



EXPLORER

1 OPEN EDITORS 1 UNSAVED

- User Settings C:\Users\pcurtis7\A...
- percipio67_chaining_exceptions...

PYTHON

- writefile.txt
- Percipio_Python3-Course
 - 01_Start
 - 02_Data-Sequence Types
 - 03_Collections-Mapping-Looping
 - 04_Modules-Functions
 - 05_Classes
 - 06_Working-with-Files
 - 07_Comprehensions
 - 08_Iterables-and-Generators
 - 09_Exceptions
 - percipio60_catching_all_excepti...
 - percipio61_catching_specific_ex...
 - percipio62_the_exception_hiera...
 - percipio62b_all_python_objects...
 - percipio63_exception_payloads...
 - percipio64_creating_new_excep...
 - percipio65_traceback_objects.py
 - percipio66_assertions.py
 - percipio67_chaining_exceptions..
 - 10_Automation Programming
- Python Projects_2014
- CMD_Python_Set-Path.txt
- excel_code.py
- excel_code_summary_master
- excel_code_summary_master - Cop...
- excel_code_summary_master.py
- python_all_objects_and_subclasses....
- Python_Clear-Window-Command.txt
- python_debug_logging_code.py
- python_exercises_00.py
- python_exercises_01.py
- Python_Tutorial_Running-Scripts.do...
- Python_Tutorials.md
- Scripts - Shortcut.lnk

User Settings percipio67_chaining_exceptions.py

```
1 '''
2 percipio67_chaining_exceptions.py
3 Percipio video: Exceptions; Chaining Exceptions
4
5 * A file with 3 different functions and 1 function call to demonstrate how implicit and
  explicit exception chaining occurs in Python.
6
7 * Python 3 will implicitly chain exceptions but code can be added to the exception handler
  to explicit chain from the exception object created.
8
9 * When using 'implicit exception chaining' the traceback shows:
10 | * "During handling of the above exception, another exception occurred"
11
12 * When explicitly using 'explicit exception chaining' with "from exception", the traceback
  shows:
13 | * "The above exception was the direct cause of the following exception"
14 | * When you see this message, you know that an explicit exception was chained
15
16 '''
17
18 # QUESTION: What is the chain referred to?
19 # QUESTION: "While trying to handle the link3 problem, there was a call to link2", WHY was
  there a call to link2? The link3 RuntimeError doesn't stop the program and it simply
  continues on after sending the RuntimeError message? Is this right?
20
21 # implicit exception chaining:
22 print('Implicit exception chaining" example')
23
24 def link3(): # this function simply prints and raises a RuntimeError with a message
25     print('link3 in the chain') # prints it's name on the chain
26     raise RuntimeError('link3 problem')
27
28 def link2(): #
29     print('link2 in the chain') # also prints it's name on the chain
30     try: # tries to execute link3 unless there is a RuntimeError as an exception
31         link3()
32     except RuntimeError as exception: # if link3 causes an error, then link2 raises a
33         RuntimeError
34         raise RuntimeError('link2 problem') #
35
36 def link1():
37     print('link1 in the chain') # also prints it's name on the chain
38     try: # tries to execute link2 unless there is a RuntimeError as an exception
39         link2()
40     except RuntimeError as exception: # if link2 causes an error, then link1 raises a
```





EXPLORER

1

OPEN EDITORS 1 UNSAVED

- User Settings C:\Users\pcurtis7\A...
- percipio67_chaining_exceptions.py

PYTHON

- writefile.txt
- Percipio_Python3-Course
 - 01_Start
 - 02_Data-Sequence Types
 - 03_Collections-Mapping-Looping
 - 04_Modules-Functions
 - 05_Classes
 - 06_Working-with-Files
 - 07_Comprehensions
 - 08_Iterables-and-Generators
 - 09_Exceptions
 - percipio60_catching_all_excepti...
 - percipio61_catching_specific_ex...
 - percipio62_the_exception_hiera...
 - percipio62b_all_python_objects...
 - percipio63_exception_payloads...
 - percipio64_creating_new_excep...
 - percipio65_traceback_objects.py
 - percipio66_assertions.py
 - percipio67_chaining_exceptions..
 - 10_Automation Programming
- Python Projects_2014
- CMD_Python_Set-Path.txt
- excel_code.py
- excel_code_summary_master
- excel_code_summary_master - Cop...
- excel_code_summary_master.py
- python_all_objects_and_subclasses....
- Python_Clear-Window-Command.txt
- python_debug_logging_code.py
- python_exercises_00.py
- python_exercises_01.py
- Python_Tutorial_Running-Scripts.do...
- Python_Tutorials.md
- Scripts - Shortcut.lnk

User Settings percipio67_chaining_exceptions.py

```
33 except RuntimeError as exception: # if link2 causes an error, then link1 raises a
    RuntimeError
34     raise RuntimeError('link1 problem') #
35 link1() # link1() function is the only function called which starts a chain reaction.
36
37 # explicit exception chaining:
38 print('"Explicit exception chaining" example')
39 '''
40 * When using 'implicit exception chaining' the traceback shows:
41     * "During handling of the above exception, another exception occurred"
42 * When explicitly using 'explicit exception chaining' with "from exception", the traceback
    shows:
43     * "The above exception was the direct cause of the following exception"
44     * When you see this message, you know that an explicit exception was chained
45 '''
46 def link3(): #
47     print('link3 in the chain') #
48     raise RuntimeError('link3 problem')
49 def link2(): #
50     print('link2 in the chain') #
51     try: #
52         link3()
53     except RuntimeError as exception: # 'exception' object created
54         raise RuntimeError('link2 problem') from exception #
55         ''' 'exception handler' code so here is the use of 'explicit exception chaining'.
            Instead of simply raising the RuntimeError, 'from exception' is added where that
            exception is this exception object that was the RuntimeError being handled when
            executing link3 (in code above, 'RuntimeError as exception')
56         '''
57 def link1():
58     print('link1 in the chain') #
59     try: #
60         link2()
61     except RuntimeError as exception: #
62         raise RuntimeError('link1 problem') # this could have used 'explicit exception
            chaining', but it was simply not used here
63 link1() #
64 '''
65 RESULT:
```





EXPLORER

1

OPEN EDITORS 1 UNSAVED

- User Settings C:\Users\pcurtis7\A...
- percipio67_chaining_exceptions...

PYTHON

- writefile.txt
- Percipio_Python3-Course
 - 01_Start
 - 02_Data-Sequence Types
 - 03_Collections-Mapping-Looping
 - 04_Modules-Functions
 - 05_Classes
 - 06_Working-with-Files
 - 07_Comprehensions
 - 08_Iterables-and-Generators
 - 09_Exceptions
 - percipio60_catching_all_excepti...
 - percipio61_catching_specific_ex...
 - percipio62_the_exception_hiera...
 - percipio62b_all_python_objects...
 - percipio63_exception_payloads...
 - percipio64_creating_new_excep...
 - percipio65_traceback_objects.py
 - percipio66_assertions.py
 - percipio67_chaining_exceptions..
 - 10_Automation Programming
- Python Projects_2014
- CMD_Python_Set-Path.txt
- excel_code_py
- excel_code_summary_master
- excel_code_summary_master - Cop...
- excel_code_summary_master.py

User Settings

percipio67_chaining_exceptions.py

```
65 RESULT:
66 "Implicit exception chaining" example
67 link1 in the chain
68 link2 in the chain
69 link3 in the chain
70 Traceback (most recent call last):
71   File "C:/Python36x64/test.py", line 9, in link2
72     link3()
73   File "C:/Python36x64/test.py", line 5, in link3
74     raise RuntimeError('link3 problem')
75 RuntimeError: link3 problem
76
77 During handling of the above exception, another exception occurred:
78
79 Traceback (most recent call last):
80   File "C:/Python36x64/test.py", line 15, in link1
81     link2()
82   File "C:/Python36x64/test.py", line 11, in link2
83     raise RuntimeError('link2 problem') #
84 RuntimeError: link2 problem
85
86 During handling of the above exception, another exception occurred:
87
88 Traceback (most recent call last):
89   File "C:/Python36x64/test.py", line 18, in <module>
90     link1()
91   File "C:/Python36x64/test.py", line 17, in link1
92     raise RuntimeError('link1 problem') #
93 RuntimeError: link1 problem
94
95
```





EXPLORER

{ } User Settings

percipio67_chaining_exceptions.py

OPEN EDITORS 1 UNSAVED

{ } User Settings C:\Users\pcurtis7\A...

percipio67_chaining_exceptions...

PYTHON

writefile.txt

Percipio_Python3-Course

01_Start

02_Data-Sequence Types

03_Collections-Mapping-Looping

04_Modules-Functions

05_Classes

06_Working-with-Files

07_Comprehensions

08_Iterables-and-Generators

09_Exceptions

percipio60_catching_all_excepti...

percipio61_catching_specific_ex...

percipio62_the_exception_hiera...

percipio62b_all_python_objects...

percipio63_exception_payloads...

percipio64_creating_new_excep...

percipio65_traceback_objects.py

percipio66_assertions.py

percipio67_chaining_exceptions..

10_Automation Programming

Python Projects_2014

CMD_Python_Set-Path.txt

excel_code_py

excel_code_summary_master

excel_code_summary_master - Cop...

excel_code_summary_master.py

94

95

96

97

98

99

100

101

102

103

104

105

106

107

108

109

110

111

112

113

114

115

116

117

118

119

120

121

122

123

124

"Explicit exception chaining" example

link1 in the chain

link2 in the chain

link3 in the chain

Traceback (most recent call last):

File "C:/Python36x64/test.py", line 15, in link2

link3()

File "C:/Python36x64/test.py", line 11, in link3

raise RuntimeError('link3 problem')

RuntimeError: link3 problem

The above exception was the direct cause of the following exception:

Traceback (most recent call last):

File "C:/Python36x64/test.py", line 23, in link1

link2()

File "C:/Python36x64/test.py", line 17, in link2

raise RuntimeError('link2 problem') from exception

RuntimeError: link2 problem

During handling of the above exception, another exception occurred:

Traceback (most recent call last):

File "C:/Python36x64/test.py", line 26, in <module>

link1()

File "C:/Python36x64/test.py", line 25, in link1

raise RuntimeError('link1 problem')

RuntimeError: link1 problem

'''

