Exception Handling with Python

Contents

[Exception Handling with Python 3](#_Toc93175229)

[Python Try Except 3](#_Toc93175230)

# Exception Handling with Python

# Python Try Except

|  |  |
| --- | --- |
| * The try block lets you test a block of code for errors. * The except block lets you handle the error. * The finally block lets you execute code, regardless of the result of the try- and except blocks. |  |
| **Exception Handling**   * When an error occurs, or exception as we call it, Python will normally stop and generate an error message. * These exceptions can be handled using the try statement: | try:   print(x) except:   print("An exception occurred") |
| **Many Exceptions**   * You can define as many exception blocks as you want | try:   print(x) except NameError:   print("Variable x is not defined") except:   print("Something else went wrong") |
| **Else**   * You can use the else keyword to define a block of code to be executed if no errors were raised: | In this example, the try block does not generate any error:  try:   print("Hello") except:   print("Something went wrong") else:   print("Nothing went wrong") |
| **Finally**   * The finally block, if specified, will be executed regardless if the try block raises an error or not. | try:   print(x) except:   print("Something went wrong") finally:   print("The 'try except' is finished") |
| **Try to open and write to a file that is not writable:** | try:   f = open("demofile.txt")   try:     f.write("Lorum Ipsum")   except:     print("Something went wrong when writing to the file")   finally:     f.close() except:   print("Something went wrong when opening the file") |
| **Raise an exception**   * To throw (or raise) an exception if a condition occurs during processing, use the raise keyword. | x = -1  if x < 0:   raise Exception("Sorry, no numbers below zero") |
|  | Raise a TypeError if x is not an integer:  x = "hello"  if not type(x) is int:   raise TypeError("Only integers are allowed") |

# References:

* <https://en.wikipedia.org/wiki/Exception_handling>
* [8. Errors and Exceptions](https://docs.python.org/3/tutorial/errors.html)
* [Built-in Exceptions](https://docs.python.org/3/library/exceptions.html#exception-context)