

TESTY

Uruchomienie

Linia poleceń:

Zainstaluj Pytest. <https://docs.pytest.org/en/latest/getting-started.html>

Wpisz polecenie: `pytest plik`

w przypadku niepowodzenia spróbuj: `python -m pytest plik`

plik musi mieć rozszerzenie: `*.py`

Pycharm:

Zainstaluj Pytest.

Konfiguracja interpretera uruchamiającego testy

File → Settings → Tools → Python Integrated Tools → W Default test runner

wybierz pytest → OK.

Uruchamianie testu

Kliknij prawym przyciskiem na skrypt → Run 'nazwa'.

Tworzenie testu

```
def funkcja():  
    # instrukcje do wykonania  
    assert warunek1  
    # instrukcje do wykonania  
    assert warunek2  
    # ...  
    assert warunekN  
    # ...
```

funkcja - musi się zaczynać od przedrostka test.

warunek - test będzie spełniony jeżeli warunek będzie prawdziwy.

Zadanie 1 (+). Napisz skrypt, który będzie zawierał małą bazę danych przechowującą dane osób (imię, nazwisko, adres), gdzie adres jest słownikiem, których klucze ustal wg własnego uznania. Dane powinny być przechowywane w liście zawierające słowniki. Zdefiniuj funkcje, które umożliwią wykonywać operację CRUDS:

funkcja, która umożliwi(a) dodanie nowej osoby (CREATE). Funkcja jako argument powinna przyjąć dane, które powinny być dodane oraz zwrócić listę zawierającą nowo dodaną daną.

funkcja/funkcje, która zwróci(ca) informację na temat określonych danych (READ). Funkcja jako argument powinna przyjąć jakieś dane i zwrócić jakąś informację odnośnie pobranych danych. Przykładowo zwracamy informację gdzie mieszka osoba, której imię i nazwisko zostało podane jako argument. Możesz stworzyć kilka funkcji zwracających informację wedle własnego uznania.

funkcja/funkcje, która umożliwi(a) zmianę danych (UPDATE). Funkcja jako argument powinna przyjąć dane na podstawie, których chcemy zmienić wartości i wartości, na które mają być one zmienione oraz zwraca listę z odpowiednio zawierającą wszystkie dane z odpowiednimi modyfikacjami. Przykładowo zmiana wieku (podajemy wiek jako argument) osób o podanym imieniu i nazwisku. Możesz stworzyć kilka funkcji zwracających informację wedle własnego uznania.

funkcja/funkcje, która umożliwi(a) usunięcie istniejącej osoby (DELETE). Funkcja jako argument powinna przyjąć informację na podstawie, których chcemy usunąć dane oraz zwrócić nową listę nie zawierającą odpowiednich danych. Przykładowo usunięcie osób, które mają na nazwisko

Kowalski. Możesz stworzyć kilka funkcji zwracających informację wedle własnego uznania.

funkcje wyszukiwania wyszukiwania (SEARCH), które wyszukują dane spełniające określone, kryteria oraz zwracają listę przechowującą słowniki, które je spełniają np.:

operacja znajdowania osób, które mieszkają w podanej miejscowości.

operacja znajdowania osób, które są starsze/młodsze niż podany wiek.

Wszystkie funkcje mogą też przyjąć aktualną bazę danych jako argument i inne dodatkowe rzeczy jeśli uznasz to za konieczne. Przetestuj działanie utworzonych funkcji.

Wejście: Zmienna lista przechowująca listę zawierająca dane słownikowe, które będą przechowywać informacje o osobach.

Wyjście: Na ekranie pojawia się odpowiednia informacja odnośnie osób.

Warunki poprawności zadania: Powinny zostać wykonane odpowiednie testy wywołań funkcji. Upewnij się, że zdefiniowane funkcje działają prawidłowo, tzn. prawidłowo działa dodawanie, czytanie, zmienianie i usuwanie danych o osobach oraz inne operacje.

Zadanie 2. Napisz skrypt, który będzie zawierał małą bazę danych przechowującą dane uczniów wraz z ich ocenami z poszczególnych przedmiotów. Dane powinny być przechowywane w liście zawierającej słowniki. Klucze ustal wg własnego uznania. Zdefiniuj funkcje, które umożliwią wykonywać operację CRUDS:

funkcja, która umożliwi(a) dodanie nowego ucznia (CREATE). Funkcja jako argument powinna przyjąć dane, które powinny być dodane oraz zwrócić listę zawierającą nowo dodane dane.

funkcja/funkcje, która zwróci(ca) informację na temat określonych danych (READ). Funkcja jako argument powinna przyjąć jakieś dane i zwrócić jakąś informację odnośnie pobranych danych. Przykładowo zwracamy informację jakie oceny ma uczeń, którego numer PESEL został podany jako argument. Możesz stworzyć kilka funkcji zwracających informację wedle własnego uznania.

funkcja/funkcje, która umożliwi(a) zmianę danych (UPDATE). Funkcja jako argument powinna przyjąć dane na podstawie, których chcemy zmienić wartości i wartości, na które mają być one zmienione oraz zwraca listę z odpowiednio zawierającą wszystkie dane z odpowiednimi modyfikacjami. Przykładowo zmiana ocen (podajemy oceny jako argument) uczniów o podanym numerze PESEL. Możesz stworzyć kilka funkcji zwracających informację wedle własnego uznania.

funkcja/funkcje, która umożliwi(a) usunięcie istniejącego ucznia (DELETE). Funkcja jako argument powinna przyjąć informację na podstawie, których chcemy usunąć dane oraz zwrócić nową listę nie zawierającą odpowiednich danych. Przykładowo usunięcie uczniów, którzy mają na nazwisko Kowalski.

funkcje wyszukiwania wyszukiwania (SEARCH), które wyszukują dane spełniające określone, kryteria oraz zwracają listę przechowującą dane (słowniki), które je spełniają, np.:

operacja znajdowania uczniów, którzy posiadają średnią arytmetyczną ocen od podanej.

operacja znajdowania przedmiotu z którego średnia arytmetyczna jest największa.

Wszystkie funkcje mogą też przyjąć aktualną bazę danych jako argument i inne dodatkowe rzeczy jeśli uznasz to za konieczne. Przetestuj działanie utworzonych funkcji.

Wejście: Zmienna lista przechowująca listę zawierająca dane słownikowe, które będą przechowywać odpowiednie dane.

Wyjście: Na ekranie pojawia się odpowiedni wynik.

Warunki poprawności zadania: Powinny zostać wykonane odpowiednie testy wywołań funkcji. Upewnij się, że zdefiniowane funkcje działają prawidłowo, tzn. prawidłowo działa dodawanie, czytanie, zmienianie i usuwanie danych o osobach oraz inne operacje.