

PRACA Z PLIKAMI, OBSŁUGA BŁĘDÓW, DOSTĘP DO SYSTEMU PLIKÓW

***ĆWICZENIA DO PREZENTACJI
MODUŁ 6***

Altkom Akademia S.A., materiały własne

1 MODUŁY I PAKIETY

ĆWICZENIE 1.1:

Użycie modułów i pakietów

UMIEJĘTNOŚCI:

- Po wykonaniu ćwiczenia zdobędziesz umiejętności:
 - tworzenia pakietów i modułów
 - wykorzystania modułów

CELE I ZADANIA:

- Utwórz klasę reprezentującą adres (ulica, kod pocztowy, miejscowość) i zapisz ją w module o tej samej nazwie
- Utwórz klasę reprezentującą osobę (imię, nazwisko, adres) i zapisz ją w module o tej samej nazwie
- W obu klasach zdefiniuj metody konwersji obiektów na tekst
- Z poziomu osobnego modułu przetestuj działanie obu klas

ALGORYTM WYKONANIA:

- Utwórz w module *adres.py* klasę *Adres*
 - utwórz w niej atrybuty instancynne reprezentujące: ulicę, kod pocztowy i miejscowości
 - zdefiniuj metody umożliwiające reprezentację tekstową instancji
- Utwórz w module *osoba.py* klasę *Osoba*
 - utwórz w niej atrybuty instancynne reprezentujące: imię, nazwisko i adres
 - zdefiniuj metody umożliwiające reprezentację tekstową obiektów
- Oba te moduły umieść w pakiecie *model*
- W module *osoby1.py* znajdującym się w pakiecie *dane* utwórz listę kilku osób
- Przedstaw dane tych osób, wykorzystując ich reprezentacje tekstowe
- Zablokuj możliwość wypisania tych danych, podczas importu tego modułu (*osoby1.py*)
- Wykorzystaj instrukcje importu w różnych wariantach i przetestuj je (w analogiczny sposób) w kolejnych modułach *osoby2.py*, *osoby3.py*, ...

ĆWICZENIE 1.2:**Importy względne****UMIEJĘTNOŚCI:**

- Po wykonaniu ćwiczenia zdobędziesz umiejętności:
 - tworzenia pakietów i modułów
 - wykorzystania importów względnych

CELE I ZADANIA:

- Korzystając z rozwiązania poprzedniego ćwiczenia 1.1, tam, gdzie to możliwe, zastosuj importy względne
- Jakie konsekwencje pociągnęły za sobą te zmiany?

ALGORYTM WYKONANIA:

- Skopiuj rozwiązanie poprzedniego ćwiczenia 1.1
- Tam, gdzie to możliwe, w modułach z pakietu *dane* zamień importy zwykłe na importy względne
- Wykorzystaj fakt, że w tych modułach tworzona jest lista osób
- Utwórz moduły *test1.py*, ..., *test4.py* w pakiecie głównym ćwiczenia
 - zainportuj utworzoną listę i wypisz jej zawartość
- Czy tutaj również można użyć importów względnych?

ĆWICZENIE 1.3:**Pakiety przestrzeni nazw****UMIEJĘTNOŚCI:**

- Po wykonaniu ćwiczenia zdobędziesz umiejętności tworzenia i użycia pakietów przestrzeni nazw

CELE I ZADANIA:

- Korzystając z rozwiązania ćwiczenia 1.1, dokonaj podziału zawartości pakietu *model* i utwórz dwie dystrybucje
- Zamień pakiety zwykłe na pakiety przestrzeni nazw
- Podobnie jak w skopowanym rozwiążaniu odwołaj się do zdefiniowanych klas i utwórz listę kilku osób

ALGORYTM WYKONANIA:

- Wzorując się na rozwiążaniu ćwiczenia 1.1 umieść w dwóch dystrybucjach zawartość pakietu *model* (po jednym module w każdej dystrybucji)
- Zachowaj strukturę pakietów
- Aby były to pakiety przestrzeni nazw, usuń pliki *__init__.py*
- W zwykłym pakiecie *dane* utwórz moduł *osoby.py*, a w nim:
 - zainportuj zawartość pakietu *model*
 - pamiętaj, aby do listy *sys.path* dodać lokalizacje obu dystrybucji
 - utwórz listę kilku osób i wypisz zawartość listy

2 OPERACJE NA PLIKACH

ĆWICZENIE 2.1:

Menedżer kontekstu

UMIEJĘTNOŚCI:

- Po wykonaniu ćwiczenia zdobędziesz umiejętności:
 - definiowania menedżerów kontekstu
 - użycia menedżerów kontekstu

CELE I ZADANIA:

- Utwórz menedżera kontekstu reprezentującego element HTML/XML
- Tworząc znacznik podaj jego nazwę i dowolną liczbę (w tym zerową) atrybutów
- Wykorzystaj powyższą klasę do wypisania poprawnej zawartości prostego dokumentu HTML/XML

ALGORYTM WYKONANIA:

- Utwórz klasę reprezentującą element HTML/XML
- Tworząc instancję elementu trzeba podać:
 - obowiązkowo nazwę elementu
 - opcjonalnie dowolną liczbę atrybutów elementu
- Atrybuty elementu podaj w postaci argumentów nazwanych
- Wewnętrznie informacje o nich przechowaj w postaci słownika
- Utwórz klasę jako menedżer kontekstu (jakie metody specjalne trzeba dodać do klasy?)
- Menedżer kontekstu powinien być odpowiedzialny za:
 - wypisanie znacznika otwierającego przed wypisaniem zawartości elementu
 - wypisanie znacznika końcowego (domykającego element) po wypisaniu zawartości elementu
- Nazwy elementów powinny być wypisane dużymi literami (niezależnie od sposobu ich podania)

- Wykorzystaj stworzoną klasę do wypisania przykładowego dokumentu HTML/XML, np.:

```
<HTML>
<HEAD>
<TITLE>
Simple HTML
</TITLE>
</HEAD>
<BODY color="white" width="80%">
Hello World
</BODY>
</HTML>
```

ĆWICZENIE 2.2:**Metamorfoza – odczyt zawartości plików****UMIEJĘTNOŚCI:**

- Po wykonaniu ćwiczenia zdobędziesz umiejętności:
 - prawidłowej pracy z plikami tekstowymi
 - odczytu danych z plików

CELE I ZADANIA:

- Wykorzystaj podane 2 pliki zawierające tekst piosenki w dwóch wersjach językowych (ang. i pl.)
- Przeczytaj i wypisz zawartość obu plików na dwa sposoby:
 - jeden plik za drugim
 - linie na przemian z jednego i drugiego pliku

ALGORYTM WYKONANIA:

- Umieść w bieżącym pakiecie dostarczone przez instruktora pliki z treścią piosenki
 - Napisz program, który umożliwi przeczytanie zawartości obu plików (jeden za drugim)
 - W rozwiązaniu możesz wykorzystać fakt, że obiekty plików (strumienie) są iterowalne
 - Zwróć uwagę, czy znaki narodowe są poprawnie przedstawione – w plikach zostało użyte kodowanie UTF-8
 - Pamiętaj, aby po zakończeniu pracy z plikami zamknąć otwarte strumienie (obiekty plików są menedżerami kontekstu)
-
- Utwórz drugi program (wzorując się na poprzednim) i wyświetl zawartość obu plików, prezentując linie z obu plików na przemian, tzn. pierwsza linia z pliku 1, pierwsza linia z pliku 2 (czyli jej tłumaczenie), druga linia z pliku 1, druga linia z pliku 2, itd.
 - Zobacz, jak teraz prezentuje się treść...

ĆWICZENIE 2.3:**Łączenie plików****UMIEJĘTNOŚCI:**

- Po wykonaniu ćwiczenia zdobędziesz umiejętność:
 - odczytu danych z plików
 - zapisu danych do pliku

CELE I ZADANIA:

- Skopiuj zawartość 2 plików i zapisz ją w nowym pliku

ALGORYTM WYKONANIA:

- Wykorzystaj przykładowe dane (2 wersje językowe piosenki) z poprzedniego ćwiczenia 2.2
- Przeczytaj zawartość z obu plików i zapisz ją w nowym pliku
- Ponieważ pliki są niewielkie, możesz spróbować przeczytać je jednym poleciением
- Zadbaj o zamknięcie wszystkich strumieni

ĆWICZENIE 2.4:

Utrwalanie i odtwarzanie obiektów

UMIEJĘTNOŚCI:

- Po wykonaniu ćwiczenia zdobędziesz umiejętności:
 - konwersji obiektów na postać tekstową
 - zapisu i odczytu danych z pliku
 - odtwarzania obiektów na podstawie ich reprezentacji tekstowej

CELE I ZADANIA:

- Wykorzystaj kod z ćwiczenia 1.1
- Zainportuj listę osób, a następnie skonwertuj ją na tekst
- Zapisz tę informację w pliku
- Odczytaj tekst z pliku
- Odtwórz instancje osób na podstawie reprezentacji tekstowych
- Wypisz na ekranie dane odtworzonych instancji

ALGORYTM WYKONANIA:

- Wykorzystaj kod z ćwiczenia 1.1 (klasy *Adres* oraz *Osoba*)
- Utwórz moduł *zapis.py*:
 - zainportuj listę osób, a następnie skonwertuj ją na tekst
 - użyj takiej reprezentacji tekstowej instancji, aby było możliwe odtworzenie tych instancji na podstawie ich treści
 - zapisz tę informację w pliku
- Uruchom skrypt i sprawdź, czy plik został utworzony oraz zobacz, jak wygląda jego zawartość
- Utwórz moduł *odczyt.py*
 - odczytaj tekst z pliku
 - odtwórz instancje osób na podstawie reprezentacji tekstowych – użyj funkcji *eval*
- Uruchom skrypt i wypisz na ekranie dane odtworzonych instancji

ĆWICZENIE 2.5:**Dane ustrukturyzowane****UMIEJĘTNOŚCI:**

- Po wykonaniu ćwiczenia zdobędziesz umiejętności:
 - konwersji danych do postaci ustrukturyzowanej
 - zapisu i odczytu wszystkich danych do/z pliku
 - odczytu wybranych danych z pliku np. na podstawie numeru rekordu

CELE I ZADANIA:

- Utwórz listę zawierającą rekordy z danymi
- Każdy rekord powinien być listą zawierającą dane opisujące jakiś obiekt (np. dane opisujące osobę)
- Zapisz dane w pliku binarnym o ustalonej strukturze (każda dana tego samego rodzaju powinna zajmować tyle samo miejsca)
- Dokonaj odczytu wszystkich danych z pliku
- Dokonaj odczytu wybranego rekordu z pliku

ALGORYTM WYKONANIA:

- Utwórz listę zawierającą dane opisujące kilka osób
- Każda osoba powinna być osobną listą
- Każdą osobę charakteryzują: imię, nazwisko, płeć (wartość logiczna) oraz rok urodzenia
- Ustal maksymalny rozmiar dla każdej danej
- Napisz funkcję, która skonwertuje dane pojedynczej osoby (listę) na strukturę reprezentowaną przez obiekt typu *bytes* – p. prezentacja
- Podobnie napisz funkcję, która dokona odwrotnej konwersji – zamieni dane z obiektu *bytes* na listę
- Utwórz funkcję, która zapisze dane osób zebranych w liście w binarnym pliku strukturalnym
- Napisz funkcję, która odczyta dane zapisane w pliku i odtworzy listę osób
- Na koniec napisz funkcję, która pozwoli na odczyt z pliku tylko wybranej osoby (rekordu)
 - ponieważ plik jest strukturalny, to można obliczyć długość każdego rekordu
 - znając numer rekordu i jego długość można obliczyć położenie w pliku danych żądanego rekordu
- Przetestuj działanie utworzonych funkcji

3 WYJĄTKI

ĆWICZENIE 3.1:

Osoby – użycie wyjątków standardowych

UMIEJĘTNOŚCI:

- Po wykonaniu ćwiczenia zdobędziesz umiejętności:
 - użycia wyjątków standardowych
 - obsługi wyjątków

CELE I ZADANIA:

- Wykorzystaj rozwiązanie ćwiczenia 2.2 z projektu *PYTH_UML_PROG_OO*
- Użyj wyjątków standardowych do sygnalizacji sytuacji wyjątkowych
- Dodaj obsługę wyjątków

ALGORYTM WYKONANIA:

- Wykorzystaj rozwiązanie ćwiczenia 2.2 z projektu *PYTH_UML_PROG_OO*
- Użyj mechanizmu wyjątków do zasygnalizowania wystąpienia sytuacji wyjątkowej
- W tym przypadku sytuacją wyjątkową będzie próba ustawienia niedozwolonej wartości atrybutu (np. pustego tekstu imienia lub nazwiska, czy też podanie roku urodzenia z przeszłości)
- Zastanów się, która standardowa klasa wyjątku może być użyta do sygnalizacji powyższych sytuacji
- Wykryj taką sytuację i wyrzuć obiekt wyjątku, aby nie dopuścić do przyjęcia błędnych danych
- Uruchom program i zobacz, co się stanie, gdy spróbujesz podać błędne lub niekompletne dane
- Dodaj obsługę wyjątku
- W jaki sposób można związać obiekt wyjątku z komunikatem?
- Jak można dotrzeć do tego komunikatu?
- Uruchom ponownie program i sprawdź jego działanie
- Co się stanie, jeśli podamy wartość niewłaściwego typu (np. rok urodzenia podamy słownie)?
- Dodaj obsługę nowego wyjątku i uruchom ponownie program

ĆWICZENIE 3.2:**Osoby – własne klasy wyjątków****UMIEJĘTNOŚCI:**

- Po wykonaniu ćwiczenia zdobędziesz umiejętności:
 - definiowania własnych klas wyjątków
 - wyrzucania wyjątków i ich obsługi

CELE I ZADANIA:

- Wykorzystaj rozwiązanie poprzedniego ćwiczenia 3.1
- Zastąp standardowe klasy sygnalizujące błędne wartości – własnymi klasami

ALGORYTM WYKONANIA:

- Zmodyfikuj rozwiązanie poprzedniego ćwiczenia 3.1
- Utwórz dedykowane klasy wyjątków o nazwach *NoValueError* oraz *BirthDateError*
- Klasy powinny mieć wbudowany komunikat błędu, informujący o niewłaściwej wartości
- Nazwę błędnego atrybutu (w przypadku klasy *NoValueError*) należy podać podczas tworzenia instancji wyjątku
- Zastąp standardowe klasy wyjątków własnymi i sprawdź działanie programu

ĆWICZENIE 3.3:**Telefon na kartę – sytuacje wyjątkowe****UMIEJĘTNOŚCI:**

- Po wykonaniu ćwiczenia zdobędziesz umiejętności:
 - definiowania i wykorzystania w sposób praktyczny własnych klas wyjątków

CELE I ZADANIA:

- Napisz program symulujący działanie telefonu na kartę
- Wykorzystaj własną klasę wyjątku do sygnalizacji sytuacji niedozwolonych

ALGORYTM WYKONANIA:

- Napisz program symulujący działanie telefonu na kartę
 - W tym celu utwórz klasę o nazwie *PrepaidPhone*
 - Klasa powinna przechować informację o ilości dostępnych minut na rozmowy
 - Ta wartość powinna zostać zainicjowana w czasie tworzenia obiektu (starter)
 - Dodaj metody umożliwiające:
 - *get_limit* – sprawdzenie aktualnego stanu minut
 - *add_to_limit* – doładowanie konta zadaną liczbą minut
 - Dodaj metodę *call* symulującą rozmowę trwającą przez zadany czas
 - W wyniku wywołania tej metody stan konta powinien zostać pomniejszony
 - Przetestuj działanie programu wywołując kilkukrotnie metodę rozmowy i metodę doładowania
 - Co się stanie, gdy czasy trwania rozmów będą przewyższały wartości doładowań i limit minut się wyczerpie?
-
- Utwórz klasę wyjątku *PrepaidPhoneError* sygnalizującą taką sytuację
 - W metodzie rozmowy wykryj sytuację wyczerpania limitu minut i zgłoś ten fakt, poprzez wyrzucenie obiektu wyjątku
 - Wykryj w programie wystąpienie wyjątku, a w jego obsłudze doładuj konto i wypisz aktualny stan minut
 - Przetestuj działanie programu