



Exceptions

חריגות (Exceptions)

פייתון תומכת בטיפול חריגות. במקרה שתוכנית נתקלת בסיטואציה חריגה, על פי רוב שגיאה או נתונים עבורם תוצאת החישוב הרצויה איננה מוגדרת, ניתן "להרים" (או "לזרוק") חריגה באמצעות המילה השמורה **raise** (מקביל ל-**throw** בשפות אחרות). החריגה הנזרקת היא אובייקט, שיילכד במעלה הקריאה לפונקציה שזרקה אותו, בבלוק ייעודי מהצורה **try ... except**, שם יטופל באופן ספציפי. אם חריגה לא נלכדת על ידי המתכנת, היא נלכדת על ידי המפרש. בפייתון כל שגיאה היא חריגה, וכל שגיאה ניתן ללכוד - חלוקה באפס, שגיאות קלט-פלט, שמות שטרם הוגדרו בקוד, וכו' - למעט שגיאות תחביר.

1. כתוב פונקציה אשר מקבלת מספר רציונלי – מספר ראשון מונה ומספר שני כמכנה ובעזרת **TUPLE** ומחזירה את שבר העשרוני שלו (הדוגמא מהרצאה 4). אליך לתפוס חריגות הבאות ולהחזיר הודעות מתאימות למשתמש:
 - (א) שהמשתמש לא הכניס מספרים.
 - (ב) חילוק באפס.

```
from operator import getitem
```

```
def make_rat(n, d):
    return (n, d)
def numer(x):
    return getitem(x, 0)
def denom(x):
    return getitem(x, 1)
def make(x):
    .....
    return result
def str_rat(x):
    """Return a string n/d for numerator n and denominator d."""
    return '{0}/{1}={2}'.format(numer(x), denom(x), make(x))
def main():
    n=input("Enter numer number:")
    d=input("Enter denom number:")
    ns=0.0
    .....
    print( str_rat(ns))
```

```
main()
```

בהצלחה !