

40-debugging

April 29, 2016

1 Debugging

- the tried and true - stick print statements in your code

```
In [1]: def fact(arg):
        print('arg', arg)
        ret = 1 if arg == 0 else arg * fact(arg-1)
        print('ret', ret)
        return(ret)

        fact(3)

arg 3
arg 2
arg 1
arg 0
ret 1
ret 1
ret 2
ret 6
```

```
Out[1]: 6
```

2 assert statement

- 1st arg is an expression
- 2nd arg is error message
- if the expression is false, an AssertionError is raised
- if optimization is turned on (-O on command line) asserts are removed

```
In [2]: def a(n):
        assert n == 4, 'did not get 4!!'
```

```
In [3]: a(4)
```

```
In [4]: a(6)
```

```
-----
AssertionError                                Traceback (most recent call last)

<ipython-input-4-28583940927d> in <module>()
----> 1 a(6)
```

```

<ipython-input-2-9a1183db0062> in a(n)
      1 def a(n):
----> 2     assert n == 4, 'did not get 4!!'

```

```

AssertionError: did not get 4!!

```

3 pdb - Python debugger

- two ways to run it

```

In [ ]: import pdb

```

```

def foo(n):
    a = 2 * n
    b = a * 3.14
    if a < b:
        print('b', b)
    else:
        print('a', a)

```

```

pdb.runeval('foo(3)')

```

```

> <string>(1)<module>()
(Pdb) s
--Call--
> <ipython-input-5-2f567135e614>(3)foo()
-> def foo(n):
(Pdb) s
> <ipython-input-5-2f567135e614>(4)foo()
-> a = 2 * n
(Pdb) s
> <ipython-input-5-2f567135e614>(5)foo()
-> b = a * 3.14
(Pdb) s
> <ipython-input-5-2f567135e614>(6)foo()
-> if a < b:
(Pdb) s
> <ipython-input-5-2f567135e614>(7)foo()
-> print('b', b)
(Pdb) s
--Call--
> /Users/lstead/anaconda/lib/python3.5/site-packages/ipykernel/iostream.py(207)write()
-> def write(self, string):
(Pdb) s
> /Users/lstead/anaconda/lib/python3.5/site-packages/ipykernel/iostream.py(208)write()
-> if self.pub_socket is None:
(Pdb) s
> /Users/lstead/anaconda/lib/python3.5/site-packages/ipykernel/iostream.py(212)write()
-> if not isinstance(string, unicode_type):
(Pdb) s

```

```

> /Users/lstead/anaconda/lib/python3.5/site-packages/ipykernel/iostream.py(215)write()
-> is_child = (self._check_mp_mode() == CHILD)
(Pdb) s
--Call--
> /Users/lstead/anaconda/lib/python3.5/site-packages/ipykernel/iostream.py(102)_check_mp_mode()
-> def _check_mp_mode(self):
(Pdb) s
> /Users/lstead/anaconda/lib/python3.5/site-packages/ipykernel/iostream.py(104)_check_mp_mode()
-> if not self._pipe_flag or self._is_master_process():
(Pdb) s
--Call--
> /Users/lstead/anaconda/lib/python3.5/site-packages/ipykernel/iostream.py(93)_is_master_process()
-> def _is_master_process(self):

```

```

In [ ]: def bar(n):
        a = 2 * n
        b = a * 3.14
        # debug will start here
        pdb.set_trace()
        if a < b:
            print('b', b)
        else:
            print('a', a)

```

```

In [ ]: bar(4)

```

```

In [ ]:

```