

Bazy Danych

Projekt

Użytkownicy:

(1) Owner (Administrator):

- (a) Istnieje tylko jeden, zakładany przy tworzeniu bazy
- (b) Może dodawać użytkowników typu Worker jak i ich usuwać
- (c) Może on:
 - (i) Wypłacać pensje dla użytkowników typu Worker(wykorzystanie transakcji)
 - (ii) Może sprawdzić historie zamówień poszczególnych użytkowników typu Client
 - (iii) Może sprawdzić aktualny bilans poprzez odpowiednią procedurę
 - (iv) Przy użyciu odpowiedniej procedury może tworzyć jak i usuwać promocję na produkty
 - (v) Może on usuwać klientów
 - (vi) Oraz wykonywać wszystkie operacje jakie może i użytkownik typu Worker

(2) Worker (Pracownik):

- (a) Istnieje ich wielu, są tworzeni przez użytkownika typu Owner
- (b) Nie może on tworzyć nowych tabel
- (c) Może on:
 - (i) Korzystać z procedury złożenia zamówienia lub jego anulowania poprzez odpowiednia procedurę
 - (ii) Może potwierdzić wysłanie złożonego zamówienia(transakcji) przez Client'a, poprzez wywołanie odpowiedniej procedury
 - (iii) Może odebrać zamówienie do magazynu, poprzez odpowiednia procedurę
 - (iv) Może dodać lub usunąć produkt poprzez odpowiednią procedurę
 - (v) Ma dostęp do sprawdzenia stanu magazynowego jak i podejrzenia produktów
 - (vi) Może anulować złożone przez klienta zamówienie

(3) Client (Klient):

- (a) Istnieje ich wielu, każdy może założyć sobie konto Client'a jak i je zamknąć kiedy jego bilans do zapłaty wynosi 0
- (b) Client może:

- (i) Składać i anulować zamówienie(transakcje) poprzez odpowiednia procedurę
- (ii) Może przeglądać wszystkie produkty dostępne na magazynie
- (iii) Może sprawdzać aktualne promocje
- (iv) Może opłacić zamówienie(transakcję)

Procedury:

1. CreateTransaction() – Wywoływana przez klienta, tworzy odpowiednie zamówienie kiedy jest wystarczająco produktów na magazynie, używa systemu transakcji
2. DeleteTransaction() – Anuluje transakcje klienta jeśli to możliwe
3. CreateDeliver() – tworzy odpowiednie zamówienie.
4. ReceiveDelivery() – odbiera odpowiednie zamówienie i aktualizuje stan wybranego magazynu jeśli jest to możliwe. Także wykorzystuje transakcje.
5. CancelDelivery() – Anuluje wcześniej złożone zamówienie
6. PayTransaction() – procedura w której klient może opłacić transakcję, dodaje nowy wpis do tabeli Balance i zmienia status Transaction
7. PaySalary() – Wypłaca wynagrodzenie jeśli to możliwe, używa systemu transakcji.
8. PayFee() – Pozwala opłacić wszystkie koszty jeśli to możliwe, używa systemu transakcji.

Funkcję:

1. GetTotal() – zwraca całą wartość zamówienie z uwzględnieniem rabatów.
2. CalculateBallance() – Zwraca aktualny bilans wszystkich operacji finansowych
3. StorageUsed() – zwraca ogólne zapełnienie magazynów.

Triggery:

1. Trigger ogólny na bazę danych, który każdą operacje zapisuję w tabeli Log
2. Trigger przy aktualizowaniu, usuwaniu i dodawaniu rekordów w tabeli Products, zmienia posiadana liczbę produktów i zapełnienia magazynów.
3. Trigger na tabelę Transactions, dodający wpisy do tabeli History oraz Balance, dodatkowo modyfikujący saldo Client'a
4. Trigger na tabelę Delivery dodający rekordy to tabeli Balance oraz modyfikujący rekordy w tabeli Products
5. Trigger na tabelę Sales zmieniający ceny w tabeli Products.
6. Trigger na tabele products, kiedy skończy się data ważności usuwa produkt i dodają do Balance nowe koszty.