

Résultats de test

Luqin REN

18/06/2020

1.16 Tables

```
postgres=# select * from budget;
```

call	proposal	bid	amount	object	type
1	2	1	800	computer	material
1	2	2	300	advertisement	operating
1	2	3	300	human	operating
1	6	1	300	computer	material
1	6	2	10000	human	operating
2	1	1	700	computer	material
2	1	2	13000	human	operating
2	2	1	600	computer	material
2	2	2	12000	human	operating
1	5	1	300	computer	material
1	5	2	1000	human	operating
2	3	1	6000	computer	material
2	3	2	9000	advertisement	operating

(13 行记录)

```
postgres=# select * from LEGALENTITY;
```

code	name	start_date	end_date	type
FR	France	1792-09-22		country
FT	France Telecom	1988-04-12		company
HF	Haut de France	2014-03-18		region
C	Compiègne	1996-01-30		city
TOOC	Tokyo Olympic Organizing Committee	2018-09-22	2020-12-31	organization
ADIT	ADIT Biology	1987-03-15		laboratory
MGL	Meuilleur Géologie Lab	2013-09-22		laboratory

(7 行记录)

```
postgres=# select * from PERSON;
```

id	name	email	legality
1	Luqin, Ren	["ren@etu.utc.fr"]	ADIT
2	Phillipe, Martin	["phillipe@etu.utc.fr", "phillipmartin@adit.fr"]	ADIT
3	Alice, Bernard	["alice@gmail.com", "alicebernard@adit.fr"]	ADIT
4	Betty, Thomas	["betty@etu.utc.fr", "bettythomas@adit.fr"]	ADIT
5	Celine, Petit	["celine@etu.utc.fr", "celinepetit@france.fr"]	FR
6	Durand, Robert	["durand@etu.utc.fr", "robertdurand@compiègne.fr"]	C
7	Emma, Richard	["richard@etu.utc.fr", "emmarichard@hf.fr"]	HF
8	Florence, Durand	["florence@etu.utc.fr", "florence@durand@ft.fr"]	FT
9	Gamma, Dubois	["gamma@etu.utc.fr", "duboisgamma@france.fr"]	FR
10	Hilton, Moreau	["hilton@gmail.com"]	C
11	Ilban, Laurent	["ilban@gmail.com"]	TOOC
12	Jimmy, Simon	["jimmy@gmail.com"]	ADIT
13	King, Michel	["king@gmail.com"]	MGL

(13 行记录)

```
postgres=# select * from LABMEMBER;
```

id	type	research_time_quota	edu_establishment	speciality	thesis_start_date	thesis_end_date	thesis_subject
1	TeacherResearcher	56	UTC				
2	ResearchEngineer			Biology	2015-02-06	2019-09-29	Importance of enviromental diversity
3	Doctor				2018-12-08		Database design
4	Doctor						

(4 行记录)

```
postgres=# select * from call;
```

id	date_launch	date_finish	theme	description	committee	organization
1	2015-02-06	2019-09-29	Environment	Environment protection	EPCCC	OCK
2	2015-02-06	2021-09-29	Food	Food nutrition	ABCDE	LOT
3	2017-02-06	2023-08-31	Earth	Earth mouvement	IYEH	MPC
4	2010-02-06	2024-10-29	Transport	Transport Design	DJBF	ELB

(4 行记录)

```
postgres=# select * from proposal;
```

call	pid	date_submit	date_response	response	project
1	2	2015-04-09	2015-07-29	refusal	
1	3	2015-05-09	2015-08-29	refusal	
1	4	2015-06-09	2015-09-29	refusal	
1	5	2015-07-09	2015-10-29	acceptance	2
1	6	2015-08-09	2015-11-29	acceptance	6
2	1	2015-03-09	2015-03-29	acceptance	3
2	2	2015-04-09	2015-05-29	acceptance	7
2	3	2015-05-09	2015-05-29	acceptance	8
3	1	2017-06-09	2020-06-17	acceptance	

(9 行记录)

```
postgres=# select * from EXPENSE;
```

project	demandor	validator	date	amount	type
2	1	2	2019-09-11	100	operating
3	1	2	2018-08-02	100	operating
6	1	3	2019-08-24	500	operating
7	2	6	2020-10-14	200	material
8	2	7	2020-11-05	300	operating
2	2	8	2019-07-11	300	operating
3	3	1	2018-07-02	300	material
6	4	2	2019-08-21	300	operating
7	5	1	2020-09-14	400	material
8	6	2	2020-10-05	200	operating
2	7	1	2019-05-11	100	material
3	8	2	2018-06-02	200	operating

(12 行记录)

```
postgres=# select * from PROJECTMEMBER;
```

project	member	position
2	1	Leader
2	2	Researcher
2	7	Cashier
2	8	Researcher
3	1	Leader
3	2	Researcher
3	3	Cashier
3	8	Researcher
6	1	Leader
6	2	Researcher
6	3	Cashier
6	4	Researcher
7	2	Leader
7	6	Researcher
7	5	Cashier
7	1	Researcher
8	2	Leader
8	6	Researcher
8	7	Cashier

(19 行记录)

2.18 Views

```
/*
    VUE pour voir les classes filles dans l'héritage par mère
*/

CREATE VIEW vTeacherResearcher AS
SELECT p.id, p.Name, lm.Research_time_quota, lm.Edu_establishment
FROM LabMember lm, Person p
WHERE lm.id = p.id AND Type = 'TeacherResearcher';
```

```
CREATE VIEW vResearchEngineer AS
SELECT p.id, p.Name, lm.Speciality
FROM LabMember lm, Person p
WHERE lm.id = p.id AND Type = 'ResearchEngineer';
```

```
CREATE VIEW vDoctor AS
SELECT p.id, p.Name, lm.Thesis_start_date, lm.Thesis_end_date, lm.Thesis_subject
FROM LabMember lm, Person p
WHERE lm.id = p.id AND Type = 'Doctor';
```

```
postgres=# select * from VTEACHERRESEARCHER;
```

id	name	research_time_quota	edu_establishment
1	["First Name": "Luqin", "Last Name": "Ren"]	56	UTC

(1 行记录)

```
postgres=# select * from VRESEARCHENGINEER;
```

id	name	speciality
2	["First Name": "Phillipe", "Last Name": "Martin"]	Biology

(1 行记录)

```
postgres=# select * from VDOCTOR;
```

id	name	thesis_start_date	thesis_end_date	thesis_subject
3	["First Name": "Alice", "Last Name": "Bernard"]	2015-02-06	2019-09-29	Importance of enviromental diversity
4	["First Name": "Betty", "Last Name": "Thomas"]	2018-12-08		Database design

(2 行记录)

```
/*
    VUE pour voir les dépenses de chaque type de chaque projet
*/
```

```
CREATE VIEW vProjectExpense AS
SELECT project, Type, SUM(Amount) AS Expense
FROM Expense
GROUP BY project, Type
ORDER BY project;
```

```
postgres=# select * from VPROJECTEXPENSE;
```

project	type	expense
2	material	100
2	operating	400
3	operating	300
3	material	300
6	operating	800
7	material	600
8	operating	500

(7 行记录)

```
/*
    VUE pour voir les membres internes de chaque projet
*/
```

```
CREATE VIEW vProjectInternalMember AS
SELECT pm.project, pm.member, pm.position, p.Name
FROM ProjectMember pm, LabMember lm, Person p
WHERE pm.member = lm.id AND p.id = lm.id
ORDER BY pm.project, pm.member;
```

```
postgres=# select * from VPROJECTINTERNALMEMBER;
```

project	member	position	name
2	1	Leader	{"First Name": "Luqin", "Last Name": "Ren"}
2	2	Researcher	{"First Name": "Phillipe", "Last Name": "Martin"}
3	1	Leader	{"First Name": "Luqin", "Last Name": "Ren"}
3	2	Researcher	{"First Name": "Phillipe", "Last Name": "Martin"}
3	3	Cashier	{"First Name": "Alice", "Last Name": "Bernard"}
6	1	Leader	{"First Name": "Luqin", "Last Name": "Ren"}
6	2	Researcher	{"First Name": "Phillipe", "Last Name": "Martin"}
6	3	Cashier	{"First Name": "Alice", "Last Name": "Bernard"}
6	4	Researcher	{"First Name": "Betty", "Last Name": "Thomas"}
7	1	Researcher	{"First Name": "Luqin", "Last Name": "Ren"}
7	2	Leader	{"First Name": "Phillipe", "Last Name": "Martin"}
8	2	Leader	{"First Name": "Phillipe", "Last Name": "Martin"}

(12 行记录)

```
/*
    VUE pour voir les budgets de chaque type dans la proposition de cha
que projet
*/
```

```
CREATE VIEW vProjectBudget AS
SELECT p.project, b.Type, SUM(b.Amount) AS Budget
```

```

FROM Budget b, Proposal p
WHERE b.proposal = p.pid AND b.call = p.call AND p.response = 'acceptance'
GROUP BY p.project, b.Type
ORDER BY p.project;

```

postgres=# select * from VPROJECTBUDGET;

project	type	budget
2	material	300
2	operating	1000
3	material	700
3	operating	13000
6	material	300
6	operating	10000
7	material	600
7	operating	12000
8	material	6000
8	operating	9000

(10 行记录)

```

/*
    VUE pour voir les projets pour lesquels il reste du budget à dépenser
*/

```

```

CREATE VIEW vProjectsHaveBudget AS
SELECT vpb.project, vpb.Type, vpb.Budget, vpe.Expense
FROM vProjectBudget vpb
LEFT JOIN vProjectExpense vpe ON vpe.project = vpb.project AND vpe.type
= vpb.type
WHERE vpb.Budget > vpe.Expense OR vpe.Expense IS NULL
ORDER BY vpb.project;

```

```
postgres=# select * from VPROJECTSHAVEBUDGET;
```

project	type	budget	expense
2	material	300	100
2	operating	1000	400
3	material	700	300
3	operating	13000	300
6	material	300	
6	operating	10000	800
7	operating	12000	
8	material	6000	
8	operating	9000	500

(9 行记录)

```
/*
    VUE pour voir les projets en cours
*/
```

```
CREATE VIEW vProjectCurrent AS
SELECT *
FROM Project
WHERE End_date > CURRENT_DATE;
```

```
postgres=# select * from VPROJECTCURRENT;
```

id	start_date	end_date
6	2019-07-21	2022-11-07
7	2020-08-14	2023-09-29
8	2020-09-05	2024-10-29

(3 行记录)

```
/*
    VUE pour voir les appels d'offre en cours non répondus
*/
```

```
CREATE VIEW vCall_Without_Proposal AS
SELECT c.id, c.Date_Launch, c.Date_Finish, c.Theme, c.Description, c.Committee, c.organization
```

```

FROM Call c
LEFT JOIN Proposal p ON c.id = p.call
WHERE p.call IS NULL AND c.Date_Finish > CURRENT_DATE;

```

```

postgres=# select * from VCALL_WITHOUT_PROPOSAL;
 id | date_launch | date_finish | theme | description | committee | organization
-----+-----+-----+-----+-----+-----+-----
  4 | 2010-02-06 | 2024-10-29 | Transport | Transport Design | DJBF | ELB
(1 行记录)

```

```

/*
    VUE pour voir le nombre de propositions répondues par type d'organi
sme projet
*/

```

```

CREATE VIEW vNb_Proposal_Responded AS
SELECT c.Organization, COUNT(p.Response) AS Nb_Response
FROM Call c
LEFT JOIN Proposal p ON c.id = p.Call
WHERE p.Response IS NOT NULL
GROUP BY c.Organization;

```

```

postgres=# select * from VNB_PROPOSAL_RESPONDED;
 organization | nb_response
-----+-----
 OCK          |          5
 MPC          |          1
 LOT          |          3
(3 行记录)

```

```

/*
    VUE pour voir le nombre de propositions acceptées par type d'organi
sme projet
*/

```

```

CREATE VIEW vNb_Proposal_Accepted AS
SELECT c.Organization, COUNT(p.Response) AS Nb_Accepted
FROM Call c
LEFT JOIN Proposal p ON c.id = p.Call
WHERE p.Response = 'acceptance'
GROUP BY c.Organization;

```



```
postgres=# select * from VNB_PROPOSAL_ACCEPTED;
```

organization	nb_accepted
LOT	3
MPC	1
OCK	2

(3 行记录)

```
/*
    VUE pour voir le nombre de propositions répondues
    par type d'organisme projet avec le nombre de projet accepté
*/
```

```
CREATE VIEW vNb_Proposal_Responded_Accepted AS
SELECT vnpr.Organization, vnpr.Nb_Response, vnpa.Nb_Accepted
FROM vNb_Proposal_Responded vnpr, vNb_Proposal_Accepted vnpa
WHERE vnpr.Organization = vnpa.Organization;
```

```
postgres=# select * from VNB_PROPOSAL_RESPONDED_ACCEPTED;
```

organization	nb_response	nb_accepted
OCK	5	2
MPC	1	1
LOT	3	3

(3 行记录)

```
/*
    VUE pour voir le mois de l'année où on fait le plus de dépenses de
    projet au laboratoire
*/
```

```
CREATE VIEW vMonth_Expense AS
SELECT EXTRACT (MONTH FROM e.Date) AS Month, SUM(e.Amount) AS Expense
FROM Expense e
GROUP BY Month
ORDER BY Expense DESC;
```

```
postgres=# select * from VMONTH_EXPENSE;
```

month	expense
8	900
7	600
9	500
10	400
11	300
6	200
5	100

(7 行记录)

```
/*
    VUE JSON pour voir les membres du laboratoire affectés au plus grand
    nombre de projet en cours (top 5)
*/
```

```
CREATE VIEW vActive_LabMembers AS
SELECT pm.member, p.Name #>> '{Last Name}' AS Last_Name, p.Name #>> '{First Name}' AS First_Name, COUNT(pm.project) AS Nb_Project
FROM vProjectCurrent vpc, LabMember lm, ProjectMember pm, Person p
WHERE vpc.id = pm.project AND p.id = lm.id AND lm.id = pm.member
GROUP BY pm.member, p.Name #>> '{Last Name}', p.Name #>> '{First Name}'

ORDER BY Nb_Project DESC
LIMIT 5;
```

```
postgres=# select * from VACTIVE_LABMEMBERS;
```

member	last_name	first_name	nb_project
2	Martin	Phillipe	3
1	Ren	Luqin	2
3	Bernard	Alice	1
4	Thomas	Betty	1

(4 行记录)

```
/*
    VUE JSON pour afficher toutes les adresses e-mail de tout le monde
```

```

*/
CREATE VIEW vPerson_Emails AS
SELECT p.Name #>> '{Last Name}' AS Last_Name, p.Name #>> '{First Name}'
  AS First_Name, pe.*
FROM Person p, JSON_ARRAY_ELEMENTS(p.Email) pe;

```

```

postgres=# select * from VPERSON_EMAILS;
 last_name | first_name | value
-----+-----+-----
 Ren       | Luqin      | "ren@etu.utc.fr"
 Ren       | Luqin      | "renluqin@adit.fr"
 Martin    | Phillipe   | "phillipe@etu.utc.fr"
 Martin    | Phillipe   | "phillipemartin@adit.fr"
 Bernard   | Alice      | "alice@gmail.com"
 Bernard   | Alice      | "alicebernard@adit.fr"
 Thomas    | Betty      | "betty@etu.utc.fr"
 Thomas    | Betty      | "bettythomas@adit.fr"
 Petit     | Celine     | "celine@etu.utc.fr"
 Petit     | Celine     | "celinepetit@france.fr"
 Robert    | Durand     | "durand@etu.utc.fr"
 Robert    | Durand     | "robertdurand@compiegne.fr"
 Richard   | Emma       | "richard@etu.utc.fr"
 Richard   | Emma       | "emmarichard@hf.fr"
 Durand    | Florence   | "florence@etu.utc.fr"
 Durand    | Florence   | "florencedurand@ft.fr"
 Dubois    | Gamma      | "gamma@etu.utc.fr"
 Dubois    | Gamma      | "duboisgamma@france.fr"
 Moreau    | Hilton     | "hilton@gmail.com"
 Laurent   | Ilban      | "ilban@gmail.com"
 Simon     | Jimmy      | "jimmy@gmail.com"
 Michel    | King       | "king@gmail.com"
(22 行记录)

```

3.4 Tests

```

/*
    TEST SUPPRESSION : La suppression d'une Proposal devrait entraîner
    la suppression de tous les Budgets de cette Proposal.
    Le résultat attendu est nul.
*/

DELETE FROM Proposal WHERE Call = '1' AND PID = '1';
SELECT * FROM Budget WHERE Call = '1' AND Proposal = '1';

```

```
postgres=# DELETE FROM Proposal WHERE Call = '1' AND PID = '1';
DELETE 0
postgres=# SELECT * FROM Budget WHERE Call = '1' AND Proposal = '1';
 call | proposal | bid | amount | object | type
-----+-----+-----+-----+-----+-----
(0 行记录)
```

/*

TEST MISE A JOUR : Accepter une proposition.

*/

```
UPDATE Proposal set Date_response = CURRENT_DATE, Response = 'acceptanc
e' WHERE call = '3' AND PID = '1';
```

```
SELECT * FROM Proposal WHERE call = '3' AND PID = '1';
```

```
postgres=# UPDATE Proposal set Date_response = CURRENT_DATE, Response = 'acceptance' WHERE call = '3' AND PID = '1'
UPDATE 1
postgres=# SELECT * FROM Proposal WHERE call = '3' AND PID = '1';
 call | pid | date_submit | date_response | response | project
-----+-----+-----+-----+-----+-----
  3   |  1   | 2017-06-09  | 2020-06-18   | acceptance |
(1 行记录)
```

/*

TEST INSERTION : Impossible de créer deux projets depuis la même proposition.

Le résultat attendu est une erreur.

```
INSERT INTO Proposal (call, PID, Date_submit, Date_response, Response,
project) VALUES ('2', '3', '2015-05-09', '2015-05-
29', 'acceptance', '9');
```

*/

```
postgres=# INSERT INTO Proposal (call, PID, Date_submit, Date_response, Response, project) VALUES ('2', '3', '2015-05-09
', '2015-05-29', 'acceptance', '9');
错误: 重复键违反唯一约束"proposal_pkey"
描述: 键值"(call, pid)=(2, 3)" 已经存在
postgres=#
```

/*

TEST REQUETE : Afficher les propositions avec plus de deux labels.

*/

```
SELECT pl.call, pl.proposal, l.LegalEntity, COUNT(pl.label) AS Nb_Label
FROM Label l, ProposalLabel pl
WHERE l.Name = pl.label
GROUP BY pl.call, pl.proposal, l.LegalEntity
HAVING COUNT(pl.label) >= 2
ORDER BY pl.call, pl.proposal;
```

```
postgres=# SELECT pl.call, pl.proposal, l.LegalEntity, COUNT(pl.label) AS Nb_Label
postgres=# FROM Label l, ProposalLabel pl
postgres=# WHERE l.Name = pl.label
postgres=# GROUP BY pl.call, pl.proposal, l.LegalEntity
postgres=# HAVING COUNT(pl.label) >= 2
postgres=# ORDER BY pl.call, pl.proposal;
 call | proposal | legalentity | nb_label
-----+-----+-----+-----
  1   |  5   | ADIT       | 2
(1 行记录)
```

4. GESTION DES DROITS

```
/*
-
- Créer un utilisateur LuqinREN qui ne nécessite pas de mot de passe pour se connecter
CREATE ROLE LuqinREN LOGIN;
-
- Créer des utilisateurs qui ont besoin d'un mot de passe pour se connecter
CREATE USER LuqinREN1 WITH PASSWORD 'rlq1';
CREATE USER Alice WITH PASSWORD 'alice';
CREATE USER Betty WITH PASSWORD 'betty';
CREATE USER Celine WITH PASSWORD 'celine';
CREATE USER Davide WITH PASSWORD 'davide';
CREATE USER Emma WITH PASSWORD 'emma';
-- Créer un utilisateur à durée limitée LuqinREN2
CREATE ROLE LuqinREN2 WITH LOGIN PASSWORD 'luqinluqinren' VALID UNTIL '2023-05-30';
-
- Créer un administrateur utilisateur avec des autorisations pour créer des bases de données et gérer les rôles
CREATE ROLE admin WITH CREATEDB CREATEROLE;
-- Créer un utilisateur avec des super droits: spadmin
CREATE ROLE spadmin WITH SUPERUSER LOGIN PASSWORD 'spadmin';

-- Autorisation de toutes les tables
GRANT SELECT,INSERT,UPDATE,DELETE ON ALL TABLES IN SCHEMA PUBLIC to LuqinREN;
-- Autorisation de table spécifique
GRANT SELECT,INSERT,UPDATE,DELETE ON LegalEntity, LegalEntityOrganization, Organization to Alice;
GRANT SELECT,INSERT,UPDATE,DELETE ON Person, LabMember, EmployeeLE, EmployeeContact to Betty;
GRANT SELECT,INSERT,UPDATE,DELETE ON Call, Proposal, Project, Budget, Label, ProposalLabMember to Celine;
GRANT SELECT,INSERT,UPDATE,DELETE ON Call, Project, Budget, ProposalLabMember, Expense, ProjectMember to Davide;
-
- Accorder différents droits en fonction de différentes colonnes de table
GRANT UPDATE (Date_response, Response, Project), INSERT(call, pid, date_submit) ON Proposal to Emma;
```

```
-- Autorisation des vues
GRANT SELECT ON vDoctor, vResearchEngineer, vTeacherResearcher TO PUBLI
C;
GRANT ALL PRIVILEGES ON vCheckOrganizationLegalEntity TO Alice;
GRANT ALL PRIVILEGES ON vDoctor, vResearchEngineer, vTeacherResearcher,
vCheckPersonLabMemberEmployeeLE, vCheckProposalLabMember, vActive_LabM
embers TO Betty;
GRANT ALL PRIVILEGES ON vNb_Proposal_Responded_Accepted, vNb_Proposal_A
ccepted, vNb_Proposal_Responded, vCall_Without_Proposal, vCheckProposal
LabMember TO Celine;
GRANT ALL PRIVILEGES ON vProjectCurrent, vProjectsHaveBudget, vProjectB
udget, vProjectInternalMember, vProjectExpense TO Davide;
*/
```

postgres=# select * from pg_roles;	rolname	rolsuper	rolinherit	rolcreaterole	rolcreatedb	rolcanlogin	rolreplication	rolcommit	rolpassword	rolvaliduntil	rolbypassrls	rolconfig	oid
pg_signal_backend	f	t	f	f	f	f	f	-1	*****		f		4290
celine	f	t	f	f	f	f	f	-1	*****		f		17158
pg_read_server_files	f	t	f	f	f	f	f	-1	*****		f		4569
postgres	t	t	t	t	t	t	t	-1	*****		f		10
betty	f	t	f	f	f	f	f	-1	*****		f		17157
pg_write_server_files	f	t	f	f	f	f	f	-1	*****		f		4570
spadmin	f	t	f	f	f	f	f	-1	*****		f		17163
luqinren2	f	t	f	f	f	f	f	-1	*****	2023-05-30 00:00:00+02	f		17161
pg_execute_server_program	f	t	f	f	f	f	f	-1	*****		f		4571
admin	f	t	t	f	f	f	f	-1	*****		f		17162
luqinren1	f	t	f	f	f	f	f	-1	*****		f		17155
pg_read_all_stats	f	t	f	f	f	f	f	-1	*****		f		3375
pg_monitor	f	t	f	f	f	f	f	-1	*****		f		3373
emma	f	t	f	f	f	f	f	-1	*****		f		17160
luqinren	f	t	f	f	f	t	f	-1	*****		f		17154
alice	f	t	f	f	f	t	f	-1	*****		f		17156
davide	f	t	f	f	f	t	f	-1	*****		f		17159
pg_read_all_settings	f	t	f	f	f	f	f	-1	*****		f		3374
pg_stat_scan_tables	f	t	f	f	f	f	f	-1	*****		f		3377

postgres=# select * from pg_user;	username	usesysid	usecreatedb	usesuper	userepl	usebypassrls	passwd	valuntil	useconfig
postgres	10	t	t	t	t		*****		
luqinren	17154	f	f	f	f		*****		
luqinren1	17155	f	f	f	f		*****		
alice	17156	f	f	f	f		*****		
betty	17157	f	f	f	f		*****		
celine	17158	f	f	f	f		*****		
davide	17159	f	f	f	f		*****		
emma	17160	f	f	f	f		*****		
luqinren2	17161	f	f	f	f		*****	2023-05-30 00:00:00+02	
spadmin	17163	f	t	f	f		*****		

(10 行记录)

角色列表		
角色名称	属性	成员属于
admin	建立角色，建立 DB，无法登录	{ }
alice		{ }
betty		{ }
celine		{ }
davide		{ }
emma		{ }
luqinren	密码有效直至2023-05-30 00:00:00+02	{ }
luqinren1		{ }
luqinren2		{ }
postgres		{ }
spadmin		{ }
	超级用户，建立角色，建立 DB，复制，绕过RLS	{ }
	超级用户	{ }