2ico.exe**

**png2ico Icon.ico Icon.png**

**bundle py script to executable file with windows icon use follow command, Icon file must be with standard size instead some errors will be occurred**

**pyinstaller -F -w -i E:\Myprog\Youtube\_get-pyCharm\Youtube\_get\Images\Icon256x256.ico E:\Myprog\Youtube\_get-pyCharm\Youtube\_get\Youtube\_get.py**

**13、 method of bundling a python project with pictures or other data file to a single file :**

**firstlly :**

**create a .qrc file :**

**<RCC>  
<qresource prefix="/">  
 <file>**Images/1.png**</file>  
 ………….  
 <file>**Images/tube\_logo.png**</file>  
</qresource>  
</RCC>**

**Perform following command consponding python file will be generated:**

**pyrcc5 my\_resourse.qrc -o my\_resourse.qrc.py**

**then import the .py file in where you need a image**

**git init**

**git commit –m ‘gjj’**

**git remote add origin** [**https://github.com/Jinjiego/youtube-get.git**](https://github.com/Jinjiego/youtube-get.git)

**git push origin master**

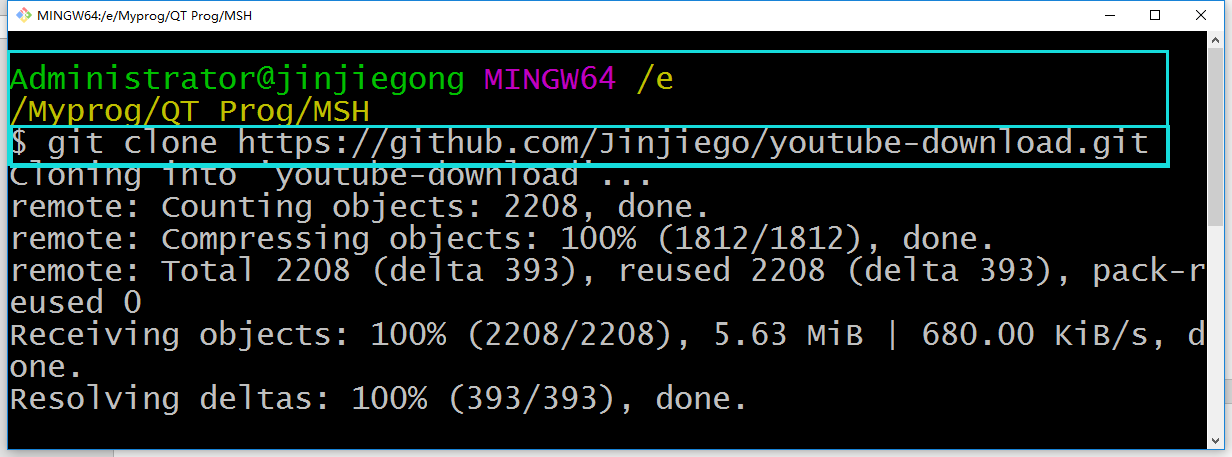
**14、 git的基本操作  
配置用户信息:  git config –global C:\Users\ADMINI~1\AppData\Local\Temp\%W@GJ$ACOF(TYDYECOKVDYB.pnguser.name  “southernriver”  
   git config –global user.email southernriver@163.com  
初始化： git  init  
跟踪：     git add index.html;  
把跟踪好的文件提交到版本库： git commit –m “add index.html first”;  
查看提交记录： git log  
创建分支：git branch testing   
切换到新建的分支：git checkout testing    
合并分支： git merge branchname;  
   注意：提交到GitHub的流程：  
（1） 在github上新建一个仓库  
（2） git clone C:\Users\ADMINI~1\AppData\Local\Temp\%W@GJ$ACOF(TYDYECOKVDYB.pnghttps://github.com/southernriver/test1.git  
（3） git add main.c  
（4） git commit –m “first commit”  
（5） git remote add origin C:\Users\ADMINI~1\AppData\Local\Temp\%W@GJ$ACOF(TYDYECOKVDYB.pnghttps://github.com/southernriver/test1.git  
（6） git push -f origin master  
（7） 输入用户名和密码**

**15、if you want to download a project from a github URL into local folder:**

**(1) right click :**



(2) type command here directly , project you want get will be stored here.



**16、If some local file be modified and you want to commit the update to github :**

**1 git add –A //this command will add modifing to local** repository

2 git commit –m ‘gjj’

3 git push –f origin master

**17、pyinstaller –F –w E:\Myprog\MshHandler\MshHandler\** **MshHandler.py**

**Pyinstaller –F –w –i E:\Myprog\youtube-download\Images1\Icon256x256.ico –p C:\Windows\System32 E:\Myprog\youtube-download\Youtube\_get.py**