

Wizualizacja danych (2024S) - Egzamin (przykładowy) - Zestaw R02

Zasady egzaminu:

- *Zadanie 1 i 2 po 15 pkt. Zadanie 3 - 20 pkt.*
- *Punktacja: 46-50 pkt - bdb(5,0); 41-45 pkt - db+(4,5); 36-40 pkt - db(4,0); 31-35 pkt - dst+(3,5); 26-30 pkt - dst(3,0); 0-25 pkt - ndst (2,0).*
- Obowiązuje regulamin zajęć.
- Czas: 75 minut.
- **Egzamin należy wykonać na komputerach zamontowanych na stałe w pracowniach.**
- Student przysyłając rozwiązania oświadcza, że rozwiązał je samodzielnie.
- W trakcie egzaminu nie można korzystać z żadnych materiałów pomocniczych w żadnej formie poza udostępnionym pdf z części wykładów. Na pendrive znajduje się również plik `requirements.txt` z rekomendowanymi wersjami bibliotek. Wszelkie kody powinny być napisane manualnie bez wspomagania się dodatkami automatycznie generującymi kod (np. Copilot, chat GPT itp.).
- Publikowanie poleceń i rozwiązań w internecie jest zabronione do czasu napisania egzaminu przez wszystkie osoby.
- Kod musi się interpretować w całości bez błędów, aby był sprawdzany. Ostrzeżenia (tzw. warningi) są dopuszczalne, o ile nie prowadzi to do błędów merytorycznych.
- Kod zakomentowany nie będzie sprawdzany.
- W trakcie egzaminu zostanie udostępniony przez prowadzącego pendrive. Zawartość pendrive będzie zawierać pliki pomocnicze do poleceń. Ten sam pendrive służy do zgrania rozwiązań. Umieszczenie rozwiązań na pendrive powinno odbyć się w czasie kolokwium.
- Rozwiązania po czasie mogą nie być sprawdzane.
- W rozwiązaniach należy przestrzegać stylu PEP8 i dobrych praktyk. W przypadku gdy zaburzenie zaleceń spowoduje niejednoznaczność interpretacji kodu, za dane polecenie mogą być obniżone punkty (nawet do zera).
- Wszelkie wykonane czynności prowadzące do wykonania wykresu (wizualizacji danych), muszą być wykonane w kodzie. Jeśli do danego zadania został dołączony plik (xlsx lub csv), to modyfikacja plików poza kodem w języku Python udostępnionym jako rozwiązane może oznaczać zero punktów za dane polecenie.
- Po kartkach z poleceniami można pisać i traktować jako brudnopis.
- Format rozwiązania:
 - Zadania powinny być umieszczone w archiwum .zip na udostępnionym pendrive.
 - Nazwa archiwum powinna być wg schematu NUMERZESTAWU_NUMERALBUMU.zip gdzie numer zestawu znajduje się na górze kartki z poleceniami. np. A23_123456.zip
 - We wnętrzu archiwum powinny znajdować się tylko same kody w języku Python, pliki powinny posiadać dokładnie nazwy (z uwzględnieniem wielkości znaków): `zad1.py`, `zad2.py`, `zad3.py`.
 - Minimalna wersja Python to 3.10 (zalecana to 3.12). Wykresy powinny bazować na bibliotece matplotlib lub seaborn lub innych bibliotekach bazujących na nich.
 - Maksymalna waga archiwum 10 MB.
 - Należy nie dołączać venv.
 - Archiwum powinno być bez hasła.
 - W przypadku pominięcia danego zadania, należy dodać plik o nazwie sprecyzowanej wyżej (zawartość może być pusta).

Polecenia są na odwrocie.

Zad.1. Odwzoruj wykres znajdujący się w pliku o nazwie w02.png. Odcienie kolorów mogą się różnić, jednak główne barwy muszą być zachowane. Zapisz wykres w formacie svg za pomocą kodu.

Zad.2. W jednym pliku wykonaj poniższe czynności:

- załaduj dane z pliku ceny02.xlsx jako ramkę danych (Data Frame),
- stwórz wykres liniowy prezentujący dane zawarte w ramce danych (tylko za jabłka i cytryny), każda linia to inny produkt, styl i szerokość każdej linii powinny być inne niż standardowe
- dodaj do wykresu strzałkę, której grot skierowany jest na najwyższy punkt na wykresie

Wykres powinien być estetyczny i podpisany. Im więcej - tym lepiej.

Zapisz wykres w formacie webp za pomocą kodu.

Zad.3. W jednym pliku wykonaj poniższe czynności:

- załaduj dane z pliku samochody02.xlsx,
- o ile to potrzebne, przekształć dane tak, aby dane liczbowe były zgodne z koncepcją “czystych danych”
- stwórz wykres słupkowy poziomy pogrupowany na podstawie tych danych
- Wykres powinien być estetyczny i podpisany. Im więcej - tym lepiej.

Zapisz wykres w formacie png za pomocą kodu.

