

DB relazionale: transazioni

Transazione:

macro-operazione che racchiude, al suo interno, una serie di sub-operazioni le quali possono raggiungere il successo o il fallimento solo restando unite.

Esempio transazione

Trasferimento di una somma da un conto corrente ad un altro

UPDATE CC

SET Saldo = Saldo - 50

WHERE Conto = 123

UPDATE CC

SET Saldo = Saldo + 50

WHERE Conto = 235

ACID: proprietà “acide”

ACID è l'acronimo inglese di:

Atomicity, Consistency, Isolation, Durability
(Atomicità, Coerenza, Isolamento e Durabilità)

ed indica l'insieme delle proprietà logiche necessarie per permettere alle transazioni di operare correttamente sui dati

la transazione è indivisibile nella sua esecuzione e la sua esecuzione deve essere o totale o nulla, non sono ammesse esecuzioni parziali.

Coerenza

quando inizia una transazione il database si trova in uno stato coerente; quando la transazione termina il database deve essere in un altro stato coerente, ovvero non deve violare eventuali vincoli di integrità; quindi non devono verificarsi contraddizioni (incoerenze) tra i dati archiviati nel DB.

Isolamento

ogni transazione deve essere eseguita in modo isolato e indipendente dalle altre transazioni, l'eventuale fallimento di una transazione non deve interferire con le altre transazioni in esecuzione.

Durabilità (persistenza)

si ottiene garantendo che l'effetto delle transazioni eseguite con successo sia mantenuto per sempre nel database, anche in caso di guasti e malfunzionamenti.

Commit

nel momento in cui avviamo una transazione, le modifiche effettuate dalle singole operazioni in essa contenute restano "sospese" ed "invisibili" agli utilizzatori del database, fino a quando non vengono confermate mediante l'operazione di "commit".

Rollback

Le modifiche "sospese" possono essere annullate del tutto mediante un'operazione di " rollback " la quale ripristina la situazione immediatamente antecedente all'inizio della *transaction*.

Esempio transazione mysql

```
START TRANSACTION;  
SELECT @totSal:=SUM(salary)  
FROM salaries WHERE type=1;  
UPDATE employee  
SET summary=@totSal  
WHERE type=1;  
COMMIT;
```

Esempio rollback

```
mysql> START TRANSACTION;
```

```
mysql> INSERT INTO fyi_links (url, id) VALUES ('google.com', 102);
```

```
mysql> INSERT INTO fyi_links (url, id) VALUES ('myspace.com', 103);
```

```
mysql> SELECT * FROM fyi_links;
```

id	url	notes	counts	created
101	fyicenter.com	NULL	NULL	2006-07-01 20:34:10
102	google.com	NULL	NULL	2006-07-01 20:37:06
103	myspace.com	NULL	NULL	2006-07-01 20:37:17
110	centerfyi.com	NULL	NULL	2006-07-01 20:34:12

```
mysql> ROLLBACK;
```

```
mysql> SELECT * FROM fyi_links;
```

id	url	notes	counts	created
101	fyicenter.com	NULL	NULL	2006-07-01 20:34:10
110	centerfyi.com	NULL	NULL	2006-07-01 20:34:12