

Task 1 - Робота з розподіленими (двофазними) транзакціями

Для роботи з розподіленими (двофазними) транзакціями пропонується використовувати СКБД PostgreSQL (<http://www.postgresql.org/download>).

Для зручності роботи з СКБД PostgreSQL можна використовувати pgAdmin (<http://www.pgadmin.org/>) або аналогічний графічний додаток.

Для виконання розподіленої транзакції Вам необхідно реалізувати свій Менеджер (координатор) розподілених транзакцій (TM). TM відправлятиме СКБД команди SQL (або використовувати API для 2PC з якоїсь з бібліотек) за допомогою яких і буде здійснюватись керування розподіленої транзакцією, на основі протоколу двофазної фіксації ([Two-PhaseCommit](#) Protocol).

TM представляє з себе просте консольне застосування чи скрипт (можливо навіть без взаємодії з користувачем), який може бути реалізований на будь-якій мові програмування. В застосуванні необхідно виконати набір SQL-запитів (або виклик ряду методів API) до СКБД PostgreSQL.

Процедуру підключення до СКБД PostgreSQL і як виконувати SQL-запити з відповідної мови, читайте в документації або літературі до вибраної мови програмування.

Для імітації розподіленої системи, в PostgreSQL необхідно створити дві окремі БД (схеми). Це можна зробити в рамках одного сервера PostgreSQL.

Одна БД буде містити дані (таблицю) квитків на літак, друга - бронювання готелю.

DB1: Fly Booking

Booking ID	Client Name	Fly Number	From	To	Date
XXX	Nik	KLM 1382	KBP	AMS	01/05/2015

DB2: Hotel Booking

Booking ID	Client Name	Hotel Name	Arrival	Departure
YYY	Nik	Hilton	01/05/2015	07/05/2015

У TM Вам необхідно реалізувати функціональність, яка заносить інформацію про броні на літак і готель для певного клієнта, використовуючи протокол двофазної фіксації.

Додайте третю **DB3: Account**

DB3: Account

Account ID	Client Name	Ammount
------------	-------------	---------

YYY	Nik	200
-----	-----	-----

На поле amount має бути накладено обмеження (Constraint) щоб воно не могло бути меншим 0.

Модифікуйте код вашого застосування таким чином, щоб під час бронювання літака та готелю виконувалось списання певної суми з рахунку клієнта (без перевірок). Якщо у результаті списання спрацює Constraint, то вся транзакція має відмінатись.

Запускаючи застосування декілька разів перевірте, що створюються нові записи, а amount зменшується.

Коли буде недостатньо грошей, має спрацювати rollback.

Фейл-сценарій

Для якоїсь з під-транзакцій закоментуйте COMMIT PREPARED у коді.

Після цього спробуйте запустити застосування декілька разів.

У результаті застосування має зависнути через лок транзакції на таблицю для якої не було виконано COMMIT PREPARED чи ROLLBACK PREPARED, і таким чином транзакція лишається у стані WAIT.

Додаткове завдання* (advanced - +7 балів)

Реалізувати додатково до 2PC-протоколу, Saga pattern одним із способів з лекції.

Вимогу до оформлення протоколу:

Завдання здається особисто без протоколу, або надсилається протокол який має містити:

- sql для створення БД та відповідних обмежень в них
- код застосування (може бути лінка на репозиторій)
- скріншот з системної таблиці з записом про заблоковану транзакцію

Додаткова інформація:

Нижче наводиться приклад SQL-запитів для виконання двухфазной транзакції.

1. Here's the sequence that two-phase commits:

```
BEGIN;
update mytable set a_col = 'something' where red_id = 1000;
PREPARE TRANSACTION 'foobar';
COMMIT PREPARED 'foobar';
```

2. Here's the sequence that rolls back, leaving the table unchanged:

```
BEGIN;  
update mytable set a_col = 'something' where red_id = 1000;  
PREPARE TRANSACTION 'foobar';  
ROLLBACK PREPARED 'foobar';
```

1. Коротка інструкція з встановлення та налаштування PostgreSQL
https://github.com/jboss-developer/jboss-developer-shared-resources/blob/master/guides/CONFIGURE_POSTGRES_SQL.md
2. Інформація по командам PostgreSQL для виконання двохфазної транзакції
<https://www.postgresql.org/docs/current/static/sql-prepare-transaction.html>
3. У разі виникнення блокування через невдачу транзакції (не були виконані команди [COMMIT PREPARED](#) або [ROLLBACK PREPARED](#)), їх можна побачити виконавши SQL-запит:

```
select * from pg_prepared_xacts;
```

з pgAdmin.
Щоб зняти блокування необхідно з pgAdmin виконати SQL-запит

```
ROLLBACK PREPARED '<gid>';
```

або

```
COMMIT PREPARED '<gid>';
```
4. У разі виникнення повідомлення виду
*XAException occurred. XAException contents and details are: The cause is org.postgresql.util.PSQLException: ERROR: prepared transactions are disabled
Error code is: XAER_RMERR (-3). Exception is: Error preparing transaction
Hint: Set max_prepared_transactions to a nonzero value.*
змінюйте файл postgresql.conf (для Windows C: \ Program Files \ PostgreSQL \ 9.x \ data \ postgresql.conf) встановивши можливість для виконання двофазних транзакцій:

```
max_prepared_transactions = 5
```