pyladies

Siadamy co drugi rząd!

pyladies

4.15
Obsługa wyjątków

pyladies

Directed by: Jacek Małyszko

21.03.2018

pyladies

WIFI:
PUT-events-WiFi
Login: user_69211
Hasło: 9my3ci6kaZI

CO NAS DZISIAJ CZEKA

1. Przyjrzymy się wyjątkom w Pythonie
2. Dowiemy się, jak je obsłużyć
3. Nauczymy się definiować i wywoływać własne wyjątki
4. W zadaniach przećwiczymy dodatkowo znane już:
    a. Pętelki (for oraz while)
    b. Słowniki i listy
    c. Funkcje
    d. Debugowanie
5. Parę słów o tworzeniu modułów

Kody wykorzystywane podczas zajęć są dostępne pod adresem:

https://goo.gl/Mw7hqM

Przykład - statystyki_treningu.py

Na ile sposobów można zepsuć wykonanie programu na tablicy?

Plik dostępny pod linkiem: https://goo.gl/Mw7hqM

- Jest to program do zapisywania czasów okrążeń zawodników podczas treningu.
- Podajemy ich inicjały i po spacji czas okrążenia.
- Na koniec (po wpisaniu EXIT) wypiszą się statystyki treningu dla poszczególnych zawodników

Na kolejnym slajdzie - przykład poprawnego działania programu.

```
Podaj inicjały i, po spacji, czas okrążenia. Jeśli chcesz wyjść, wpisz "EXIT"
mk 12.8
Podaj inicjały i, po spacji, czas okrążenia. Jeśli chcesz wyjść, wpisz "EXIT"
pk 11.47
Podaj inicjały i, po spacji, czas okrążenia. Jeśli chcesz wyjść, wpisz "EXIT"
dj 9.18
Podaj inicjały i, po spacji, czas okrążenia. Jeśli chcesz wyjść, wpisz "EXIT"
pk 9.15
Podaj inicjały i, po spacji, czas okrążenia. Jeśli chcesz wyjść, wpisz "EXIT"
mk 16.4
Podaj inicjały i, po spacji, czas okrążenia. Jeśli chcesz wyjść, wpisz "EXIT"
EXIT
Podsumowanie:
Zawodnik: Michał Kowalski
    Średni czas okrążenia: 14.6
    Liczba okrążeń: 2
Zawodnik: Paweł Kawa
    Średni czas okrążenia: 10.31
    Liczba okrążeń: 2
Zawodnik: Dawid Jachnik
    Średni czas okrążenia: 9.18
    Liczba okrążeń: 1
Program zakończył się sukcesem. Dzięki!
```

Przykład - różne typy błędów w różnych sytuacjach

ValueError: problem z wartością, nie da się przekonwertować stringa na floata

```
Podaj inicjały i, po spacji, czas okrążenia. Jeśli chcesz wyjść, wpisz "EXIT"
mk 12.8
Podaj inicjały i, po spacji, czas okrążenia. Jeśli chcesz wyjść, wpisz "EXIT"
pk 9,15
Traceback (most recent call last):
  File "/home/jacek/PycharmProjects/zaoczni1/1.py", line 20, in <module>
    zawodnicy_okrazenia[inicjaly].append(float(czas))
ValueError: could not convert string to float: '9,15'
```

Powinno być 12.7 (kropka, nie przecinek).

Przykład - różne typy błędów w różnych sytuacjach

KeyError: brak żądanego klucza w słowniku.

```
Podaj inicjały i, po spacji, czas okrążenia. Jeśli chcesz wyjść, wpisz "EXIT"
MK 12
Traceback (most recent call last):
  File "/home/jacek/PycharmProjects/zaoczni1/1.py", line 20, in <module>
    zawodnicy_okrazenia[inicjaly].append(float(czas))
KeyError: 'MK'
```

Nie ma ‘MK’, tylko ‘mk’ (małymi literami)

Przykład - różne typy błędów w różnych sytuacjach

ValueError: problem z wartością, próbował przypisać wynik split() do dwóch zmiennych, a wyszła tablica jednoelementowa

```
Podaj inicjały i, po spacji, czas okrążenia. Jeśli chcesz wyjść, wpisz "EXIT"
exit
Traceback (most recent call last):
  File "/home/jacek/PycharmProjects/zaoczni1/1.py", line 18, in <module>
    inicjaly, czas = wejscie.split()
ValueError: not enough values to unpack (expected 2, got 1)
```

Jak do tego doszło? Zdebugujmy!

Przykład - różne typy błędów w różnych sytuacjach

Próba dzielenia przez zero we wskazanej linii!

```
Podaj inicjały i, po spacji, czas okrążenia. Jeśli chcesz wyjść, wpisz "EXIT"
EXIT
Traceback (most recent call last):
  File "/home/jacek/PycharmProjects/zaoczni1/1.py", line 27, in <module>
    print('\tŚredni czas okrążenia:', sum(zawodnicy_okrazenia[zaw]) / len(zawodnicy_okrazenia[zaw]))
ZeroDivisionError: division by zero
Podsumowanie:
Zawodnik: Michał Kowalski
```

Jak do tego doszło?

Musieliśmy nie zarejestrować żadnego okrążenia dla mk.
Wtedy len(zawodnicy_okrazenia[‘mk’]) jest zerem, a przez zero dzielić nie można.

Jak obsługiwać wyjątki? try/catch

```
print('Program do dzielenia dwóch liczb. Podaj pierwszą i wciśnij enter')
try:
    a = float(input())
    print('Podaj drugą i wciśnij enter:')
    b = float(input())
    wynik = a / b
    print('Wynik to:', wynik)
except ValueError as ex:
    print('Chyba nie podałeś liczby w cyfrach...')
except ZeroDivisionError as err:
    print('Nie można dzielić przez zero, głuptasku!')
except:
    print('Wystąpił nieoczekiwany błąd!')
```

Try / catch - zasady

- polecenia objęte pakietem try rozpoczną być wykonywane normalnie, linijka po linijce
- jeżeli w czasie ich wykonania nastąpi błąd rodzaju wskazanego przez jedną ze zdefiniowanych klauzul except to:
  - nastąpi przerwanie wykonania programu w linii, która spowodowała wyjątek
  - kontrola zostanie przekazana do właściwego excepta
  - pozostałe polecenia wewnątrz try nie zostaną wykonane
- jeżeli nastąpi błąd w pakiecie except lub powstanie błąd nieobsługiwany przez żaden z except to:
  - nastąpi przerwanie wykonania programu w linii, która spowodowała wyjątek
  - interpreter przekazuje wyjątek do zewnętrznego zasięgu (funkcji wywołującej daną funkcję)
  - przekazywanie postępuje “w górę” aż do złapania wyjątku lub opuszczenia programu
- jeżeli nie nastąpi błąd, bloki except zostaną pominięte

Alternatywa do try/except: niedopuszczenie do powstania błędu

```
print('Program do dzielenia dwóch liczb. Podaj pierwszą i wciśnij enter')
try:
    a = float(input())
    print('Podaj drugą i wciśnij enter:')
    b = float(input())
    if b != 0: # zanim spróbuję podzielić upewniam się, że mogę!
        wynik = a / b
        print('Wynik to:', wynik)
    else:
        print('Nie można dzielić przez zero, głuptasku!')
except ValueError as ex:
    print('Chyba nie podałeś liczby w cyfrach...')
except:
    print('Wystąpił nieoczekiwany błąd!')
```

Ćwiczenie 1

Zmodyfikujmy kod programu do śledzenia treningów tak, żeby był jak najbardziej odporny na nieoczekiwane sytuacje.

Albo, bardziej slangowo:

Zmodyfikujmy kod programu do śledzenia treningów tak, żeby był “idiotoodporny” *(idiot-proof)*

Zgłaszanie wyjątków i definiowanie ich nowych typów

- Wyjątki możemy nie tylko wyłapywać - możemy też je wywoływać za pomocą instrukcji raise
- Możemy też definiować własne typy wyjątków.
- Przykład: Konwerter jednostek.
  - Mogę przekonwertować 100 centymetrów na cale, ale nie mogę przekonwertować 100 centymetrów na stopnie celsjusza
  - Jak użytkownik chce, żebym zrobił taką niemożliwą konwersję to jest to sytuacja wyjątkowa - dobry moment na zgłoszenie wyjątku!

Kod przykładu dostępny pod adresem:
https://goo.gl/Mw7hqM

Ćwiczenie 2

- Do naszego konwertera jednostek dodaj nowy typ błędu - BelowZeroAbsoluteException, gdy podana temperatura jest poniżej zera absolutnego
- Wyrzucaj (raise) ten wyjątek w odpowiednim miejscu
- Przetestuj zdefiniowany wyjątek i go obsłuż za pomocą try/catch

Przy okazji: co to jest to if __name__ == '__main__': ?

- Chodzi o nasz przykład konwertera - coś takiego jest tam na dole pliku
- Fragment kodu pod tym ifem wykona się tylko wówczas, gdy bezpośrednio odpalimy ten plik
- Nie wykona się natomiast, jeśli ten plik jest importowany jako moduł (import konwerter)
- Zobaczmy na tablicy...

Ćwiczenie 3

Zmodyfikujmy kod programu do śledzenia treningów:

- Przy uruchamianiu program prosi o ścieżkę do pliku z listą uczestników
- Wczytujemy plik - nowe nazwisko w każdej linii
- Automatycznie generujemy inicjały i tworzymy odpowiednie słowniki

Możliwe wyjątki przy wczytywaniu:

- Plik pod podaną ścieżką nie istnieje
- Nowy typ błędu RepeatedInitialsException gdy dwóch zawodników ma takie same inicjały

Po każdym błędzie informujemy użytkownika i prosimy o poprawienie pliku/ścieżki. Próbujemy tak długo, aż się uda


QUARTERS ONLY
25¢
1. DROP COIN IN SLOT.
2. PULL HANDLE DOWN ALL THE WAY-LET GO
3. TO CLEAR VISION - TURN RED KNOB.
2414

Q&A

A STOJĄ ZA TYM: