

## ZeroDivisionError

```
value = input()
result = str(5 / int(value))
print("5 divided by " + value + " = " + result)
```

0

```
-----
-----
ZeroDivisionError                                Traceback (most
recent call last)
<ipython-input-4-d3ba6166a0d9> in <module>
      1 value = input()
----> 2 result = str(5 / int(value))
      3 print("5 divided by " + value + " = " + result)

ZeroDivisionError: division by zero
```

Im oberen Bild sieht man, dass die Anwendung abstürzt, wenn Benutzer die **Zahl Null** als Input verwenden. Der vom Interpreter ausgegebene Fehler ist die Ausnahme `ZeroDivisionError`. Der Name gibt uns einen Hinweis darauf, warum die Anwendung abgestürzt ist. Er sagt uns, dass das Programm versucht hat, etwas durch null zu teilen. Solltest du jemals über einen Fehler stolpern, den du nicht verstehst, findest du im Internet zahlreiche hilfreiche Ressourcen.

Schauen wir uns noch ein paar Aspekte der Fehlermeldung an. Am Ende der Meldung siehst du eine saubere Beschreibung des Fehlers. In diesem Fall steht dort „division by zero“, also „Teilung durch null“, was die Dinge etwas klarer macht, falls du verwirrt gewesen sein solltest. Weiterhin zeigt der Interpreter mit einem Pfeil auf die Stelle im Code, wo der Fehler aufgetreten ist, genau, wie er es auch bei Syntaxfehlern macht. In diesem Fall sehen wir, dass wir in **Zeile 2** durch eine Variable teilen und diesen Fehler erhalten, wenn diese Variable null entspricht.

Schauen wir uns eine weitere Ausnahme an:

## Eine Datei öffnen und lesen

```
file_name = input()
my_file = open(file_name, "r")
print(my_file.read())
my_file.close()
```

Im Code oben sollen Benutzer einen Wert eingeben, den wir dann in einer Variable namens „file\_name“ speichern. Dann öffnen wir eine Datei mit dem Namen, der in *file\_name*

**Kommentiert [AP1]:** I've specified "the number zero" because "null" is a concept in programming. And while in Python, null is represented by "None", I thought it best to just avoid any potential confusion.

**Kommentiert [AP2]:** Lower numbers are usually written out in German, but I've used a numeral here to make it easier to reference the correct line in the code, as well as to keep things more consistent throughout the book.

gespeichert ist. Das zweite Argument der `open` Funktion, `"r"`, gibt an, dass wir die Datei zum Lesen öffnen. Danach lesen wir die Datei aus, zeigen ihren Inhalt an und schließen die Datei wieder.

Wenn du diesen Code ausführst und einen gültigen Dateinamen angibst, sollte der Inhalt dieser Datei auf dem Bildschirm ausgegeben werden. Gibst du allerdings einen ungültigen Dateinamen an, verursacht das eine Ausnahme.

**Kommentiert [AP3]:** Suggestion: Format words that refer to variables, functions etc. in code in a monospace font. This makes it clearer for the reader and also avoids any potential confusion should quotes appear in the code itself (for example: Now copy `print("hello")` from line 12...)