

Zadanie 3 - Program do Obliczania Różnic Kursowych

Wstęp

W ramach tego zadania, będziecie tworzyć program w języku Python, który ma na celu pomoc w obliczaniu różnic kursowych dla faktur od zagranicznych kontrahentów. Projekt ten będziecie realizować w oparciu o metodykę Agile, pracując na podstawie określonych historyjek (user stories), dostępnych na platformie Trello.

Link do Trello: [User stories](#)

Cel Programu

Program ma za zadanie:

- Przyjmować dane dotyczące faktur (kwota, waluta, data wystawienia) oraz informacje o płatnościach (kwota, waluta, data płatności).
- Pobierać z API Narodowego Banku Polskiego (NBP) aktualne kursy walut i na ich podstawie obliczać różnice kursowe.
- Zapewniać zapis wprowadzonych danych oraz wyników obliczeń do plików.
- Działać w trybie interaktywnym oraz wsadowym.
- Zapewniać dokumentację opisującą działanie programu i sposób jego użycia.
- Obliczanie Różnic Kursowych

Różnice kursowe oblicza się, porównując kurs waluty w dniu wystawienia faktury i w dniu jej zapłaty. Kurs waluty należy pobrać z API NBP z odpowiedniej daty. Pamiętaj, że API może zwracać błędy lub nie mieć danych dla niektórych dni (np. weekendy, święta), więc konieczna jest obsługa takich przypadków.

Korzystanie z API NBP

Do pobrania kursów walut użyjemy publicznie dostępnego API NBP. Aby uzyskać kurs z danego dnia, należy wykonać zapytanie do API z odpowiednią datą. Pamiętaj, że API może zwracać błędy lub nie mieć danych dla niektórych dni (np. weekendy, święta), więc konieczna jest obsługa takich przypadków.

Użycie Systemu Kontroli Wersji

Zachęcamy do używania systemu kontroli wersji Git oraz hostingu takiego jak GitHub, Bitbucket czy GitLab. Każda historyjka powinna być realizowana w ramach co najmniej jednego commitu. Ważne, aby commity były właściwie opisane, odzwierciedlając zmiany wprowadzane w danym momencie pracy nad projektem. Regularne commitowanie zmian oraz pushowanie kodu do zdalnego repozytorium nie tylko jest dobrą praktyką, ale także pozwala na śledzenie postępu w pracy nad projektem.

Standardy Kodowania

Oczekujemy, że kod będzie:

- Czytelny i logicznie strukturyzowany.
- Posiadał właściwie i logicznie nazwane zmienne i funkcje.
- Zawierał komentarze w miejscach, gdzie są one potrzebne do wyjaśnienia działania kodu.

- Spełniał ogólne standardy jakościowe kodu Pythonowego, zgodnie z PEP 8.

Raport Projektowy

Na koniec projektu, prosimy o przygotowanie krótkiego, zwięzłego raportu, który opíše proces tworzenia programu, napotkane wyzwania, sposób ich rozwiązania oraz wnioski z realizacji projektu. Raport powinien być napisany jasno i zrozumiale.

Podsumowanie

To zadanie pozwoli Wam na praktyczne zastosowanie umiejętności programistycznych oraz zrozumienie, jak praca nad prawdziwym projektem IT może wyglądać. Pamiętajcie, że kluczowe jest tutaj nie tylko techniczne wykonanie zadania, ale także umiejętność pracy zgodnie z określonymi wymaganiami, organizacja pracy oraz dokumentowanie postępów. Powodzenia!