

## TRANSAKSI

Transaksi adalah sebuah unit dari eksekusi program yang mampu mengakses dan mengupdate berbagai data yang memiliki kaitan logika transaksi itu sendiri dimana dalam menjaga konsistensi data secara terintegrasi dipertahankan.

Sebuah transaksi harus memiliki sifat –ACID : Atomik, Consisten, Isolasi, Durability.

## ACID

### 1. ATOMICITY

- Dimana semua operasi dalam transaksi harus bekerja secara utuh/total atau tidak bekerja sama sekali (artinya pekerjaan transaksi tidak boleh dikerjakan sebagian).
- Atomicity men-isyaratkan bahwa setiap transaksi adalah "semua atau tidak": jika salah satu bagian dari transaksi gagal, seluruh transaksi gagal, dan database yang tersisa tidak berubah. Sebuah sistem atom harus menjamin atomicity dalam setiap situasi, termasuk gangguan listrik, kesalahan, dan crash. Ke dunia luar, transaksi berkomitmen muncul (oleh dampaknya pada database) tidak dapat dibagi ("atom"), dan transaksi dibatalkan tidak terjadi.

Contoh Kasus

Contoh (Gift Diamond Game Online sebesar 1000 dari Player A kepada Player B):

- Maka stepnya adalah sebagai berikut:
  - 1. Cek saldo Player A apakah mencukupi, jika mencukupi maka lanjut ke step ke-2. Jika tidak maka transaksi selesai dan tidak ada perubahan.
  - 2. Kurangi diamond Player A sebesar 1000 diamond
  - 3. Kirim Diamond ke User B
  - 4. Player B menerima Diamond sebesar 1000 diamond
  - Nah seluruh step diatas harus digabungkan sehingga apabila setelah step ke-2 terjadi kesalahan maka seluruh perubahan transaksi harus di batalkan. Setelah transaksi berhasil maka akan di lakukan *commit*.

## 2.Consistency

- Dampak eksekusi dari sebuah transaksi harus menjamin keadaan data yg konsisten.
- konsistensi memastikan bahwa transaksi akan membawa database dari satu ke yang lain yang valid. Setiap data ditulis ke database harus berlaku sesuai dengan semua aturan yang ditetapkan.

### Contoh

Pada Kasus sebelumnya di kasus Gift Diamond jumlah Diaomnd yang mau dikirim adalah sejumlah 1000 maka player 1 akan kehilangan 1000 diamond untuk penarikan dan jumlahnya harus tepat dan juga Player 2 akan menerima 1000 diamond pas tanpa dikurangi biaya apapun sesuai aturan yang berlaku pada masing masing aplikasi, Jadi data yang diproses harus keadaan data yang konsisten dan tidak bias berubah saat proses transaksi

## 3.Isolation

- Pada sejumlah Transaksi yang terjadi secara bersamaan , setiap transaksi tidak boleh terpengaruh transaksi lainnya yang juga sedang berjalan walaupun berhubungan atau menuju pada database yang sama. Hasil transaksi sementara harus terjaga dan terlindungi dari eksekusi-eksekusi yang lain.
  - isolasi memastikan bahwa pelaksanaan bersamaan dari hasil transaksi dalam keadaan sistem yang akan diperoleh jika transaksi dieksekusi secara serial, yaitu satu demi satu. Memberikan isolasi adalah tujuan utama dari kontrol konkurensi . Tergantung pada metode kontrol konkurensi, efek dari transaksi yang tidak lengkap bahkan mungkin tidak terlihat oleh transaksi lain.
- 
- Misal ada 2 transaksi transfer antara rek A, dan rek B yang terjadi secara bersamaan:
  - T1 : Transfer A->B senilai 500
  - Kurangi saldo A 500
  - Tambah saldo B 500
  - T2 : Transfer B->A senilai 1000
  - Kurangi saldo B 1000
  - Tambah saldo A 1000

- Nah bayangkan jika transaksi tersebut terjadi secara bersamaan dan di proses tanpa isolasi sehingga akan menjadi:
- Kurangi saldo A 500
- Tambah saldo B 500
- Kurangi saldo B 1000
- Tambah saldo A 1000
- Jika terjadi kegagalan pada T1, maka data akan tidak konsisten sebab pada T2 sudah mengganti menambahkan saldo pada B. Demikian pentingnya isolasi, sehingga data tidak akan bisa di akses secara bersamaan oleh lebih dari satu operasi.

#### 4. Durability

- Setelah terjadinya transaksi maka akan diikuti update atau perubahan data pada database, maka perubahan tersebut harus tetap bertahan dan dianggap paling valid dan konsisten.
- Daya tahan berarti bahwa sekali transaksi telah dilakukan, itu akan tetap demikian, bahkan dalam hal kehilangan kekuasaan, crash , atau kesalahan. Dalam database relasional, misalnya, pernah sekelompok pernyataan SQL mengeksekusi, hasilnya harus disimpan secara permanen (bahkan jika database crash segera setelahnya). Untuk mempertahankan melawan daya yang hilang, transaksi (atau efek mereka) harus dicatat dalam memori non-volatile .

#### Contoh kasus

Pada Pengiriman Diamond Game Online yang dimana Player A mengirim diamond sejumlah 1000 Diamond yang dimana total Diamond yang dimiliki Player A akan berkurang sejumlah diamond yang dikirim dan data itu akan terus konsisten begitu juga dengan Player B yang menerima diamond dari Player A dan ditambahkan/Update ke total diamond Player B dan data akhirnya akan konsisten terus sampai perubahan/transaksi selanjutnya dilakukan