

DCL, TCL, Build Structure

Reja:

1. DCL (Grant, Revoke)
2. TCL(Commit, Rollback, Savepoint)
3. Build Structure(ONE to ONE, ONE to MANY,
MANY to ONE, MANY to MANY)

DCL - DATA CONTROL LANGUAGE

Ushbu bo`lim ma`lumotlar bazasidagi imtiyozlarni boshqarish uchun ishlatiladigan buyruqlardan iborat.

Buyruqlar quyidagilardir:

- [GRANT](#)
- [REVOKE](#)

GRANT

- GRANT buyrug`i foydalanuvchiga kirish huquqini yoki sxema uchun boshqa imtiyozlarni berish uchun ishlatiladi.

Syntax:

```
GRANT privileges ON object TO user;
```

Example:

```
1 | GRANT INSERT ON TeachersInfo TO PUBLIC;
```

REVOKE

- REVOKE buyrug`i GRANT buyrug`i yordamida foydalanuvchining kirish huquqlarini bekor qilish uchun ishlatiladi.

Syntax:

```
REVOKE privileges ON object FROM user;
```

Example:

```
1 | REVOKE INSERT ON TeachersInfo FROM PUBLIC;
```

PostgreSQL da TCL **tranzaksiyalarni boshqarish tilini anglatadi** .

- Ma`lumotlar bazasidagi yagona ish birligi buyruqlar ketma-ket bajarilgandan so`ng tranzaksiya shakllanadi.
- PostgreSQLda TCL buyruqlari deb nomlanuvchi ma`lum buyruqlar mavjud bo`lib, ular foydalanuvchiga ma`lumotlar bazasida amalga oshiriladigan tranzaksiyalarni boshqarishga yordam beradi.

Ular quyidagilar:

- BEGIN
- COMMIT
- ROLLBACK
- SAVEPOINT
- RELEASE SAVEPOINT
- SET TRANSACTION

BEGIN

- Tranzaksiyani boshlash uchun BEGIN TRANSACTION buyrug'i ishlatiladi.

Syntax:

```
BEGIN;
```

```
BEGIN TRANSACTION;
```

Example:

```
1 | BEGIN;  
2 | DELETE * FROM TeachersInfo WHERE Salary = 65000;
```

COMMIT

- COMMIT buyrug`i oxirgi COMMIT yoki ROLLBACK buyrug`idan keyin ma`lumotlar bazasiga barcha operatsiyalarni saqlaydi.

Syntax:

```
COMMIT;
```

Example:

```
1 | DELETE * FROM TeachersInfo WHERE Salary = 65000;  
2 | COMMIT;
```


ROLLBACK

- ROLLBACK buyrug'i so'nggi COMMIT yoki ROLLBACK buyrug'i chaqirilgandan keyin tranzaktsiyalarni bekor qilish uchun ishlatiladi.

Syntax:

ROLLBACK;

Example:

```
1 | DELETE * FROM TeachersInfo WHERE Salary = 65000;  
2 | ROLLBACK;
```

SAVEPOINT

- SAVEPOINT buyrug'i joriy tranzaksiya doirasida yangi saqlash nuqtasini belgilaydi.

Syntax:

`SAVEPOINT savepoint_name; --Syntax for saving the SAVEPOINT`

`ROLLBACK TO savepoint_name --Syntax for rolling back to the SAVEPOINT`

Example:

```
1 | SAVEPOINT SP1;  
2 | DELETE FROM TeachersInfo WHERE Fees = 65000;  
3 | SAVEPOINT SP2;
```

RELEASE SAVEPOINT

- RELEASE SAVEPOINT buyrug`i siz yaratgan SAVE POINTni olib tashlash uchun ishlatiladi.

Syntax:

```
RELEASE SAVEPOINT savepoint_name;
```

Example:

```
1 | RELEASE SAVEPOINT SP2;
```

SET TRANSACTION

- SET TRANSACTION buyrug`i joriy tranzaksiya xarakteristikalarini o`rnatadi.

Syntax:

```
SET TRANSACTION transaction_mode;
```

Ma`lumotlar bazasi jadvalidagi aloqa turlari

Relyatsion ma`lumotlar bazasi jadvallar, yozuvlar va ustunlardan foydalanadigan har xil turdagi ma`lumotlar to`plamini to`playdi. U relyatsion ma`lumotlar bazalarini osongina saqlash uchun ma`lumotlar bazasi jadvallari o`rtasida aniq belgilangan munosabatlarni yaratish uchun ishlatiladi. Relyatsion ma`lumotlar bazasining bir nechta muhim parametrlari mavjud:

- U relyatsion modelga asoslangan (jadvallardagi ma`lumotlar).
- Jadvaldagi har bir satr noyob identifikator, kalitga ega.
- Jadval ustunlari ma`lumotlar atributlarini o`z ichiga oladi.

Xodimlar jadvali (Munosabatlar / Jadval nomi)

EmpID	EmpName	EmpAge	Mamlakat nomi
Emp 101	Endryu Metyu	24	AQSH
Emp 102	Markus dugles	27	Angliya
Emp 103	Engidi Nathem	28	Fransiya
Emp 104	Jeyson Quilt	21	Yaponiya
Emp 108	Robert	29	Italiya

Quyida relyatsion ma'lumotlar bazasi jadvallarining har xil turlari keltirilgan:

ONE to ONE (birga bir munosabat (1:1)): Bu birinchi jadvalning bitta qatori ikkinchi jadvalning faqat bitta va faqat bitta yozuvi bilan bog`lanishi mumkin bo`lgan ikkita jadval o`rtasida munosabatlarni yaratish uchun ishlatiladi. Xuddi shunday, ikkinchi jadval qatori ham birinchi jadvalning istalgan qatori bilan bog`lanishi mumkin.

Customer **(Table A)**

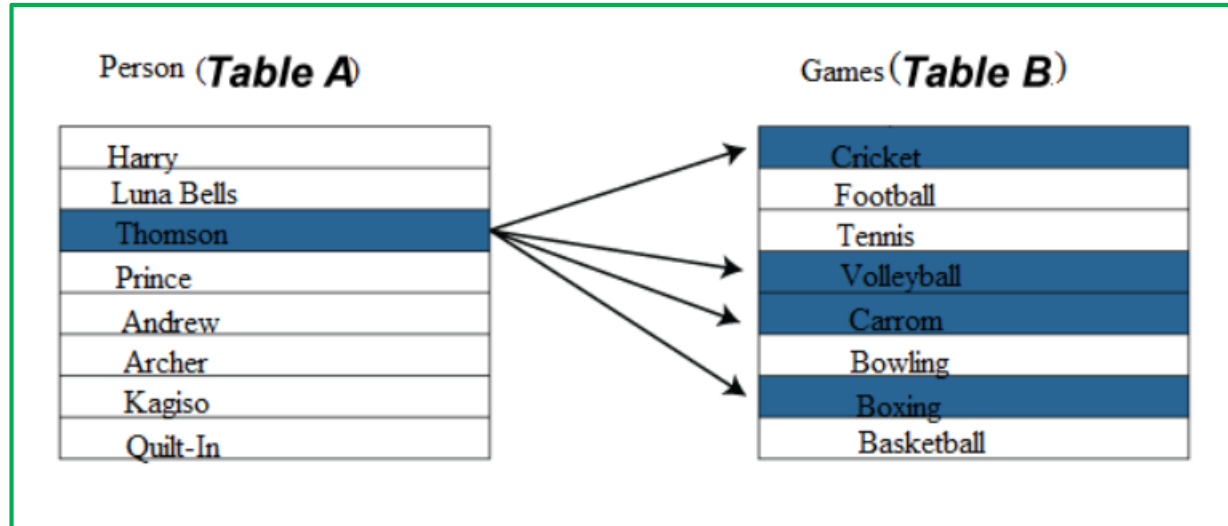
Peter
Susan McLain
Rosales
Bob Jones
Sue Williams
Brown
Howard
Taylor



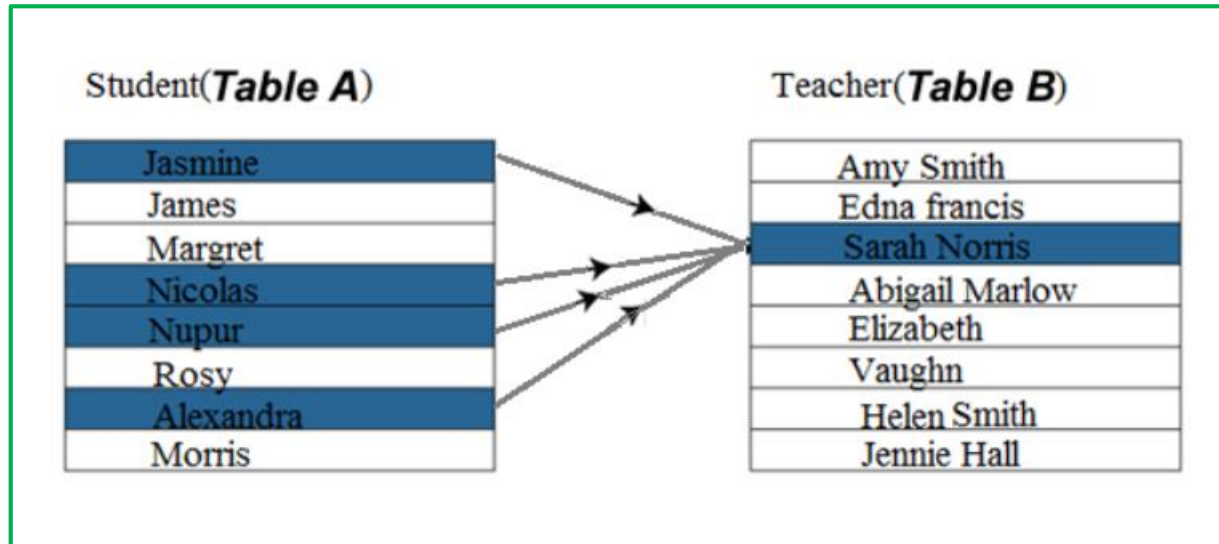
Order **(Table B)**

Ord102
Ord103
Ord105
Ord107
Ord108
Ord120
Ord106
Ord100

ONE to MANY (birga ko'p munosabat): U ikkita jadval o`rtasida munosabatlarni yaratish uchun ishlatiladi. Birinchi jadvalning har qanday bitta satri ikkinchi jadvalning bir yoki bir nechta satri bilan bog`liq bo`lishi mumkin, lekin ikkinchi jadvalning satrlari faqat birinchi jadvaldagi yagona qatorga tegishli bo`lishi mumkin. U **ko`pdan birga** munosabatlar sifatida ham tanilgan .



MANY to ONE (ko`pga bir munosabat): U ikkita jadval o`rtasida munosabatlarni yaratish uchun ishlatiladi. Birinchi jadvalning ko`plab satri ikkinchi jadvalning bitta satri bilan bog`liq bo`lishi mumkin, lekin birinchi jadvalning satrlari faqat ikkinchi jadvaldagi yagona qatorga tegishli bo`lishi mumkin.



MANY to MANY (ko`pga ko`p munosabatlar): U ko`p ko`p ikki jadval orasidagi munosabatni yaratish uchun ishlatiladi. Birinchi jadvalning har bir yozuvi ikkinchi jadvaldagi har qanday yozuvga (yoki hech qanday yozuvga) tegishli bo`lishi mumkin. Xuddi shunday, ikkinchi jadvalning har bir yozuvi birinchi jadvalning bir nechta yozuvlariga tegishli bo`lishi mumkin.

“Many to Many” Exercise

- What is the name of Lisa's child? _____



dbo.Parent	
P_ID	P_Name
1001	Lisa
1002	Barry
1003	Tina
1004	Sally
1005	Clive

dbo.ParentChild	
P_ID	C_ID
1001	2002
1002	2004
1002	2005
1003	2004
1003	2005
1004	2003
1005	2002
1005	2001

dbo.Child	
C_ID	C_Name
2001	Billy
2002	Nate
2003	Tonya
2004	Rachel
2005	Mark



E`TIBORINGIZ UCHUN RAHMAT!