

Forum:[Hack Forums](#)

Thread: [Downloading error with urllib2](#)

Komodo - April 17, 2017, 6:20 pm

So I am currently learning python as I skipped it.

Trying a "simple" code where an exe is downloaded and ran(it can be and extension doesn't have to be exe.

```
Code import urllib2
u = urllib2.urlopen('Url im using')
localFile = open('Exe im using', 'wb')
localFile.write(u.read())
localFile.close()
```

This I was trying with Python 3, and would get a this process is being used by pythonw error message after I tried to open the program it downloaded manually.

```
Code import urllib.request as urllib2
u = urllib2.urlopen('Url im using')
localFile = open('Exe im using', 'wb')
localFile.write(u.read())
localFile.close()
```

This I was trying with a previous version of Python, and would get:
"The version of the file is not compatible with the version of Windows you're running. Check your computer's system information to see whether you need an x86 (32-bit) or x64 (64-bit) version of the program, and then contact the software publisher"
This compatibility shouldn't be an issue.

I tried both scripts with 'w' and 'wb'

Are the codes I'm trying wrong, or is there something I am doing wrong?

Any help is appreciated.

Folks - April 17, 2017, 9:56 pm

Is there a specific reason for suing urllib2?

PHP Code

```
<?php
```

```
import requests
```

```
r = requests.get('https://www.bleepstatic.com/tutorials/xp-slipstream/extract.jpg')
output = r.content
```

```
with open('test.png', 'wb') as f:
    f.write(output)
```

this works for me using requests.

EDIT: Your above code works for me too though anyway... Can you give a link to something that doesn't work for you?

Komodo - April 17, 2017, 10:19 pm

(Apr 17, 2017 09:56 PM)Folks Wrote: [\[CITE\]](#) Is there a specific reason for suing urllib2?

PHP Code

```
<?php
```

```
import requests
```

```
r = requests.get('https://www.bleepstatic.com/tutorials/xp-slipstream/extract.jpg')
```

```
output = r.content
```

```
with open('test.png', 'wb') as f:
```

```
    f.write(output)
```

this works for me using requests.

EDIT: Your above code works for me too though anyway... Can you give a link to something that doesn't work for you? Pretty new to it, thought it was the way to go.

I tried it with anything ranging from direct exes, mozilla firefox, etc. I got compatibility or corrupted files from each of them.

Just for future reference, why is this code better than the other. Just curious.

Folks - April 17, 2017, 10:31 pm

(Apr 17, 2017 10:19 PM)Komodo Wrote: [\[CITE\]](#) Pretty new to it, thought it was the way to go.

I tried it with anything ranging from direct exes, mozilla firefox, etc. I got compatibility or corrupted files from each of them.

Just for future reference, why is this code better than the other. Just curious.

It's more of a preference thing I think but from what I have read more people like to use requests than urllib.

I am currently on a Mac but this didn't corrupt my file and it worked for the images when on my PC. Are you sure you are waiting for the download to finish?

PHP Code

```
<?php
```

```
import urllib.request as urllib2
```

```
u = urllib2.urlopen(
```

```
'https://download-installer.cdn.mozilla.net/pub/firefox/releases/52.0.2/mac/en-US/Firefox%2052.0.2.dmg')
```

```
localFile = open('test.dmg', 'wb')
```

```
localFile.write(u.read())
```

```
localFile.close()
```

captainspalding - April 17, 2017, 10:43 pm

(Apr 17, 2017 10:31 PM)Folks Wrote: [\[CITE\]](#) It's more of a preference thing I think but from what I have read more people like to use requests than urllib.

I am currently on a Mac but this didn't corrupt my file and it worked for the images when on my PC. Are you sure you are waiting for the download to finish?

PHP Code

```
<?php
import urllib.request as urllib2

u = urllib2.urlopen(
'https://download-installer.cdn.mozilla.net/pub/firefox/releases/52.0.2/mac/en-US/Firefox%2052.0.2.dmg')
localFile = open('test.dmg', 'wb')
localFile.write(u.read())
localFile.close()
```

To add to the logic behind using 'with open' to open files, when you 'with open', there is an implicit close. This means less lines of code to open and close the file, no chance of forgetting to close a file when done working with it, and in general is more pythonic.

[Style Guide For Python Code](#)

SIM-1866423 - April 18, 2017, 3:20 am

What about asynchoronus things? Python3.5 +

PHP Code

```
<?php
import aiohttp
import asyncio
import async_timeout
import os
```

```
async def download_coroutine(session, url):
    with async_timeout.timeout(10):
        async with session.get(url) as response:
            filename = os.path.basename(url)
            with open(filename, 'wb') as f_handle:
                while True:
                    chunk = await response.content.read(1024)
                    if not chunk:
                        break
                    f_handle.write(chunk)
            return await response.release()
```

```
async def main(loop):
    urls = [
```

```
"http://www.blog.pythonlibrary.org/wp-content/uploads/2016/04/Python201_cover20160330_sm-237x300.jpg",
```

```
"http://www.blog.pythonlibrary.org/wp-content/uploads/2014/02/mousecovertitlejpg_sm2-237x300.jpg",
```

```
]
```

```
async with aiohttp.ClientSession(loop=loop) as session:
    for url in urls:
        await download_coroutine(session, url)
```

```
if __name__ == '__main__':
    loop = asyncio.get_event_loop()
    loop.run_until_complete(main(loop))
```

Let you download multiple file at once, but can be require semaphore(delimit) if you want download too much file.

Source : <https://www.blog.pythonlibrary.org/2016/11/09/an-intro-to-aiohttp/>

ps : oops wrong example use it with aiofiles for asynchronous writing.

hunter24957 - April 18, 2017, 6:21 am

(Apr 17, 2017 10:43 PM)captainspalding Wrote: [\[CITE\]](#) To add to the logic behind using 'with open' to open files, when you 'with open', there is an implicit close. This means less lines of code to open and close the file, no chance of forgetting to close a file when done working with it, and in general is more pythonic.

[Style Guide For Python Code](#)

These are called context managers if you would like to search for them, as 'with open' is not explicit enough.

They are used like so:

PHP Code

```
<?php
```

with open('filename.exe', 'wb') as f:

```
    f.write(r.content)
```

Folks - April 18, 2017, 10:46 am

(Apr 17, 2017 10:43 PM)captainspalding Wrote: [\[CITE\]](#) To add to the logic behind using 'with open' to open files, when you 'with open', there is an implicit close. This means less lines of code to open and close the file, no chance of forgetting to close a file when done working with it, and in general is more pythonic.

[Style Guide For Python Code](#)

Was this in response to me or OP? I use context managers myself but was pointing out that OP's code worked fine for me.

Komodo - April 18, 2017, 11:37 am

(Apr 18, 2017 10:46 AM)Folks Wrote: [\[CITE\]](#) Was this in response to me or OP? I use context managers myself but was pointing out that OP's code worked fine for me.Can you give an example of a program you downloaded with it, I tried everything I could get my hands on.

Folks - April 18, 2017, 12:30 pm

(Apr 18, 2017 11:37 AM)Komodo Wrote: [\[CITE\]](#) Can you give an example of a program you downloaded with it, I tried everything I could get my hands on.

Firefox worked for me, how are you running the file? cmd, idle, etc?

EDIT: I am assuming windows since you are using exe's but what OS are you using?

PHP Code

```
<?php
import urllib.request as urllib2

u = urllib2.urlopen(
'https://stubdownloader.cdn.mozilla.net/builds/firefox-stub/en-US/win/7b231be55962f90d39ce7b354
65dec88b4142e539a72277f3347391326002e1d/Firefox%20Setup%20Stub%2052.0.2.exe')
localFile = open('test.exe', 'wb')
localFile.write(u.read())
localFile.close()
```

Komodo - April 18, 2017, 1:13 pm

(Apr 18, 2017 12:30 PM)Folks Wrote: [\[CITE\]](#) Firefox worked for me, how are you running the file?
cmd, idle, etc?

EDIT: I am assuming windows since you are using exe's but what OS are you using?

PHP Code

```
<?php
import urllib.request as urllib2

u = urllib2.urlopen(
'https://stubdownloader.cdn.mozilla.net/builds/firefox-stub/en-US/win/7b231be55962f90d39ce7b354
65dec88b4142e539a72277f3347391326002e1d/Firefox%20Setup%20Stub%2052.0.2.exe')
localFile = open('test.exe', 'wb')
localFile.write(u.read())
localFile.close()
I was using it with idle.
Tried on Windows 10 and 7
```

How would I go about running it outside of idle, would I just save it as a pv and run it. Still fairly new to this.

Folks - April 18, 2017, 1:16 pm

(Apr 18, 2017 01:13 PM)Komodo Wrote: [\[CITE\]](#) I was using it with idle.
Tried on Windows 10 and 7

How would I go about running it outside of idle, would I just save it as a pv and run it. Still fairly new to this.

Does the program exit after you run it or does it hang? You don't need to run it outside of idle but I personally use visual studio code and a command prompt.

But yes you would just save it as .py and run it if you didn't want to use a cmd prompt.

Do you have teamviewer or something? I could help you before I go to my next class.

Komodo - April 18, 2017, 2:22 pm

(Apr 18, 2017 01:16 PM)Folks Wrote: [\[CITE\]](#) Does the program exit after you run it or does it hang?
You don't need to run it outside of idle but I personally use visual studio code and a command

prompt.

But yes you would just save it as .py and run it if you didn't want to use a cmd prompt.

Do you have teamviewer or something? I could help you before I go to my next class. I was out, just kinda trying to figure out why it won't run for me. It will download (I have tried mozilla in the past) and save on my desktop as an exe without an icon then it will be corrupt.

If I either run it while still in idle or double click it, it will open then close a cmd then an icon will appear on my desktop (the exe) then nothing happens.

captainspalding - April 18, 2017, 3:04 pm

(Apr 18, 2017 10:46 AM) Folks Wrote: [\[CITE\]](#) Was this in response to me or OP? I use context managers myself but was pointing out that OP's code worked fine for me. Just adding to your answer to OP's question why your code was 'better'.