

Python Speedtest Tutorial

Contents

Python Speedtest Tutorial	1
Speedtest-cli	1
Tutorial 1: Research	1
Tutorial 2: Speedtest CLI Beta	2
Tutorial 3: Speedtest CLI Simple	3
Tutorial 4: Speedtest CLI OOP	5
Tutorial 5: Speedtest GUI Tkinter	8
Assignment Submission	12

Time required: 90 minutes

Speedtest-cli

Speedtest.net that tests the speed and performance of your internet connection. Speedtest is available in many forms, web site, apps, and a Python library located at <https://pypi.org/project/speedtest-cli>

The **speedtest-cli** Python library provides a command line interface (CLI) for testing internet bandwidth using speedtest.net.

The following tutorials show a development process from research to testing to final product.

Tutorial 1: Research

We want to be able to measure internet bandwidth from a Python program.

1. Google **python speedtest**
2. The first result usually is a CLI application from Speedtest.net. This is the type of program we are looking for. <https://www.speedtest.net/apps/cli>
3. The next result is usually a Python library, <https://pypi.org/project/speedtest-cli/> A step in the right direction. We have found a Python library that wraps speedtest-cli program.

4. Let's look for some tutorials to get a better idea of how this library works. We will search for Python speedtest-cli tutorials or python internet speedtest.
 - a. <https://www.geeksforgeeks.org/test-internet-speed-using-python/>
 - b. <https://yourblogcoach.com/how-to-test-internet-speed-using-python/>
 - c. <https://pyshark.com/test-internet-speed-using-python/>
 - d. <https://www.codespeedy.com/test-internet-speed-using-python/>

Tutorial 2: Speedtest CLI Beta

This is the first version after researching tutorials and speed-cli documentation. This version is to make sure the basic methods are working.

Create a Python program named **speedtest_cli_1.py**

```
1  """
2      Name: speedtest_cli_1.py
3      Author: William A Loring
4      Created: 12/8/21
5      speedtest-cli is a Python module
6      that uses speedtest.net to test internet bandwidth
7      https://github.com/sivel/speedtest-cli
8      https://pypi.org/project/speedtest-cli/
9  """
10
11 # pip install speedtest-cli
12 from speedtest import Speedtest
13
14 print(" Starting SpeedTest . . . please be patient . . .")
15
16 # Create speedtest object
17 speed = Speedtest(secure=True)
18
19 # Get download bandwidth, returns bits per second
20 download_result = speed.download()
21
22 # Get upload bandwidth, returns bits per second
23 upload_result = speed.upload()
24
25 # Get ping results/latency, return ms
26 ping_result = speed.results.ping
27
28 # Display results of speedtest
29 print(f"\n Download Bandwidth: {download_result}")
30 print(f"      Upload Bandwidth: {upload_result}")
31 print(f"      Latency (ping): {ping_result}")
```

There isn't any feedback on this program until the end. Be patient, it can take almost 30 seconds to finish the test.

Example run:

```
Download Bandwidth: 627602969.6985183  
Upload Bandwidth: 742244957.0179085  
Latency (ping): 16.213
```

Tutorial 3: Speedtest CLI Simple

Copy the last file and rename it as **speedtest_cli_2.py**

Keeping previous versions of programs is helpful if the changes you are making break the program.

This version added conversion to megabits per second and some display additions.

```

1  """
2      Name: speedtest_cli_2.py
3      Author: William A Loring
4      Created: 12/8/21
5      speedtest-cli is a Python module
6      that uses speedtest.net to test internet bandwidth
7      https://github.com/sivel/speedtest-cli
8      https://pypi.org/project/speedtest-cli/
9  """
10
11 # speedtest-cli return bandwidth in bits per second
12 # A megabit is 1 million bits
13 # Bandwidth is typically measured in megabits per second (mbps)
14
15 # pip install speedtest-cli
16 from speedtest import Speedtest
17
18 # Create speedtest object
19 speed = Speedtest(secure=True)
20
21 #----- GET SERVER INFO -----#
22 print(" Start SpeedTest . . .")
23
24 # Return the nearest test server and location in dictionary format
25 # A ping test determines the server with the lowest latency
26 server = speed.get_best_server()
27
28 # Get information about nearest server from returned server dictionary
29 sponsor = f'{server.get("sponsor")}'
30 location = f'{server.get("name")}'
31 country_code = f'{server.get("cc")}'
32
33 #----- GET DOWNLOAD BANDWIDTH -----#
34 print(" Get Download Bandwidth . . .")
35
36 # Get download bandwidth, returns bits per second
37 download_result = speed.download()
38 # Convert from bits per second to megabits per second
39 # There are 1,000,000 bits per second in 1 megabit per second
40 download_result = download_result / 1000 / 1000

```

```

42 #----- GET UPLOAD BANDWIDTH-----#
43 print(" Get Upload Bandwidth . . .")
44
45 # Get upload bandwidth, returns bits per second
46 upload_result = speed.upload()
47 # Convert from bits per second to megabits per second
48 # There are 1,000,000 bits per second in 1 megabit per second
49 upload_result = upload_result / 1000 / 1000
50
51 #----- GET PING LATENCY -----#
52 print(" Get Ping Latency . . .")
53
54 # Get ping results/latency, return ms
55 ping_result = speed.results.ping
56
57 #----- DISPLAY SPEEDTEST RESULTS -----#
58 print(f"\n {sponsor} - {location}, {country_code}")
59 print(f" Download Bandwidth: {download_result:.2f} Mbps")
60 print(f" Upload Bandwidth: {upload_result:.2f} Mbps")
61 print(f" Latency (ping): {ping_result} ms")
62
63 input("\n Press Enter to exit")

```

Example run:

```

Start SpeedTest . . .
Get Download Bandwidth . . .
Get Upload Bandwidth . . .
Get Ping Latency . . .

Viaero Wireless - Fort Morgan, CO, US
Download Bandwidth: 565.36 Mbps
Upload Bandwidth: 258.20 Mbps
Latency (ping): 11.313 ms

```

Tutorial 4: Speedtest CLI OOP

This version converts the program to OOP. The code is pretty much the same, just in OOP format.

Save the previous program as **speedtest_cli_3.py**

```

1  """
2      Name: speedtest_cli_3_oop.py
3      Author: William A Loring
4      Created: 12/8/21
5      speedtest-cli is a Python module
6      that uses speedtest.net to test internet bandwidth
7      https://github.com/sivel/speedtest-cli
8      https://pypi.org/project/speedtest-cli/
9  """
10
11 # speedtest-cli return bandwidth in bits per second
12 # A megabit is 1 million bits
13 # Bandwidth is typically measured in megabits per second (mbps)
14
15 # pip install speedtest-cli
16 from speedtest import Speedtest
17 import utils
18
19
20 class SpeedtestCLI:
21     def __init__(self):
22         # Create speedtest object
23         self._speedtest = Speedtest(secure=True)
24
25 #----- GET SERVERS -----#
26     def get_servers(self):
27         # Return the nearest test server and location in dictionary format
28         # A ping test determines the server with the lowest latency
29         server = self._speedtest.get_best_server()
30
31         # Get information about nearest server from returned server dictionary
32         self._sponsor = f'{server.get("sponsor")}'
33         self._location = f'{server.get("name")}'
34         self._country_code = f'{server.get("cc")}'
35
36 #----- GET DOWNLOAD BANDWIDTH -----#
37     def get_download_bandwidth(self):
38         print(" Get Download Bandwidth . . .")
39         # Get download bandwidth, returns bits per second
40         download_result = self._speedtest.download()
41         # Convert from bits per second to megabits per second
42         # There are 1,000,000 bits per second in 1 megabit per second
43         self._download_result = download_result / 1000 / 1000
44
45 #----- GET UPLOAD BANDWIDTH -----#
46     def get_upload_bandwidth(self):
47         print(" Get Upload Bandwidth . . .")
48         # Get upload bandwidth, returns bits per second
49         upload_result = self._speedtest.upload()
50         # Convert from bits per second to megabits per second
51         # There are 1,000,000 bits per second in 1 megabit per second
52         self._upload_result = upload_result / 1000 / 1000
53

```

```

54 #----- GET PING LATENCY -----#
55 def get_ping_latency(self):
56     print(" Get Ping Latency . . .")
57     # Get ping results/latency, return ms
58     self._ping_result = self._speedtest.results.ping
59
60 #----- DISPLAY SPEEDTEST RESULTS -----#
61 def display_results(self):
62     # Display results of speedtest
63     print(f"\n {self._sponsor} - {self._location}, {self._country_code}")
64     print(f"\n Download Bandwidth: {self._download_result:.2f} Mbps")
65     print(f"    Upload Bandwidth: {self._upload_result:.2f} Mbps")
66     print(f"    Latency (ping): {self._ping_result} ms")
67
68     input("\n Press [Enter] to exit")
69
70
71 def main():
72     print(utils.title("Internet Speed Test"))
73     # Create program object
74     speedtest = SpeedtestCLI()
75     # Call program object methods
76     speedtest.get_servers()
77     speedtest.get_download_bandwidth()
78     speedtest.get_upload_bandwidth()
79     speedtest.get_ping_latency()
80     speedtest.display_results()
81
82 # Start the program
83 main()

```

Example run:

```

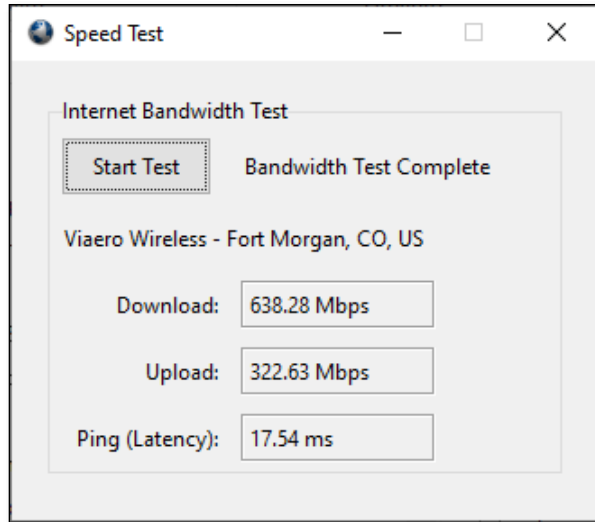
+-----+
| Internet Speed Test |
+-----+
Get Download Bandwidth . . .
Get Upload Bandwidth . . .
Get Ping Latency . . .

Viaero Wireless - Fort Morgan, CO, US

Download Bandwidth: 601.44 Mbps
Upload Bandwidth: 290.72 Mbps
Latency (ping): 11.716 ms

```

Tutorial 5: Speedtest GUI Tkinter



Sketch out the GUI before you start to build it. The example shown is built inside a single frame using the grid layout manager with two columns and multiple rows. It would be good practice to create your own UI.

All the methods in the OOP CLI program will be modified to fit the GUI. All the logic will be the same. The changes will be to fit a Tkinter GUI program.

You will want to download an **internet.ico** file of some sort.

Save the last tutorial as **speedtest_gui.py**


```

1  """
2      Name: speedtest_gui.py
3      Author: William A Loring
4      Created: 12/05/21
5      Purpose: Test internet upload and download speed with python
6      using Speedtest.net
7      https://github.com/sivel/speedtest-cli
8      https://pypi.org/project/speedtest-cli/
9  """
10 from tkinter import *
11 from tkinter.ttk import *
12 # pip install speedtest-cli
13 from speedtest import Speedtest
14
15
16 class SpeedTestGui:
17     # Define the initialize method
18     def __init__(self):
19         # Create speedtest object
20         self._speedtest = Speedtest(secure=True)
21
22         self.root = Tk()
23         self.root.title("Internet Speed Test")
24         self.root.geometry("325x255")
25         self.root.iconbitmap("speed.ico")
26         self.root.resizable(False, False)
27
28         # Call method to create all the widgets
29         self.create_widgets()
30
31         # Start main program loop
32         mainloop()
33
34 # ----- RUN SPEEDTEST -----#
35 def run_speedtest(self, *args):
36     # Clear all labels
37     self.lbl_server.config(text="")
38     self.lbl_download.config(text="")
39     self.lbl_upload.config(text="")
40     self.lbl_ping.config(text="")
41
42     # Call methods to run speed test
43     self.get_best_server()
44     self.get_download_bandwidth()
45     self.get_upload_bandwidth()
46     self.get_ping_latency()

```

```

48 # ----- GET DOWNLOAD BANDWIDTH -----#
49 def get_download_bandwidth(self):
50     """Get download bandwidth from test server."""
51     self.lbl_status.configure(text="Get Download Bandwidth . . .")
52     # Update the frame to show the label changes
53     self.main_frame.update()
54     # Get download bandwidth, returns bits per second
55     download_result = self._speedtest.download()
56     # Convert from bits per second to megabits per second
57     self._download_result = download_result / 1000 / 1000
58
59     self.lbl_download.configure(
60         text=f" {self._download_result:.2f} Mbps")
61     # Update the frame to show the label changes
62     self.main_frame.update()
63
64 # ----- GET UPLOAD BANDWIDTH -----#
65 def get_upload_bandwidth(self):
66     """Get upload bandwidth from test server."""
67     # Get and print upload speed
68     self.lbl_status.configure(text="Get Upload Bandwidth . . .")
69     self.main_frame.update()
70
71     # Get upload bandwidth, returns bits per second
72     upload_result = self._speedtest.upload()
73     # Convert from bits per second to megabits per second
74     self._upload_result = upload_result / 1000 / 1000
75
76     self.lbl_upload.configure(
77         text=f" {self._upload_result:.2f} Mbps")
78     # Update the frame to show the label changes
79     self.main_frame.update()
80
81 # ----- GET PING LATENCY -----#
82 def get_ping_latency(self):
83     """Get ping latency from test server."""
84     # Get ping results/latency, return ms
85     self._ping_result = self._speedtest.results.ping
86
87     self.lbl_ping.configure(text=f" {self._ping_result:.2f} ms")
88     self.lbl_status.configure(text="Speed Test Complete")
89     # Update the frame to show the label changes
90     self.main_frame.update()

```

```

192 # ----- GET BEST SERVER -----#
193 def get_best_server(self):
194     # Returns the nearest server
195     # Return the nearest test server and location in dictionary format
196     best_server = self._speedtest.get_best_server()
197     sponsor = best_server.get("sponsor")
198     name = best_server.get("name")
199     cc = best_server.get("cc")
200     self.lbl_server.configure(text=f"{sponsor} - {name}, {cc}")
201
202     # Update the frame to show the label changes
203     self.main_frame.update()
204
205 # ----- CREATE WIDGETS -----#
206 def create_widgets(self):
207     self.main_frame = LabelFrame(
208         self.root,
209         text="Internet Speed Test",
210         relief=GROOVE)
211
212     # Fill the frame to the width of the window
213     self.main_frame.pack(fill=X)
214     # Keep the frame size regardless of the widget sizes
215     self.main_frame.pack_propagate(False)
216
217     self.lbl_status = Label(
218         self.main_frame,
219     )
220     self.lbl_server = Label(
221         self.main_frame,
222     )
223
224     self.lbl_download_label = Label(
225         self.main_frame,
226         anchor=E,
227         text="Download:",
228         width=13
229     )
230     self.lbl_upload_label = Label(
231         self.main_frame,
232         anchor=E,
233         text="Upload:",
234         width=13
235     )
236     self.lbl_ping_label = Label(
237         self.main_frame,
238         anchor=E,
239         text="Ping (Latency):",
240         width=13
241     )

```

```

143         self.lbl_download = Label(
144             self.main_frame,
145             width=16,
146             relief=GROOVE
147         )
148         self.lbl_upload = Label(
149             self.main_frame,
150             width=16,
151             relief=GROOVE
152         )
153         self.lbl_ping = Label(
154             self.main_frame,
155             width=16,
156             relief=GROOVE
157         )
158         self.button = Button(
159             self.main_frame,
160             text="Start Test",
161             command=self.run_speedtest
162         )
163
164         self.button.grid(row=0, column=0, sticky=W)
165         self.lbl_status.grid(row=0, column=1, sticky=W)
166         self.lbl_server.grid(row=1, column=0, sticky=W, columnspan=2)
167
168         self.lbl_download_label.grid(row=2, column=0, sticky=EW)
169         self.lbl_upload_label.grid(row=3, column=0, sticky=EW)
170         self.lbl_ping_label.grid(row=4, column=0)
171
172         self.lbl_download.grid(row=2, column=1, sticky=W)
173         self.lbl_upload.grid(row=3, column=1, sticky=W)
174         self.lbl_ping.grid(row=4, column=1, sticky=W)
175
176         # Set padding between frame and window
177         self.main_frame.pack_configure(padx=20, pady=20)
178         # Set padding for all widgets
179         for child in self.main_frame.winfo_children():
180             child.grid_configure(padx=5, pady=5, ipadx=4, ipady=4)
181         # The enter key will activate the calculate method
182         self.root.bind("<Return>", self.run_speedtest)
183         self.root.bind("<KP_Enter>", self.run_speedtest)
184
185
186         # Create object from the program class to run the program
187         speed_test = SpeedTestGui()

```

Assignment Submission

1. Attach all program files.

2. Attach a screenshot of each successful program run.
3. Submit the assignment in Blackboard.