

Izjeme in datoteke

- ▶ Poglejmo naslednjo kodo

```
imeDatoteke = input("Podaj ime datoteke:  
" )
```

```
datoteka = open(imeDatoteke)
```

```
print(datoteka.read())
```

```
print("To je vse.")
```

- ▶ Kaj se zgodi, če datoteke ni?

```
Podaj ime datoteke: bla.txt
```

```
Traceback (most recent call last):
```

```
  File "C:/Python31/a.py", line 2, in <module>
```

```
    datoteka = open(imeDatoteke)
```

```
IOError: [Errno 2] No such file or directory: 'bla.txt'
```



Kaj pomeni sporočilo

- ▶ V zadnji vrstici nam python sporoča, da je ugotovil vhodno-izhodno napako ("IOError")
- ▶ Ker ne ve kako nadaljevati, prekinja program.
- ▶ Lahko sestavimo svoj program, ki bo "ulovil" to napako.
- ▶ Pythonu v tem primeru ni treba skrbeti zanjo. To naredimo tako:



Ulovljena izjema

- ▶ Poglejmo naslednjo kodo

```
imeDatoteke = input("Podaj ime datoteke: ")
```

```
try:
```

```
    datoteka = open(imeDatoteke)
```

```
    print(datoteka.read())
```

```
except:
```

```
    print("Datoteke", imeDatoteke, "ni!")
```

```
print("To je vse.")
```

- ▶ Kaj se sedaj zgodi, če datoteke ni?

```
>>> test()  
Podaj ime datoteke: bla.txt  
Datoteke bla.txt ni!  
To je vse.  
>>>
```

Ali datoteka obstaja?

- ▶ Sestavimo metodo, ki ugotovi, ali določena datoteka obstaja
- ▶ ... In se delamo kot da še nismo slišali za
- ▶ `os.path.isfile(imeDat)`
- ▶ Ideja:
 - ▶ V varovalnem bloku odpremo datoteko
 - ▶ Seveda za branje (zakaj ne za pisanje?)
 - ▶ In jo zapremo
 - ▶ In vrnemo `True`
 - ▶ Če pa v varovanem bloku pride do izjeme
 - ▶ Očitno datoteke ni
 - ▶ Vrnemo `False`



Izjema kot orodje

```
def obstaja(ime):  
    try:  
        dat = open(ime)  
        dat.close()  
        return True  
    except:  
        return False
```

- ▶ Funkcija poizkusi (try:) odpreti datoteko ime
- ▶ Če datoteka obstaja, potem datoteko odpre in nato zapre ter vrne True
- ▶ Če datoteka ne obstaja, potem se je zgodila izjema ob poskusu odpiranja,
- ▶ To izjemo smo presegli (except:) in vrnili False, ker datoteka ne obstaja
- ▶ Primeri (v isti mapi kot je datoteka z zgornjo kodo Izjema.py, naj bo dat.txt):
obstaja('dat.txt') -> True
obstaja('izmisljena.txt') -> False



Ulovljena izjema

- ▶ Poglejmo naslednjo kodo

```
imeDatoteke = input("Podaj ime datoteke: ")
try:
    datoteka = open(imeDatoteke)
    print(datoteka.read())
except IOError:
    print("Datoteke", imeDatoteke, "ni!")
print("To je vse.")
```

- ▶ Kaj se sedaj zgodi, če datoteke ni?

```
>>> test()
Podaj ime datoteke: bla.txt
Datoteke bla.txt ni!
To je vse.
>>>
```

In kaj če

```
imeDatoteke = input("Podaj ime datoteke: ")
```

```
try:
```

```
    datoteka = open(imeDatoteke)
```

```
    print(datoteka.read())
```

```
    x = 1 / 0
```

```
except IOError:
```

```
    print("Datoteke", imeDatoteke, "ni!")
```

```
    print("Ali pa smo delili z 0!")
```

```
print("To je vse.")
```

▶ Kaj se sedaj zgodi?

▶ Če datoteka je?

▶ In če je ni?



Kaj pa sedaj?

```
imeDatoteke = input("Podaj ime datoteke: ")
```

```
try:
```

```
    datoteka = open(imeDatoteke)
```

```
    print(datoteka.read())
```

```
    x = 1 / 0
```

```
except:
```

```
    print("Datoteke", imeDatoteke, "ni!")
```

```
    print("Ali pa smo delili z 0!")
```

```
print("To je vse.")
```

► Kaj se sedaj zgodi?

- Če datoteka je?

- In če je ni?



Informacija o napaki

```
imeDatoteke = input("Podaj ime datoteke: ")
try:
    datoteka = open(imeDatoteke)
    print(datoteka.read())
    x = 1 / 0
except Exception as e: # v e je opis napake
    print("Prišlo je do napake", e)
print("To je vse.")
```

- ▶ Kaj se sedaj zgodi?
 - ▶ Če datoteka je?
 - ▶ In če je ni?



Izpiši število vrstic

- ▶ Sestavimo metodo, ki izpiše bodisi:
 - ▶ Datoteka ne obstaja
 - ▶ Datoteka ima 12 vrstic.
- ▶ Ideja:
 - ▶ `readlines` vrne seznam, kjer je posamezni element seznama vrstica datoteke
 - ▶ Izpišemo dolžino tega seznama
- ▶

```
def izpisStVrstic(imeDat) :  
    if os.path.isfile(imeDat) :  
        dat = open(imeDat, 'r')  
        print(imeDat, 'ima',  
              len(dat.readlines()),  
              'vrstic.')
```

```
        dat.close()  
    else :  
        print('Datoteka ne obstaja.')
```



Brez os.path?

- ▶ Uporabimo lovljenje izjem
- ▶

```
def izpisStVrstic(imeDat) :  
    try :  
        dat = open(imeDat, 'r')  
    except :  
        print('Datoteka ne obstaja.')    else :  
        print(imeDat, 'ima',  
              len(dat.readlines()),  
              'vrstic.')        dat.close()
```

