

LAB 1 SUPPLEMENTARY:

BASIC DEBUGGING WITH PYTHON

General Tips on Minimizing Errors

Debugging with 'print'

Debugging with PDB

Using Google/Stack Overflow



General Tips on Minimizing Errors

Do not panic when you get errors

Outline your code structure ahead of time

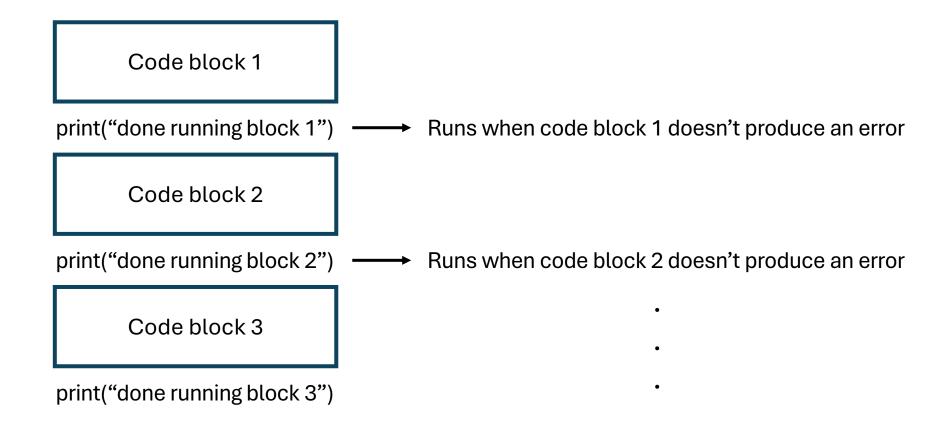
Keep your code organized

Test your code often

More tips on avoiding errors by Berkeley online textbook https://pythonnumericalmethods.berkeley.edu/notebooks/chapter10.00-Errors-Practices-Debugging.html

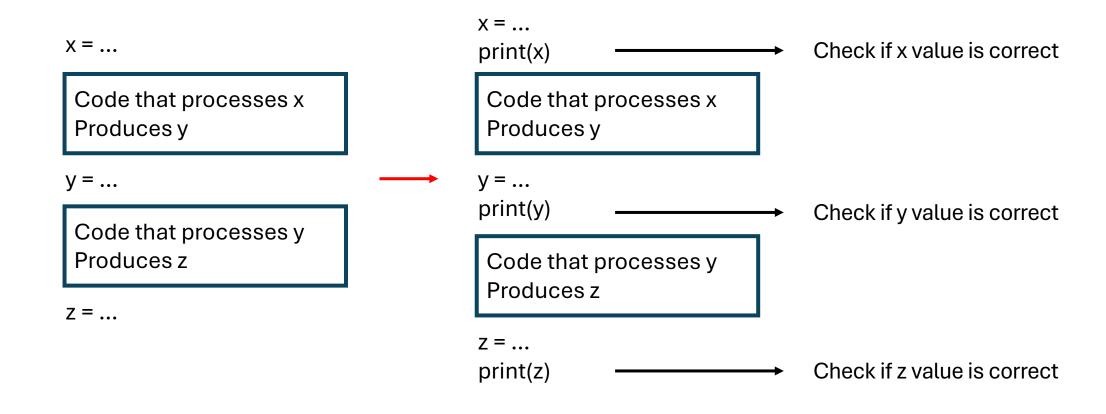


Basic Debugging with 'print'





Basic Debugging with 'print'



Original code

Debugging each step with print()



Python Debugger

What is Python Debugger (PDB)?

- Debugging tool built in Python
- No installation required
- Provides Interactive environment
- Widely used for systemic debugging





PDB Basics

```
def divide(divisor):
                                  Upon execution of the code, activates Python
   breakpoint() -
                                  debugger console at this line
   val = 1/divisor
   return val
divide(0)
> <ipython-input-24-03e5b3dd8265>(5)divide()
           breakpoint()
      3
         val = 1/divisor
           return val
                                                         Python debugger console
ipdb>
```





divide(0)

> <ipython-input-24-03e5b3dd8265>(5)divide()

PDB Useful Commands

h: help

```
w: where
          breakpoint()
                                                                            n: next
          val = 1/divisor
                                                                            c: continue
     7
          return val
                                                                            p: print
ipdb> h
Documented commands (type help <topic>):
                                                                            l: list
      commands
               enable
                                                         until
                                                                            q: quit
      condition exit
                        longlist
                                 psource skip hidden
                                         skip predicates
alias cont
      context
                help
                                 auit
                                          source
                                                         whatis
args
                         next
      continue
               ignore
                                          step
                                                         where
                interact
                                 restart tbreak
break d
                        pdef
      debug
                        pdoc
bt
                                 return
      disable
                                 retval
                                         unalias
                jump
                        pfile
      display
                        pinfo
                                         undisplay
                                 run
clear down
                        pinfo2
                                          unt
Miscellaneous help topics:
exec pdb
                                                           Provide documentation for the command
ipdb> h l
Print lines of code from the current stack frame
ipdb>
```



```
divide(2)
                                                                         h: help
> <ipython-input-1-f4a9c3842530>(7)divide()
                                                                         w: where
          val = 1/divisor
          breakpoint()
                                                                         n: next
                                                                         c: continue
---> 7
          return val
                                                                         p: print
   [... skipping 27 hidden frame(s)]
                                                                         l: list
  <ipython-input-2-cfdb0f794c84>(1)<module>()
  --> 1 divide(2)
                                                                         q: quit
 <ipython-input-1-f4a9c3842530>(7)divide()
          val = 1/divisor
          breakpoint()
          return val
                                                     Print a stack trace (i.e. the order of code being
                                                     executed), with the most recent frame at the
ipdb>
                                                     bottom.
```



```
h: help
divide(2)
                                                                 w: where
> <ipython-input-1-03e5b3dd8265>(5)divide()
           breakpoint()
                                                                 n: next
                                                                 c: continue
         val = 1/divisor
                                                                 p: print
           return val
                                                                 l: list
> <ipython-input-1-03e5b3dd8265>(7)divide()
                                                                 q: quit
           breakpoint()
           val = 1/divisor
           return val
                                                         Executes the next line of code
ipdb>
```

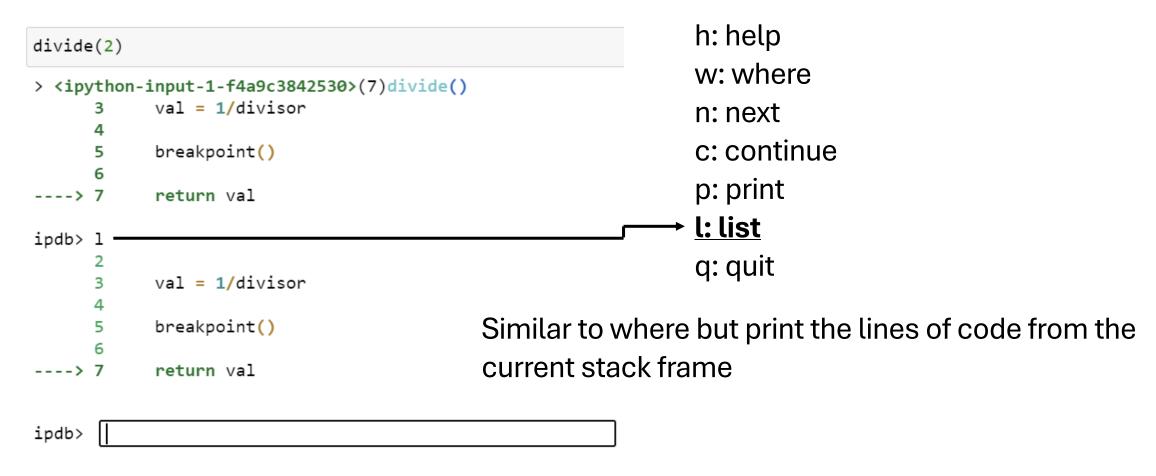


```
h: help
divide(2)
                                                                  w: where
> <ipython-input-1-a669feb31165>(5)divide()
           breakpoint()
                                                                  n: next
           val = 1/divisor
                                                                  c: continue
                                                                  p: print
           breakpoint()
                                                                  l: list
ipdb> c
> <ipython-input-1-a669feb31165>(9)divide()
                                                                  q: quit
           val = 1/divisor
           breakpoint()
           return val
                                                          Run the code until the next breakpoint()
ipdb>
```



Print the value of the desired variable







```
divide(2)
                                                                                      h: help
> <ipython-input-1-f4a9c3842530>(7)divide()
          val = 1/divisor
                                                                                      w: where
          breakpoint()
                                                                                      n: next
---> 7
         return val
ipdb> a
                                                                                      c: continue
                                                                                      p: print
BdbQuit
                                   Traceback (most recent call last)
<ipvthon-input-2-cfdb0f794c84> in <module>
----> 1 divide(2)
                                                                                      l: list
<ipython-input-1-f4a9c3842530> in divide(divisor)
          breakpoint()
                                                                                      q: quit
---> 7
         return val
<ipython-input-1-f4a9c3842530> in divide(divisor)
          breakpoint()
---> 7
         return val
~\anaconda3\lib\bdb.py in trace dispatch(self, frame, event, arg)
                return # None
    87
             if event == 'line':
                                                                    Exit the debugger console
---> 88
                return self.dispatch line(frame)
             if event == 'call':
                return self.dispatch call(frame, arg)
~\anaconda3\lib\bdb.py in dispatch line(self, frame)
                                                         Useful video tutorial of basic usage of Python
             if self.stop_here(frame) or self.break_here(frame):
   111
   112
                self.user line(frame)
                                                         debugger:
                if self.quitting: raise BdbQuit
--> 113
             return self.trace_dispatch
   114
   115
                                                        https://www.youtube.com/watch?v=aZJnGOwzHtU
```

BdbQuit:



Using Google/Stack Overflow

