

Przykładowe kolokwium - Zestaw #1

Zadania mogą być wykonane jako skrypt lub notatnik jupytera. Rozwiązania należy umieścić:

- opcja 1: (jako kody) w prywatnym repozytorium na Githubie
- opcja 2: spakować jako zip i umieścić na pendrive.

Dot zad. 1-3: Każdy obiekt należy stworzyć co najmniej jeden raz i każdą metodę/funkcję należy wywołać co najmniej jeden raz (o ile nie napisano inaczej). (8pkt)

1. (12pkt) Stwórz klasę `Travel` a w niej następujące składniki:

- dodaj zmienne instancyjne: `goal (str)`, `price (float)`, `travel date` (dowolny typ przechowujący datę z modułu `datetime`).
- w inicjatorze nadaj wartości zmiennym z podanych odpowiednio argumentów
- dodaj odpowiednią metodę magiczną zwracając napis z reprezentacją obiektu
- dodaj odpowiednią metodę magiczną umożliwiającą sortowanie obiektów typu `Travel` wg klucza: najpierw sortowanie odbywa się wg ceny - od najtańszej do najdroższej, przy równości sortowanie odbywa się wg daty - od najstarszej do najnowszej.
- dodaj odpowiednią metodę magiczną odpowiadającą za sprawdzanie równości obiektów tak, aby obiekty były równe, gdy jedynie zmienne `goal` i `price` są takie same.

2. (6pkt) Stwórz klasę `TrainJourney` dziedziczącą po `Travel` ze składowymi:

- dodaj zmienne instancyjne `train_class (int)` oraz `seat (str)`
- rozszerz `__init__` by ustawił dodatkowe dwa pola z argumentów
- dodaj odpowiednią metodę magiczną zwracając napis z reprezentacją obiektu z uwzględnieniem zmiennych z klasy pochodnej

3. (6pkt) Stwórz klasę `AirTravel` dziedziczącą po `Travel` ze składowymi:

- dodaj zmienne instancyjne `airline (str)` oraz `seat (str)`
- rozszerz `__init__` by ustawił dodatkowe dwa pola z argumentów
- dodaj odpowiednią metodę magiczną zwracając napis z reprezentacją obiektu z uwzględnieniem zmiennych z klasy pochodnej

4. (8pkt) Stwórz listę, w której będą po 3 obiekty typu `Travel`, `AirTravel` oraz `TrainJourney`. Następnie posortuj listę. Upewnij się, że "printując" listę wyświetlana jest reprezentacja obiektu pochodząca z odpowiedniej metody magicznej zdefiniowanej bezpośrednio we właściwej klasie.

5. (10pkt) Stwórz klasę `Person` ze zmiennym `name (str)` oraz `age (int)`. Dodaj inicjalizator ustawiający zmienne z argumentów. Następnie zaimplementuj odpowiednią metodę magiczną by uzyskać mechanizm płytkiej kopii (wnętrze dowolne). Następnie dodaj w klasie statyczną metodę z dwoma argumentami, która zwraca obiekt typu `Person` (argumenty tej metody ustawiają zmienne obiektu). Zaimplementuj odpowiednią metodę magiczną, która odpowiada za sprawdzanie nierówności tak, aby dwa obiekty nie były sobie równe wtedy, gdy mają różne wartości pola `age`.

Poza klasą za pomocą metody statycznej stwórz dwa obiekty. Następnie wyświetl wynik sprawdzenia tego, czy obiekty nie są sobie równe.