Write code in Python 3.6 ▼

```
1
```

Help improve this tool by completing a **short user survey**
Keep this tool free by making a **small donation** (PayPal, Patreon, credit/debit card)

Visualize Execution Live Programming Mode

Advanced instructions: setting breakpoints | hiding variables | live programming

hide exited frames [default] ▼ inline primitives but don't nest objects [default] ▼
draw pointers as arrows [default] ▼

Create test cases

Show example code and courses

Generate permanent link

Generate shortened link

Click above to create a permanent link to your visualization (video demo). To report bugs, paste the link along with an error description in an email to philip@pgbovine.net

Python Tutor (code on GitHub) supports seven languages (despite its name!):

1. Python 2.7 and 3.6 with these limited module imports: __future__, abc, array, bisect, calendar, cmath, collections, copy, datetime, decimal, doctest, fractions, functools, hashlib, heapq, io, itertools, json, locale, math, operator, pickle, pprint, random, re, string, time, types, unittest, StringIO (Python 2), typing (Python 3). (There is also an experimental version of Python 3.6 with Anaconda, which lets you import many more modules.) Backend source code.

2. Java using Oracle's Java 8. The original Java visualizer was created by David Pritchard and Will Gwozdz. It supports StdIn, StdOut, most other stdlib libraries, Stack, Queue, and ST. (To access Java's built-in Stack/Queue classes, write `import java.util.Stack;` — note, `import java.util.*;` won't work.) Backend source code.

3. JavaScript ES6 running in Node.js v6.0.0. Backend source code.

4. TypeScript 1.4.1 running in Node.js v6.0.0. Backend source code.

5. Ruby 2 using MRI 2.2.2. Backend source code.

6. C using gcc 4.8, C11, and Valgrind Memcheck. Backend source code.

7. C++ using gcc 4.8, C++11, and Valgrind Memcheck. Backend source code.

Privacy Policy: By using Python Tutor, your visualized code, options, user interactions, text chats, and IP address are logged on our server and may be analyzed for research purposes. Nearly all web services collect this basic information from users in their server logs. However, Python Tutor does not collect any personally identifiable information from its users. It uses Google Analytics for website analytics.

Terms of Service: The Python Tutor service is provided for free on an as-is basis. Use this service at your own risk. Do not use it to share confidential information. The developers of Python Tutor are not responsible for the chat messages or behaviors of any of the users on this website. We are also not responsible for any damages caused by using this website. Finally, it is your responsibility to follow appropriate academic integrity standards.