

localhost:8888/notebooks/DSML%20Intermediate/Mine/Cl...

File Edit View Insert Cell Kernel Widgets Help Trusted Python 3 (ipykernel) O

Exception Handling

In [1]: `print(1/0)`

ZeroDivisionError
1 last)
Input In [1], in <cell line: 1>()
----> 1 print(1/0)

ZeroDivisionError: division by zero

In []:

1/1

The image shows a Jupyter Notebook interface running in Microsoft Edge. The title bar indicates the URL is localhost:8888/notebooks/DSML%20Intermediate/Mine/Cl... The menu bar includes File, Edit, View, Insert, Cell, Kernel, Widgets, Help, Trusted, and Python 3 (ipykernel). The main area displays a section titled 'Exception Handling'. In cell [1], the code 'print(1/0)' is run, resulting in a 'ZeroDivisionError' exception. The error message 'ZeroDivisionError: division by zero' is highlighted with a red box and a red question mark is placed next to it. The bottom cell is empty, labeled 'In []:'.

File Edit View Insert Cell Kernel Widgets Help Trusted | Python 3 (ipykernel) 0

With error

Decision 2

```
In [28]: def send_sms(phone_num):  
    # use some 3rd party APIs  
    if phone_num % 10 == 4:  
        print('Message is NOT successfully sent')  
        return  
    print('Message is successfully sent')
```

```
In [29]: contact_nums = [9102341411, 9102341412, 9102341413, 9102341414]
```

```
In [30]: for i in contact_nums:  
    send_sms(i)
```

Message is successfully sent

Message is successfully sent

Message is successfully sent

Message is NOT successfully sent

In []:

In []:

localhost:8888/notebooks/DSML%20Intermediate/Mine/Cl...

File Edit View Insert Cell Kernel Widgets Help Trusted Python 3 (ipykernel) O

```
print('Message is successfully sent')
```

In [29]: contact_nums = [9102341411, 9102341412, 9102341413, 9102341414]

In [30]: for i in contact_nums:
 send_sms(i)

Message is successfully sent
Message is successfully sent
Message is successfully sent
Message is NOT successfully sent

In [31]: try:
 for i in contact_nums:
 send_sms(i)
except:
 # try again
 pass

Message is successfully sent
Message is successfully sent
Message is successfully sent
Message is NOT successfully sent

In []:

localhost:8888/notebooks/DSML%20Intermediate/Mine/Cl...

File Edit View Insert Cell Kernel Widgets Help Trusted Python 3 (ipykernel) O

With error

```
In [35]: def send_sms(phone_num):
    # use some 3rd party APIs
    if phone_num % 10 == 4:
        # print('Message is NOT successfully sent')
        raise Exception("This is a bug")
    return
    print('Message is successfully sent')
```

```
In [36]: contact_nums = [9102341411, 9102341412, 9102341413, 9102341414]
```

```
In [37]: for i in contact_nums:
    send_sms(i)
```

Message is successfully sent
Message is successfully sent
Message is successfully sent

Exception

1 last)
Input In [37], in <cell line: 1>()
1 for i in contact_nums:
----> 2 send_sms(i)

Traceback (most recent call

localhost:8888/notebooks/DSML%20Intermediate/Mine/Cl...

File Edit View Insert Cell Kernel Widgets Help Trusted Python 3 (ipykernel)

```
In [37]: for i in contact_nums:  
    send_sms(i)
```

Message is successfully sent
Message is successfully sent
Message is successfully sent

Exception
1 last)
Input In [37], in <cell line: 1>()
1 for i in contact_nums:
----> 2 send_sms(i)

Input In [35], in send_sms(phone_num)
1 def send_sms(phone_num):
2 # use some 3rd party APIs
3 if phone_num % 10 == 4:
4 # print('Message is NOT successfully sent')
----> 5 raise Exception("This is a bug")
6 return
7 print('Message is successfully sent')

Exception: This is a bug

```
In [31]: try:
```

localhost:8888/notebooks/DSML%20Intermediate/Mine/Cl...

File Edit View Insert Cell Kernel Widgets Help Trusted Python 3 (ipykernel) O

In [31]:

```
try:  
    for i in contact_nums:  
        send_sms(i)  
except:  
    # try again  
    pass
```

Message is successfully sent
Message is successfully sent
Message is successfully sent
Message is NOT successfully sent

In [38]:

```
1/0
```

ZeroDivisionError

1 last)
Input In [38], in <cell line: 1>()
----> 1 1/0

ZeroDivisionError: division by zero

Traceback (most recent call)

In []:

localhost:8888/notebooks/DSML%20Intermediate/Mine/Cl...

File Edit View Insert Cell Kernel Widgets Help Trusted Python 3 (ipykernel) O

send_sms()

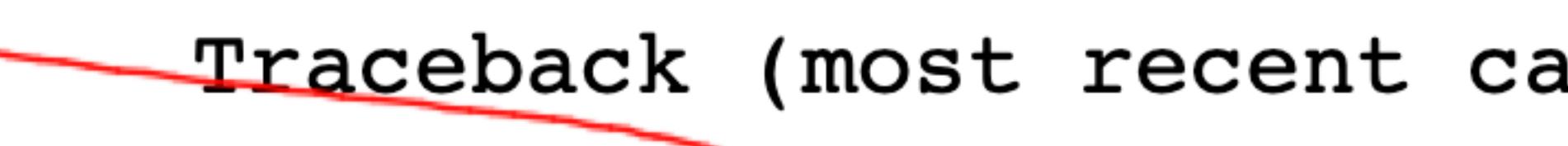
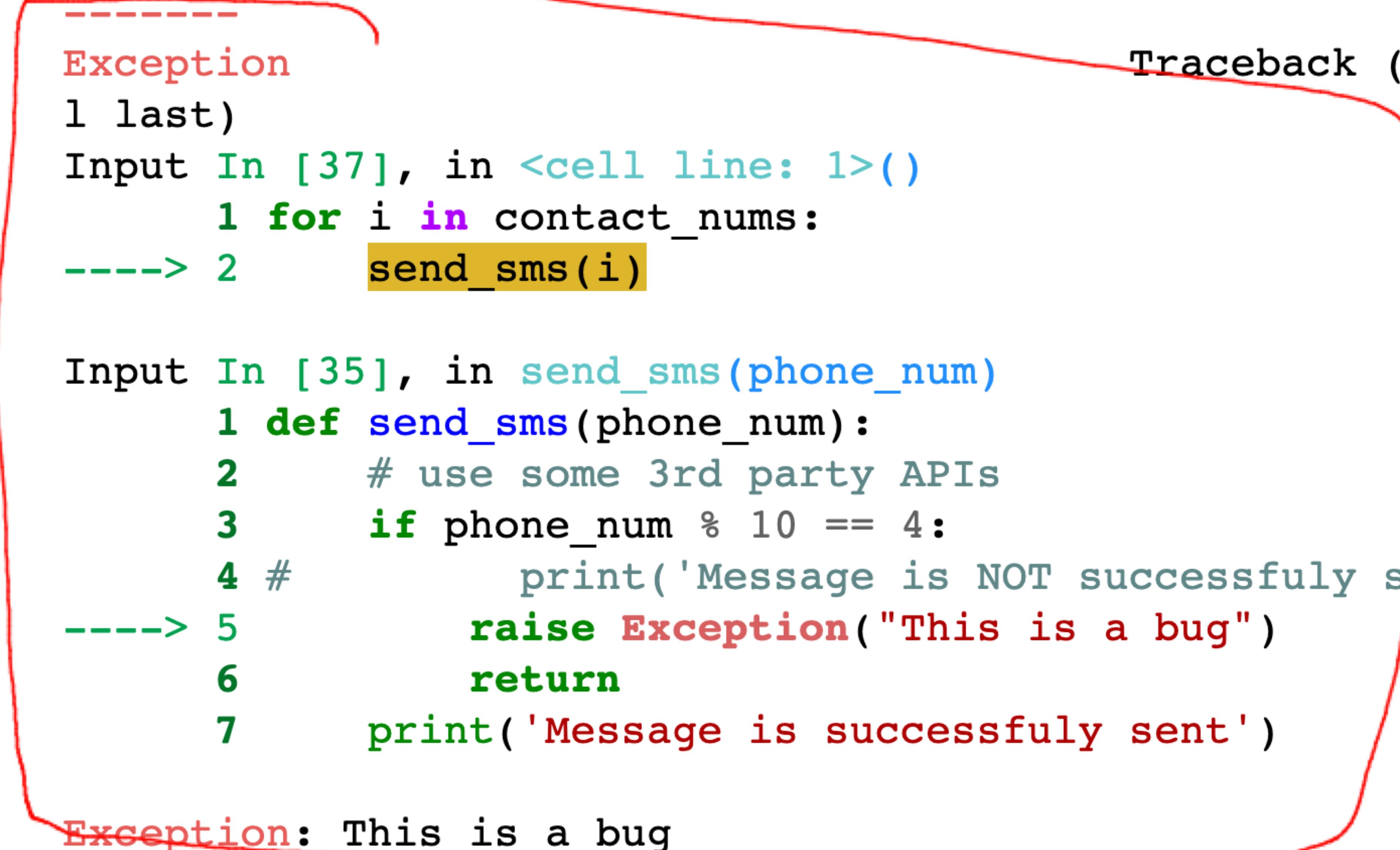
Message is successfully sent
Message is successfully sent
Message is successfully sent

Exception Traceback (most recent call last)
Input In [37], in <cell line: 1>()
1 for i in contact_nums:
----> 2 send_sms(i)

Input In [35], in send_sms(phone_num)
1 def send_sms(phone_num):
2 # use some 3rd party APIs
3 if phone_num % 10 == 4:
4 # print('Message is NOT successfully sent')
----> 5 raise Exception("This is a bug")
6 return
7 print('Message is successfully sent')

Exception: This is a bug

In [31]: try:
for i in contact_nums:



localhost:8888/notebooks/DSML%20Intermediate/Mine/Cl...

File Edit View Insert Cell Kernel Widgets Help Trusted Python 3 (ipykernel) 🔎

```
----> 5         raise Exception("This is a bug")
               6         print('Message is successfully sent')
```

Exception: This is a bug

In [50]: contact_nums = [9102341411, 9102341412, 9102341413, 9102341414, 9102341415]

In [51]: try:
 for i in contact_nums:
 send_sms(i)
 except:
 # try again
 pass

Message is successfully sent ✓
Message is successfully sent ✓
Message is successfully sent ✓

In []:

In []:

In []:

8/8

Python Tutor: Visualize code in Python, JavaScript, C, C++, and Java

Python 3.6
[\(known limitations\)](#)

```
1 contact_nums = [9102341411, 9102341412, 9102341413]
2
3 def send_sms(phone_num):
4     # use some 3rd party APIs
5     if phone_num % 10 == 4:
6         # print('Message is NOT successfully sent')
7         raise Exception("This is a bug")
8     print('Message is successfully sent')
9
10 try:
11     for i in contact_nums:
12         send_sms(i)
13 except:
14     # try again
15     pass
```

[Edit this code](#)

→ line that just executed
→ next line to execute

<< First < Prev Next > Last >>

Step 26 of 31

[Customize visualization](#)

9 / 9

Print output (drag lower right corner to resize)

```
Message is successfully sent
Message is successfully sent
Message is successfully sent
```

Frames	Objects
Global frame	
contact_nums	list
send_sms	function
i	9102341414

send_sms
phone_num 9102341414

ORACLE

Trusted

Python 3 (ipykernel) 0

File Edit View Insert Cell Kernel Widgets Help

Example 2

```
In [53]: def something(x):
    print(1 // x)
    print("A")
```

```
In [ ]: try:  
    # can this code given an exception  
    something(3) → 0  
    something(4) →   
    something(0) →   
    something(1) →   
except:  
    print("Some error has occurred")
```

In [1]:

Notebook saved

Trusted

Python 3 (ipykernel)

File Edit View Insert Cell Kernel Widgets Help

```
In [*]: try:  
    number1 = int(input())  
    print(number1 * 3)  
  
    number2 = int(input())  
    print(number2 * 3)  
  
except:  
    print('x')  
    print('e')
```

$$\text{int}(c_2)$$

2

~~x~~

In [1]:

Trusted

Python 3 (ipykernel) 0

File Edit View Insert Cell Kernel Widgets Help

0

A

Some error has occurred

Quizzes

```
In [57]: try:  
    number1 = int(input())  
    print(number1 * 3)  
  
    number2 = int(input())  
    print(number2 * 3)  
  
except:  
    print('x')  
    print('e')
```

Z

x

e

449.2 ms × 4

InterviewBit Practice Resources Contests Online IDE Free Mock Scaler

index.py Python 3 Run

```
1 try:
2     number1 = int(input())
3     print(number1 * 3)
4
5     number2 = int(input())
6     print(number2 * 3)
7 except:
8     print('x')
9     print('e')
```

Custom Input ⓘ

7
9
...

Output

[Success] Your code was executed successfully

x
e

13 / 13



Got suggestions? we would love to hear your feedback



Trusted

Python 3 (ipykernel) C

File Edit View Insert Cell Kernel Widgets Help

```
ValueError: invalid literal for int() with base 10: 'z'
```

In [60]: **try:**

```
try:  
    number1 = int(input())  
    print(number1 * 3)  
  
    number2 = int(input())  
    print(number2 * 3)  
  
except:  
    print('x')  
print('e')
```

7

2

e

2

2

449.2 ms × 4

InterviewBit Practice Resources Contests Online IDE Free Mock Scaler

index.py Python 3 Run

```
1 try:
2     number1 = int(input())
3     print(number1 * 3)    2 1
4
5     number2 = int(input())
6     print(number2 * 3)    2 7
7     except:
8         print('x')
9     print('e')
```

Custom Input ⓘ

7
9
...

Output

[Success] Your code was executed successfully

21
27
e

if _____:
else:

15 / 15



Got suggestions? we would love to hear your feedback



449.2 ms × 4

InterviewBit Practice Resources Contests Online IDE Free Mock Scaler

index.py Python 3 Run

```
1 try:
2     number1 = int(input())
3     print(number1 * 3)
4
5     number2 = int(input())
6     print(number2 * 3)
7 except:
8     print('x')
9 print('e')
```

Custom Input ⓘ

z
9
...

Output

[Success] Your code was executed successfully

x
e



Got suggestions? we would love to hear your feedback



localhost:8888/notebooks/DSML%20Intermediate/Mine/Cl...

File Edit View Insert Cell Kernel Widgets Help Trusted Python 3 (ipykernel) O

In [64]:

```
def something(x):
    → print(1 // x)
    return 1
```

//*hello*

In [65]:

```
for e in l:
    try:
        # doing some operation on e
        print(f'Current element - {e}')
        → res = something(e)
        print(f'Result - {res}')
    except:
        → print('Error occurred!')

    print('-'*25)
```

[2, 0, 'hello']

Current element - 2

✓ 0

✓ Result - 1

✓ Current element - 0

✓ Error occurred!

Current element - hello

Error occurred!

Current element - None

Error occurred!

17 / 17

localhost:8888/notebooks/DSML%20Intermediate/Mine/Cl...

In [76]:

```
for e in l:
    try:
        # doing some operation on e
        print(f'Current element - {e}')
        res = something(e)
        print(f'Result - {res}')
    except ZeroDivisionError: # as soon as an error comes => catch it
        print('Division by Zero occurred!')
    except TypeError:
        print('Input is invalid')

    print('-'*25)
```

Current element - 2
Result - 0

Current element - 0
Division by Zero occurred!

Current element - hello
Input is invalid

Current element - None
Input is invalid

if

elif

elif

Python Tutor: Visualize code in Python, JavaScript, C, C++, and Java

Python 3.6
([known limitations](#))

```
1 l = [2, 0, 'hello', None]
2
3 def something(x):
4     return (1 // x) raise Exception?
5
6 for e in l:
7     try:
8         # doing some operation on e
9         print(f'Current element - {e}')
10        res = something(e)
11        print(f'Result - {res}')
12    except ZeroDivisionError: # as soon as an erro
13        print('Division by Zero occurred!')
14    except TypeError:
15        print('Input is invalid')
16
17 print('*'*25)
```

Print output (drag lower right corner to resize)

```
Current element - 2
Result - 0
-----
Current element - 0
```

Frames Objects

Global frame

I	list
something	0 1 2 "hello" None
e	0
res	0

function something(x)

something

x	0
---	---

line that just executed

next line to execute

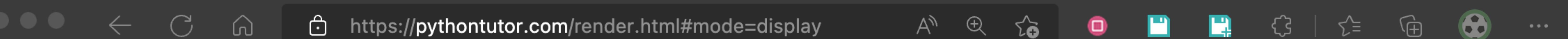
<< First < Prev Next > > Last >>

Step 17 of 50

[Edit this code](#)

[Customize visualization](#)

19 / 19



Python Tutor: Visualize code in Python, JavaScript, C, C++, and Java

Python 3.6
([known limitations](#))

```

1 l = [2, 0, 'hello', None]
2
3 def something(x):
4     return (1 // x)
5
6 for e in l:
7     try:
8         # doing some operation on e
9         print(f'Current element - {e}')
10        res = something(e)
11        print(f'Result - {res}')
12    except ZeroDivisionError: # as soon as an erro
13        print('Division by Zero occurred!')
14    except TypeError:
15        print('Input is invalid')
16
17    print('*'*25)

```

[Edit this code](#)

→ line that just executed
→ next line to execute

<< First < Prev Next > >> Step 32 of 50

TypeError: unsupported operand type(s) for //: 'int' and 'str'
(see [UNSUPPORTED FEATURES](#))

Print output (drag lower right corner to resize)

```

-----
Current element - 0
Division by Zero occurred!
-----
Current element - hello

```

Frames	Objects
Global frame	list 0 1 2 "hello" 3 2 0 None
something	function something(x)
e "hello"	
res 0	

Python Tutor: Visualize code in Python, JavaScript, C, C++, and Java

Python 3.6
([known limitations](#))

```
1 l = [2, 0, 'hello', None]
2
3 def something(x):
4     return (1 // x)
5
6 for e in l:
7     try:
8         # doing some operation on e
9         print(f'Current element - {e}')
10        res = something(e)
11        print(f'Result - {res}')
12    except ZeroDivisionError: # as soon as an erro
13        print('Division by Zero occurred!')
14    except TypeError:
15        print('Input is invalid')
16
17    print('*'*25)
```

Print output (drag lower right corner to resize)

Division by Zero occurred!

Current element - hello
Input is invalid

Frames Objects

Global frame

I		list	0	1	2	"hello"	3
something			2	0			None
e	"hello"						
res	0						

function something(x)

line that just executed

next line to execute

<< First < Prev Next > > Last >>

Step 37 of 50

[Edit this code](#)

[Customize visualization](#)

21 / 21

localhost:8888/notebooks/DSML%20Intermediate/Mine/Cl...

File Edit View Insert Cell Kernel Widgets Help Trusted Python 3 (ipykernel) O

Current element - hello
Input is invalid

Current element - None
Input is invalid

In [79]:

```
ll = [2, 0, "hello", None]
for e in ll:
    try:
        result = 5 / int(e)
        print("N")
    except Exception as ex:
        print("E")
    except ZeroDivisionError as z:
        print("Z")
```

N
E
E
E

In []:

In []:

22 / 22

localhost:8888/notebooks/DSML%20Intermediate/Mine/Cl...

File Edit View Insert Cell Kernel Widgets Help Trusted Python 3 (ipykernel) O

```
except ZeroDivisionError as z:  
    print("Z")
```

N
E
E
E

In [80]: l1 = [2, 0, "hello", None] ✓ ↴
for e in l1:
 try:
 result = 5 / int(e)
 print("N")
 except ZeroDivisionError as z:
 print("Z")
 except Exception as ex:
 print("E")

N
Z
E
E

In []:

23 / 23

B.py – DSML Intermediate

```
lsml-jun-22-beginner-intermediate-main > Modules > B.py
1 import A
2
3 print(A.add(5, 6))
4
5 print(f'__name__ in B.py {__name__}')
```

```
● dsml-jun-22-beginner-intermediate-main/Modules$ python B.py
This is something over here in A.py
__name__ in A.py A
11
__name__ in B.py __main__
(base) sahilbansal:~/Downloads/DSML Intermediate/
○ dsml-jun-22-beginner-intermediate-main/Modules$
```

