

Rapport Projet INFO0505

Bases de données avancées

Laboratoire de Recherche

Timofte Béatrice & Bouchhioua Wejdane
Groupe S5O6 A, CMII L3 INFORMATIQUE

Date
6 décembre 2023

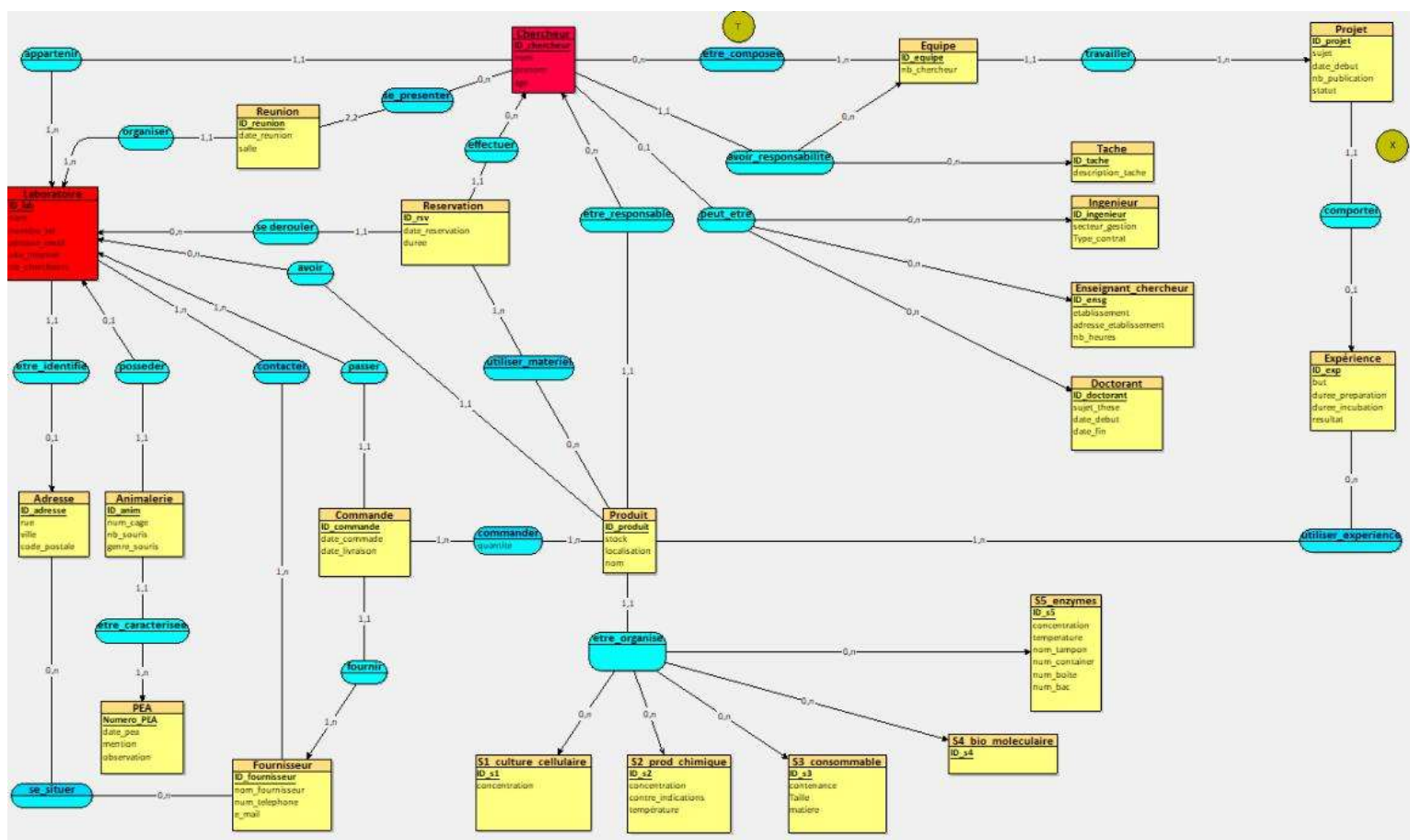
SOMMAIRE

1. Analysé et modélisation
2. Script de création de la base de données
3. Script d'insertion et d'enregistrement des données
4. Partie SQL : Requêtes
5. Partie PL/SQL : Programmes

1. Analyse et modélisation

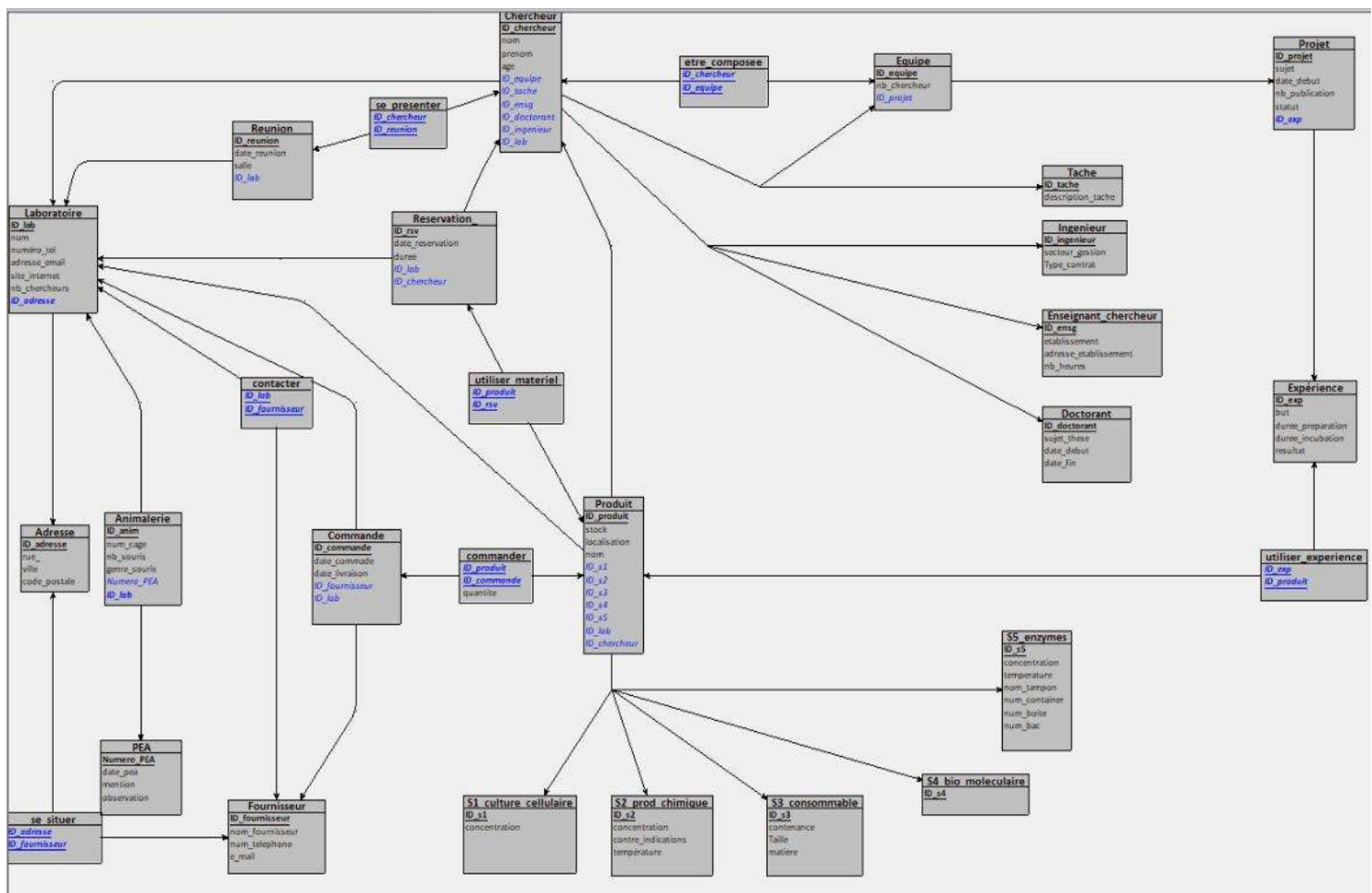
MCD = MODÈLE CONCEPTUEL DE DONNÉES :

Le MCD est une représentation graphique de haut niveau qui permet facilement et simplement de comprendre comment les différents éléments sont liés entre eux. Faisant partie de la boîte à outil Merise, le MCD décrit les données utilisées par le système d'information et leurs relations.



MLD = MODÈLE LOGIQUE DE DONNÉES :

En partant d'un MCD, un MLD introduit les éléments, les relations et les détails contextuels nécessaires à la structure des données afin de se rapprocher de la mise en œuvre.



MR

1. Script de création de la base de données

TABLES:

```
CREATE TABLE Adresse(  
  ID_adresse INT,
```

TABLES:

```
CREATE TABLE Adresse(  
  ID_adresse INT,  
  
  rue_ VARCHAR(50),  
  ville VARCHAR(50),  
  code_postale VARCHAR(50),  
  PRIMARY KEY(ID_adresse)  
);  
  
CREATE TABLE Laboratoire(  
  ID_lab INT,  
  nom VARCHAR(50),  
  numéro_tel VARCHAR(50),  
  adresse_email VARCHAR(50),  
  site_internet VARCHAR(50),  
  nb_chercheurs INT,  
  ID_adresse INT NOT NULL,  
  PRIMARY KEY(ID_lab),  
  UNIQUE(ID_adresse),  
  FOREIGN KEY(ID_adresse) REFERENCES Adresse(ID_adresse)  
);  
  
CREATE TABLE Enseignant_chercheur(  
  ID_engs INT,  
  etablissement VARCHAR(50),  
  adresse_etablissement VARCHAR(50),  
  nb_heures INT,  
  PRIMARY KEY(ID_engs)  
);  
  
CREATE TABLE Doctorant(  
  ID_doctorant INT,  
  sujet_these VARCHAR(50),  
  date_debut DATE,  
  date_fin DATE,  
  PRIMARY KEY(ID_doctorant)  
);
```

```
CREATE TABLE Ingenieur_(  
  ID_ingenieur INT,  
  secteur_gestion VARCHAR(50),  
  Type_contrat VARCHAR(50),  
  PRIMARY KEY(ID_ingenieur)
```

```
CREATE TABLE Experience(  
  ID_exp INT PRIMARY KEY,  
  but VARCHAR2(50),  
  duree_preparation NUMBER,  
  duree_incubation NUMBER,  
  resultat VARCHAR2(50)  
);
```

```
CREATE TABLE S1_culture_cellulaire(  
  ID_s1 INT,  
  concentration NUMBER,  
  PRIMARY KEY(ID_s1)  
);
```

```
CREATE TABLE S2_prod_chimique(  
  ID_s2 INT,  
  concentration NUMBER,  
  contre_indications VARCHAR(50),  
  temperature NUMBER,  
  PRIMARY KEY(ID_s2)  
);
```

```
CREATE TABLE S3_consommable(  
  ID_s3 INT,  
  contenance NUMBER,  
  Taille VARCHAR(50),  
  matiere VARCHAR(50),  
  PRIMARY KEY(ID_s3)  
);
```

```
CREATE TABLE S4_bio_moleculaire(  
  ID_s4 INT,  
  PRIMARY KEY(ID_s4)  
);
```

```
CREATE TABLE S5_enzymes(  
  ID_s5 INT,  
  concentration NUMBER,  
  temperature NUMBER,  
  nom_tampon VARCHAR(50),
```



```
num_container INT,  
num_boite INT,  
num_bac INT,  
PRIMARY KEY(ID_s5)  
);
```

```
CREATE TABLE Fournisseur(  
  ID_fournisseur INT,  
  nom_fournisseur VARCHAR(50),  
  num_telephone VARCHAR(50),  
  e_mail VARCHAR(50),  
  PRIMARY KEY(ID_fournisseur)  
);
```

```
CREATE TABLE Tache(  
  ID_tache INT,  
  description_tache VARCHAR(50),  
  PRIMARY KEY(ID_tache)  
);
```

```
CREATE TABLE PEA(  
  Numero_PEA INT,  
  date_pea DATE,  
  mention VARCHAR(50),  
  observation VARCHAR(50),  
  PRIMARY KEY(Numero_PEA)  
);
```

```
CREATE TABLE Projet(  
  ID_projet INT,  
  sujet VARCHAR(50),  
  date_debut DATE,  
  nb_publication INT,  
  statut VARCHAR(50),  
  ID_exp INT NOT NULL,  
  PRIMARY KEY(ID_projet),  
  UNIQUE(ID_exp),  
  FOREIGN KEY(ID_exp) REFERENCES Experience(ID_exp)  
);
```

```
CREATE TABLE Commande(  
  ID_commande INT,
```

```
date_commade DATE,  
date_livraison DATE,  
ID_fournisseur INT NOT NULL,  
ID_lab INT NOT NULL,  
PRIMARY KEY(ID_commade),  
FOREIGN KEY(ID_fournisseur) REFERENCES Fournisseur(ID_fournisseur),  
FOREIGN KEY(ID_lab) REFERENCES Laboratoire(ID_lab) );
```

```
CREATE TABLE Animalerie(  
ID_anim INT,  
num_cage INT,  
nb_souris INT,  
genre_souris VARCHAR(50),  
Numero_PEA INT NOT NULL,  
ID_lab INT NOT NULL,  
PRIMARY KEY(ID_anim),  
UNIQUE(ID_lab),  
FOREIGN KEY(Numero_PEA) REFERENCES PEA(Numero_PEA),  
FOREIGN KEY(ID_lab) REFERENCES Laboratoire(ID_lab) );
```

```
CREATE TABLE Reunion(  
ID_reunion INT,  
date_reunion DATE,  
heure_reunion TIMESTAMP,  
salle VARCHAR(50),  
ID_lab INT NOT NULL,  
PRIMARY KEY(ID_reunion),  
FOREIGN KEY(ID_lab) REFERENCES Laboratoire(ID_lab)  
);
```

```
CREATE TABLE Equipe(  
ID_equipe INT,  
nb_chercheur INT,  
ID_projet INT NOT NULL,  
PRIMARY KEY(ID_equipe),  
FOREIGN KEY(ID_projet) REFERENCES Projet(ID_projet)  
);
```

```
CREATE TABLE Chercheur(  
  ID_chercheur INT,  
  nom VARCHAR(50),  
  prenom VARCHAR(50),  
  age INT,  
  ID_equipe INT NOT NULL,  
  ID_tache INT NOT NULL,  
  ID_ensg INT,  
  ID_doctorant INT,  
  ID_ingenieur INT,  
  ID_lab INT NOT NULL,  
  PRIMARY KEY(ID_chercheur),  
  FOREIGN KEY(ID_equipe) REFERENCES Equipe(ID_equipe),  
  FOREIGN KEY(ID_tache) REFERENCES Tache(ID_tache),  
  FOREIGN KEY(ID_ensg) REFERENCES Enseignant_chercheur(ID_ensg),  
  FOREIGN KEY(ID_doctorant) REFERENCES Doctorant(ID_doctorant),  
  FOREIGN KEY(ID_ingenieur) REFERENCES Ingenieur_(ID_ingenieur),  
  FOREIGN KEY(ID_lab) REFERENCES Laboratoire(ID_lab)  
);
```

```
CREATE TABLE Produit(  
  ID_produit INT,  
  stock INT,  
  localisation VARCHAR(50),  
  nom VARCHAR(50),  
  ID_s1 INT NOT NULL,  
  ID_s2 INT NOT NULL,  
  ID_s3 INT NOT NULL,  
  ID_s4 INT NOT NULL,  
  ID_s5 INT NOT NULL,  
  ID_lab INT NOT NULL,  
  ID_chercheur INT NOT NULL,  
  PRIMARY KEY(ID_produit),  
  FOREIGN KEY(ID_s1) REFERENCES S1_culture_cellulaire(ID_s1),  
  FOREIGN KEY(ID_s2) REFERENCES S2_prod_chimique(ID_s2),  
  FOREIGN KEY(ID_s3) REFERENCES S3_consommable(ID_s3),  
  FOREIGN KEY(ID_s4) REFERENCES S4_bio_moleculaire(ID_s4),  
  FOREIGN KEY(ID_s5) REFERENCES S5_enzymes(ID_s5),  
  FOREIGN KEY(ID_lab) REFERENCES Laboratoire(ID_lab),  
  FOREIGN KEY(ID_chercheur) REFERENCES Chercheur(ID_chercheur)  
);
```

```
CREATE TABLE Reservation_(
```

```
ID_rsv INT,  
date_reservation DATE,  
duree NUMBER,  
ID_lab INT NOT NULL,  
ID_chercheur INT NOT NULL,  
PRIMARY KEY(ID_rsv),  
FOREIGN KEY(ID_lab) REFERENCES Laboratoire(ID_lab),  
FOREIGN KEY(ID_chercheur) REFERENCES Chercheur(ID_chercheur)  
);
```

RELATIONS:

```
CREATE TABLE etre_composee(  
ID_chercheur INT,  
ID_equipe INT,  
PRIMARY KEY(ID_chercheur, ID_equipe),  
FOREIGN KEY(ID_chercheur) REFERENCES Chercheur(ID_chercheur),  
FOREIGN KEY(ID_equipe) REFERENCES Equipe(ID_equipe)  
);
```

```
CREATE TABLE se_situer(  
ID_adresse INT,  
ID_fournisseur INT,  
PRIMARY KEY(ID_adresse, ID_fournisseur),  
FOREIGN KEY(ID_adresse) REFERENCES Adresse(ID_adresse),  
FOREIGN KEY(ID_fournisseur) REFERENCES Fournisseur(ID_fournisseur)  
);
```

```
CREATE TABLE se_presenter(  
ID_chercheur INT,  
ID_reunion INT,  
PRIMARY KEY(ID_chercheur, ID_reunion),  
FOREIGN KEY(ID_chercheur) REFERENCES Chercheur(ID_chercheur),  
FOREIGN KEY(ID_reunion) REFERENCES Reunion(ID_reunion)  
);
```

```
CREATE TABLE commander(  
ID_produit INT,
```

```
ID_commande INT,  
quantite INT,  
PRIMARY KEY(ID_produit, ID_commande),  
FOREIGN KEY(ID_produit) REFERENCES Produit(ID_produit),  
FOREIGN KEY(ID_commande) REFERENCES Commande(ID_commande)  
);
```

```
CREATE TABLE utiliser_experience(  
ID_exp INT,  
ID_produit INT,  
PRIMARY KEY(ID_exp, ID_produit),  
FOREIGN KEY(ID_exp) REFERENCES Experience(ID_exp),  
FOREIGN KEY(ID_produit) REFERENCES Produit(ID_produit)  
);
```

```
CREATE TABLE utiliser_materiel(  
ID_produit INT,  
ID_rsv INT,  
PRIMARY KEY(ID_produit, ID_rsv),  
FOREIGN KEY(ID_produit) REFERENCES Produit(ID_produit),  
FOREIGN KEY(ID_rsv) REFERENCES Reservation_(ID_rsv)  
);
```

```
CREATE TABLE contacter(  
ID_lab INT,  
ID_fournisseur INT,  
PRIMARY KEY(ID_lab, ID_fournisseur),  
FOREIGN KEY(ID_lab) REFERENCES Laboratoire(ID_lab),  
FOREIGN KEY(ID_fournisseur) REFERENCES Fournisseur(ID_fournisseur)  
);
```

CONTRAINTES:

-- Table intermédiaire représentant la relation entre les projets
et les expériences

```
CREATE TABLE Projet_Experience (  
ID_projet INT,
```

```
ID_exp INT,  
PRIMARY KEY (ID_projet, ID_exp),  
FOREIGN KEY (ID_projet) REFERENCES Projet(ID_projet),  
FOREIGN KEY (ID_exp) REFERENCES Experience(ID_exp)  
);
```

-- Contrainte unique pour garantir qu'une expérience ne peut être utilisée que par un seul projet

```
ALTER TABLE Projet_Experience  
ADD CONSTRAINT UQ_Projet_Experience UNIQUE (ID_exp);
```

3. Script d'insertion et d'enregistrement des données

Table ADRESSE :

```
INSERT INTO Adresse (ID_adresse, rue_, ville, code_postale)  
VALUES (2, '5 Boulevard de la Paix', 'Paris', '95000');
```

```
INSERT INTO Adresse (ID_adresse, rue_, ville, code_postale)  
VALUES (3, '10 Rue des Chercheur', 'Marseille', '13008');
```

```
INSERT INTO Adresse (ID_adresse, rue_, ville, code_postale)  
VALUES (4, '15 Avenue des Sciences', 'Lyon', '69002');
```

```
INSERT INTO Adresse (ID_adresse, rue_, ville, code_postale)  
VALUES (5, '27 Rue de la République', 'Paris', '95000');
```

```
INSERT INTO Adresse (ID_adresse, rue_, ville, code_postale)  
VALUES (6, '18 Rue des Chevre', 'Cannes', '17000');
```

```
INSERT INTO Adresse (ID_adresse, rue_, ville, code_postale)
VALUES (7, '150 Avenue des Sacres', 'Lyon', '69002');
```

```
INSERT INTO Adresse (ID_adresse, rue_, ville, code_postale)
VALUES (8, '2 Rue de la Ferme', 'Paris', '95000');
```

ID_ADRESSE	RUE_	VILLE	CODE_POSTALE
1	4 Rue de l'Espoir	Reims	51100
2	5 Boulevard de la Paix	Paris	95000
3	10 Rue des Chercheur	Marseille	13008
4	15 Avenue des Sciences	Lyon	69002
5	27 Rue de la République	Paris	95000
6	18 Rue des Chevre	Cannes	17000
7	150 Avenue des Sacres	Lyon	69002
8	2 Rue de la Ferme	Paris	95000

Table LABORATOIRE :

```
INSERT INTO Laboratoire (ID_lab, nom, numéro_tel,
adresse_email, site_internet, nb_chercheurs, ID_adresse)
VALUES (1, 'Laboratoire de Biologie Moléculaire',
'0329105622', 'lab_bio@yahoo.com', 'http://lab-bio.com', 3, 1);
```

```
INSERT INTO Laboratoire (ID_lab, nom, numéro_tel,
adresse_email, site_internet, nb_chercheurs, ID_adresse)
VALUES (2, 'Institut de Recherche Génétique', '0369851200',
'institut_genetique@gmail.com', 'http://genetics-institute.com', 2, 2);
```

```
INSERT INTO Laboratoire (ID_lab, nom, numéro_tel,
adresse_email, site_internet, nb_chercheurs, ID_adresse)
```


VALUES (3, 'Centre de Biotechnologie Avancée', '0387201599',
'biotech_center@yahoo.com', 'http://biotech-center.com', 2, 3);

Table LABORATOIRE

ID_LAB	NOM	NUMÉRO_TEL	ADRESSE_EMAIL	SITE_INTERNET	NB_CHERCHEUR
1	Laboratoire de Biologie Moléculaire	329105622	lab_bio@yahoo.com	http://lab-bio.com	
2	Institut de Recherche Génétique	369851200	institut_genetique@gmail.com	http://genetics-institute.com	
3	Centre de Biotechnologie Avancée	387201599	biotech_center@yahoo.com	http://biotech-center.com	

Table REUNION :

INSERT INTO Reunion (ID_reunion, date_reunion, heure_reunion, salle, ID_lab)
VALUES (1, TO_DATE('2023-11-14', 'YYYY-MM-DD'),
TO_TIMESTAMP('20:31:51', 'HH24:MI:SS'), 'Salle G4T2 etage 6', 1);

INSERT INTO Reunion (ID_reunion, date_reunion, heure_reunion, salle, ID_lab)
VALUES (2, TO_DATE('2023-10-19', 'YYYY-MM-DD'),
TO_TIMESTAMP('10:30:28', 'HH24:MI:SS'), 'Salle L8K1, bat L, et 0', 1);

INSERT INTO Reunion (ID_reunion, date_reunion, heure_reunion, salle, ID_lab)
VALUES (3, TO_DATE('2023-12-14', 'YYYY-MM-DD'),
TO_TIMESTAMP('11:45:31', 'HH24:MI:SS'), 'Salle K1M9 et 5', 2);
INSERT INTO Reunion (ID_reunion, date_reunion, heure_reunion, salle, ID_lab)
VALUES (4, TO_DATE('2023-12-10', 'YYYY-MM-DD'),
TO_TIMESTAMP('12:00:19', 'HH24:MI:SS'), 'Salle O64, bat O', 3);

INSERT INTO Reunion (ID_reunion, date_reunion, heure_reunion, salle, ID_lab)

VALUES (5, TO_DATE('2023-12-17', 'YYYY-MM-DD'),
TO_TIMESTAMP('10:30:19', 'HH24:MI:SS'), 'Salle O64, bat O', 3);

ID_REUNION	DATE_REUNION	HEURE_REUNION	SALLE	ID_
1	14-NOV-23	01-NOV-23 08.31.51.000000 PM	Salle G4T2 etage 6	1
2	19-OCT-23	01-NOV-23 10.30.28.000000 AM	Salle L8K1, bat L, et 0	1
3	14-DEC-23	01-NOV-23 11.45.31.000000 AM	Salle K1M9 et 5	2
4	10-DEC-23	01-NOV-23 12.00.19.000000 PM	Salle 064, bat 0	3
5	17-DEC-23	01-NOV-23 10.30.19.000000 AM	Salle 064, bat 0	3

Table TACHE :

INSERT INTO Tache (ID_tache, description_tache)
VALUES (0, 'Le chercheur n'est pas responsable d'aucune tache.');

INSERT INTO Tache (ID_tache, description_tache)
VALUES (1, 'Coordonner équipe');

INSERT INTO Tache (ID_tache, description_tache)
VALUES (2, 'Gestionner les produits');

INSERT INTO Tache (ID_tache, description_tache)
VALUES (3, 'Collecter des échantillons biologiques');

INSERT INTO Tache (ID_tache, description_tache)
VALUES (4, 'Préparer des présentations');

INSERT INTO Tache (ID_tache, description_tache)
VALUES (5, 'Valider les protocoles expérimentaux');

Table INGENIEUR :

INSERT INTO Ingenieur_ (ID_ingenieur,

ID_TACHE	DESCRIPTION_TACHE
0	Le chercheur n'est pas responsable d'aucune tache.
1	Coordonner équipe
2	Gestionner les produits
3	Collecter des échantillons biologiques
4	Préparer des présentations
5	Valider les protocoles expérimentaux

secteur_gestion, Type_contrat)

VALUES (1, 'Gestion des équipements de laboratoire', 'CDI');

INSERT INTO Ingenieur_ (ID_ingenieur, secteur_gestion, Type_contrat)

VALUES (2, 'Maintenance des instruments scientifiques', 'CDD');

ID_INGENIEUR	SECTEUR_GESTION	TYPE_CONT
1	Gestion des équipements de laboratoire	CDI
2	Maintenance des instruments scientifiques	CDD

Table ENSEIGNANT_CHERCHEUR :

INSERT INTO Enseignant_chercheur (ID_ensg,

etablissement, adresse_etablissement, nb_heures)

VALUES (1, 'Institut de Recherche Médicale', '456 Avenue de la Recherche', 15);

```
INSERT INTO Enseignant_chercheur (ID_ensg,  
etablissement, adresse_etablissement, nb_heures)  
VALUES (2, 'Faculté des Sciences Appliquées', '789 Boulevard Scientifique', 25);
```

ID_ENSG	ETABLISSEMENT	ADRESSE_ETABLISSEMENT	NB_HEURES
1	Institut de Recherche Médicale	456 Avenue de la Recherche	15
2	Faculté des Sciences Appliquées	789 Boulevard Scientifique	25

Table DOCTORANT :

```
INSERT INTO Doctorant (ID_doctorant, sujet_these, date_debut, date_fin) VALUES  
(1, 'Impact des mutations sur la croissance cellulaire', TO_DATE('2024-03-10',  
'YYYY-MM-DD'), TO_DATE('2027-03-10', 'YYYY-MM-DD'));
```

ID_DOCTORANT	SUJET_THESE	DATE_DEBUT	DATE_FIN
1	Impact des mutations sur la croissance cellulaire	10-MAR-24	10-MAR-27

Table PEA :

```
INSERT INTO PEA (Numero_PEA, date_pea, mention, observation)  
VALUES (1, TO_DATE('2023-11-09', 'YYYY-MM-DD'), 'Réussi',  
'Aucune complication détectée');
```

```
INSERT INTO PEA (Numero_PEA, date_pea, mention, observation)  
VALUES (3, TO_DATE('2024-02-11', 'YYYY-MM-DD'), 'Échec',  
'Difficultés techniques');
```

```
INSERT INTO PEA (Numero_PEA, date_pea, mention, observation)  
VALUES (2, TO_DATE('2023-08-15', 'YYYY-MM-DD'), 'En cours', 'Observation  
en cours évaluation');
```

NUMERO_PEA	DATE_PEA	MENTION	OBSERVATION
1	09-NOV-23	Réussi	Aucune complication détectée
3	11-FEB-24	Échec	Difficultés techniques
2	15-AUG-23	En cours	Observation en cours évaluation

Table ANIMALERIE :

INSERT INTO Animalerie (ID_anim, num_cage, nb_souris, genre_souris, Numero_PEA, ID_lab)
VALUES (1, 15, 150, 'MALES', 1, 1);

INSERT INTO Animalerie (ID_anim, num_cage, nb_souris, genre_souris, Numero_PEA, ID_lab)
VALUES (2, 8, 60, 'FEMELLES', 2, 2);

INSERT INTO Animalerie (ID_anim, num_cage, nb_souris, genre_souris, Numero_PEA, ID_lab)
VALUES (3, 45, 300, 'MALES', 3, 3);

ID_ANIM	NUM_CAGE	NB_SOURIS	GENRE_SOURIS	NUMERO_PEA	ID_LAB
1	15	150	MALES	1	1
2	8	60	FEMELLES	2	2
3	45	300	MALES	3	3

Table FOURNISSEUR :

```
INSERT INTO fournisseur (ID_fournisseur, nom_fournisseur,  
num_telephone, e_mail) VALUES  
(1, 'ServicepourLabo', '+33658100923', 'ServicepourLabo@outlook.com');
```

```
INSERT INTO fournisseur (ID_fournisseur, nom_fournisseur,  
num_telephone, e_mail) VALUES  
(2, 'TousProduitsFourni', '+338729307190', 'TouProduitsFourni@gmail.com');
```

```
INSERT INTO fournisseur (ID_fournisseur, nom_fournisseur,  
num_telephone, e_mail) VALUES  
(3, 'FourniToutFrance', '+33975792028', 'FourniTouFran06@yahoo.fr');
```

```
INSERT INTO fournisseur (ID_fournisseur, nom_fournisseur,  
num_telephone, e_mail) VALUES  
(4, 'BestProduitsFourni', '+33498400280', 'BestProduitsFourni07@yahoo.fr');
```

```
INSERT INTO fournisseur (ID_fournisseur, nom_fournisseur,  
num_telephone, e_mail) VALUES  
(5, 'FournisseurChimie', '+33798493098', 'Fournichimie05@gmail.com')
```

Table FOURNISSEUR

ID_FOURNISSEUR	NOM_FOURNISSEUR	NUM_TELEPHONE	E_MAIL
1	ServicepourLabo	33658100923	ServicepourLabo@outlook.com
2	TousProduitsFourni	338729307190	TouProduitsFourni@gmail.com
3	FourniToutFrance	33975792028	FourniTouFran06@yahoo.fr
4	BestProduitsFourni	33498400280	BestProduitsFourni07@yahoo.fr
5	FournisseurChimie	33798493098	Fournichimie05@gmail.com

Table S1_culture_cellulaire :

```
INSERT INTO s1_culture_cellulaire (ID_s1, concentration) VALUES (1, 20);
```

```
INSERT INTO s1_culture_cellulaire (ID_s1, concentration) VALUES (0, NULL);
```

Table S1

ID_S1	CONCENTRATION
1	20
0	-

Table S2_prod_chimique :

```
INSERT INTO s2_prod_chimique (ID_s2, concentration,  
contre_indications, température) VALUES  
(2, 60, 'risque d"inflammation ', 38);
```

```
INSERT INTO s2_prod_chimique (ID_s2, concentration,  
contre_indications, température) VALUES  
(0, NULL, NULL, NULL);
```

Table S2

ID_S2	CONCENTRATION	CONTRE_INDICATIONS	TEMPÉRATURE
2	60	risque d"inflammation	38
0	-	-	-

Table S3_consommable :

```
INSERT INTO s3_consommable (ID_s3, contenance, Taille, matiere) VALUES  
(1, 30, NULL, 'verre');
```

```
INSERT INTO s3_consommable (ID_s3, contenance, Taille, matiere) VALUES  
(2, NULL, 'M', 'Plastique');
```

```
INSERT INTO s3_consommable (ID_s3, contenance, Taille, matiere)  
VALUES (0, NULL, NULL, NULL);
```

Table S3

ID_S3	CONTENANCE	TAILLE	MATIERE
1	30	-	verre
2	-	M	Plastique
0	-	-	-

Table S4_bio_moleculaire :

```
INSERT INTO s4_bio_moleculaire (ID_s4) VALUES (0);
```

```
INSERT INTO s4_bio_moleculaire (ID_s4) VALUES (1);
```

S4

ID_S4
0
1

Table S5_enzymes :

```
INSERT INTO s5_enzymes (ID_s5, concentration, temperature,  
nom_tampon, num_container, num_boite, num_bac) VALUES  
(5, 77, 40, 'NEBuffer', 4, 3, 16);
```

```
INSERT INTO s5_enzymes (ID_s5, concentration, temperature,  
nom_tampon, num_container, num_boite, num_bac) VALUES (0, NULL,  
NULL, NULL, NULL, NULL, NULL);
```

Table S5

ID_S5	CONCENTRATION	TEMPERATURE	NOM_TAMPON	NUM_CONTAINER	NUM_BOITE	NUM_BAC
5	77	40	NEBuffer	4	3	16
0	-	-	-	-	-	-

Table EXPERIENCE:

INSERT INTO Experience (ID_exp, but, duree_preparation, duree_incubation, resultat) VALUES (1, 'Verification de la souche de bactérie A ', 15, 33, 'Non existence ou non développement ');

INSERT INTO Experience (ID_exp, but, duree_preparation, duree_incubation, resultat) VALUES (2, ' chlore mélangé à de alcool', 34, 66, 'Mélange possible');

INSERT INTO Experience (ID_exp, but, duree_preparation, duree_incubation, resultat) VALUES (3, 'La bettrave contient elle du collagène ', 25, 47, 'Résultat positive ');

INSERT INTO Experience (ID_exp, but, duree_preparation, duree_incubation, resultat) VALUES (4, 'Les enzymes survivront-t-ils sous 15 degres?', 23, 56, 'Resultat négatif');

INSERT INTO Experience (ID_exp, but, duree_preparation, duree_incubation, resultat) VALUES (5, 'Regenerer les cellules mortes ', 46, 87, 'Non abouti');

Table EXPERIENCE

ID_EXP	BUT	DUREE_PREPARATION	DUREE_INCUBATION	RESULTAT
1	Verification de la souche de bactérie A	15	33	Non existence ou non développement
2	chlore mélangé à de alcool	34	66	Mélange possible
3	La bettrave contient elle du collagène	25	47	Résultat positive
4	Les enzymes survivront-t-ils sous 15 degres?	23	56	Resultat négatif
5	Regenerer les cellules mortes	46	87	Non abouti

Table COMMANDE :

INSERT INTO commande (ID_commande, date_commade, date_livraison, ID_fournisseur, ID_lab) VALUES (1, TO_DATE('2023-04-10', 'YYYY-MM-DD'), TO_DATE('2023-04-20', 'YYYY-MM-DD') , 1, 1);

INSERT INTO commande (ID_commande, date_commade, date_livraison, ID_fournisseur, ID_lab) VALUES (2, TO_DATE('2023-10-20', 'YYYY-MM-DD'), TO_DATE('2023-10-30', 'YYYY-MM-DD') , 1, 2);

```
INSERT INTO commande (ID_commande, date_commande,  
date_livraison, ID_fournisseur, ID_lab) VALUES  
(3, TO_DATE('2019-10-05', 'YYYY-MM-DD'), TO_DATE('2020-11-04', 'YYYY-  
MM-DD'), 4, 3);
```

```
INSERT INTO commande (ID_commande, date_commande,  
date_livraison, ID_fournisseur, ID_lab) VALUES  
(4, TO_DATE('2023-04-04', 'YYYY-MM-DD'), TO_DATE('2023-05-05', 'YYYY-  
MM-DD'), 1, 1);
```

```
INSERT INTO commande (ID_commande, date_commande,  
date_livraison, ID_fournisseur, ID_lab) VALUES  
(5, TO_DATE('2022-07-10', 'YYYY-MM-DD'), TO_DATE('2022-08-30',  
'YYYY-MM-DD'), 3, 2);
```

```
INSERT INTO commande (ID_commande, date_commande,  
date_livraison, ID_fournisseur, ID_lab) VALUES  
(8, TO_DATE('2022-07-11', 'YYYY-MM-DD'), TO_DATE('2022-08-31',  
'YYYY-MM-DD'), 2, 3);
```

ID_COMMANDE	DATE_COMMANDE	DATE_LIVRAISON	ID_FOURNISSEUR	ID_LAB
1	10-APR-23	20-APR-23	1	1
2	20-OCT-23	30-OCT-23	1	2
3	05-OCT-19	04-NOV-20	4	3
4	04-APR-23	05-MAY-23	1	1
5	10-JUL-22	30-AUG-22	3	2

Table CONTACTER :

INSERT INTO contacter (ID_lab, ID_fournisseur) VALUES (1, 2);

INSERT INTO contacter (ID_lab, ID_fournisseur) VALUES (3, 4);

INSERT INTO contacter (ID_lab, ID_fournisseur) VALUES (2, 2);

INSERT INTO contacter (ID_lab, ID_fournisseur) VALUES (3, 5);

Table CONTACTER

ID_LAB	ID_FOURNISSEUR
1	2
2	2
3	4
3	5

Table PROJET :

```
INSERT INTO Projet (ID_projet, sujet, date_debut, nb_publication, statut,  
ID_exp)
```

```
VALUES (1, 'Recherche sur le cancer', TO_DATE('2023-11-30', 'YYYY-MM-DD'),  
5, 'en cours', 1);
```

```
INSERT INTO Projet (ID_projet, sujet, date_debut, nb_publication, statut, ID_exp)  
VALUES (2, 'Développement de médicaments', TO_DATE('2022-11-01', 'YYYY-  
MM-DD'), 10, 'en cours - pas finalisé', 2);
```

```
INSERT INTO Projet (ID_projet, sujet, date_debut, nb_publication, statut,  
ID_exp) VALUES (3, 'Développement d'un vaccin contre la grippe',  
TO_DATE('2024-01-07', 'YYYY-MM-DD'), 3, 'En état de lancement', 3);
```

ID_PROJET	SUJET	DATE_DEBUT	NB_PUBLICATION	STATUT	ID_EXP
1	Recherche sur le cancer	30-NOV-23	5	en cours	1
2	Développement de médicaments	01-NOV-22	10	en cours - pas finalisé	2
3	Développement d'un vaccin contre la grippe	07-JAN-24	3	En état de lancement	3

Table EQUIPE :

```
INSERT INTO Equipe (ID_equipe, nb_chercheur, ID_projet)  
VALUES (1, 2, 1);
```

```
INSERT INTO Equipe (ID_equipe, nb_chercheur, ID_projet)  
VALUES (2, 1, 1);
```

```
INSERT INTO Equipe (ID_equipe, nb_chercheur, ID_projet)  
VALUES (3, 2, 2);
```

```
INSERT INTO Equipe (ID_equipe, nb_chercheur, ID_projet)  
VALUES (4, 2, 3);
```

ID_EQUIPE	NB_CHERCHEUR	ID_PROJET
1	2	1
2	1	1
3	2	2
4	2	3

Table CHERCHEUR :

```
INSERT INTO Chercheur (ID_chercheur, nom, prenom, age,  
ID_equipe, ID_tache, ID_engs, ID_doctorant, ID_ingenieur, ID_lab)  
VALUES (1, 'Dupoint', 'Marcel', 35, 1, 1, NULL, NULL, NULL, 1);
```

```
INSERT INTO Chercheur (ID_chercheur, nom, prenom, age,  
ID_equipe, ID_tache, ID_engs, ID_doctorant, ID_ingenieur, ID_lab)  
VALUES (2, 'Duval', 'Jean-Philippe', 37, 1, 2, NULL, 1, NULL, 1);
```

```
INSERT INTO Chercheur (ID_chercheur, nom, prenom, age,
ID_equipe, ID_tache, ID_ensg, ID_doctorant, ID_ingenieur, ID_lab)
VALUES (3, 'Legrand', 'Armand', 30, 2, 3, NULL, NULL, 1, 1);
```

```
INSERT INTO Chercheur (ID_chercheur, nom, prenom, age,
ID_equipe, ID_tache, ID_ensg, ID_doctorant, ID_ingenieur, ID_lab)
VALUES (4, 'Ruban', 'Frederique', 29, 3, 4, 1, NULL, NULL, 2);
```

```
INSERT INTO Chercheur (ID_chercheur, nom, prenom, age,
ID_equipe, ID_tache, ID_ensg, ID_doctorant, ID_ingenieur, ID_lab)
VALUES (5, 'Françoise', 'Marie', 32, 3, 5, NULL, NULL, 2, 2);
```

```
INSERT INTO Chercheur (ID_chercheur, nom, prenom, age,
ID_equipe, ID_tache, ID_ensg, ID_doctorant, ID_ingenieur, ID_lab)
VALUES (6, 'Martin', 'Robert', 40, 4, 0, 2, NULL, NULL, 3);
```

```
INSERT INTO Chercheur (ID_chercheur, nom, prenom, age,
ID_equipe, ID_tache, ID_ensg, ID_doctorant, ID_ingenieur, ID_lab)
VALUES (7, 'Hubert', 'David', 35, 4, 0, NULL, NULL, NULL, 3);
```

ID_CHERCHEUR	NOM	PRENOM	AGE	ID_EQUIPE	ID_TACHE	ID_ENSG	ID_DOCTORANT	ID_INGENIEUR	ID_LAB
1	Dupont	Marcel	35	1	1	-	-	-	1
2	Duval	Jean-Philippe	37	1	2	-	1	-	1
3	Legrand	Armand	30	2	3	-	-	1	1
4	Ruban	Frederique	29	3	4	1	-	-	2
5	Françoise	Marie	32	3	5	-	-	2	2
6	Martin	Robert	40	4	0	2	-	-	3
7	Hubert	David	35	4	0	-	-	-	3

Table ETRE_COMPOSEE :

```
INSERT INTO etre_composee (ID_chercheur, ID_equipe)
VALUES (3, 2);
```

```
INSERT INTO etre_composee (ID_chercheur, ID_equipe)
VALUES (4, 3);
```

```
INSERT INTO etre_composee (ID_chercheur, ID_equipe)
VALUES (5, 3);
```

```
INSERT INTO etre_composee (ID_chercheur, ID_equipe)
VALUES (6, 4);
```

```
INSERT INTO etre_composee (ID_chercheur, ID_equipe)  
VALUES (7, 4);
```

```
INSERT INTO etre_composee (ID_chercheur, ID_equipe)  
VALUES (2, 2);
```

Table ETRE_COMPOSEE

ID_CHERCHEUR	ID_EQUIPE
3	2
4	3
5	3
6	4
7	4

Table SE_PRESENTER :

```
INSERT INTO Se_Presenter (ID_chercheur, ID_reunion)  
VALUES (1, 1);
```

```
INSERT INTO Se_Presenter (ID_chercheur, ID_reunion)  
VALUES (2, 1);
```

```
INSERT INTO Se_Presenter (ID_chercheur, ID_reunion)  
VALUES (3, 1);
```

```
INSERT INTO Se_Presenter (ID_chercheur, ID_reunion)  
VALUES (1, 2);
```

```
INSERT INTO Se_Presenter (ID_chercheur, ID_reunion)  
VALUES (2, 2);
```

```
INSERT INTO Se_Presenter (ID_chercheur, ID_reunion)  
VALUES (3, 2);
```

```
INSERT INTO Se_Presenter (ID_chercheur, ID_reunion)  
VALUES (4, 3);
```

```
INSERT INTO Se_Presenter (ID_chercheur, ID_reunion)  
VALUES (5, 3);
```

```
INSERT INTO Se_Presenter (ID_chercheur, ID_reunion)  
VALUES (6, 4);
```

```
INSERT INTO Se_Presenter (ID_chercheur, ID_reunion)  
VALUES (7, 4);
```

```
INSERT INTO Se_Presenter (ID_chercheur, ID_reunion)  
VALUES (6, 5);
```

```
INSERT INTO Se_Presenter (ID_chercheur, ID_reunion)  
VALUES (7, 5);
```

ID_CHERCHEUR	ID_REUNION
1	1
1	2
2	1
2	2
3	1
3	2
4	3
5	3
6	4
6	5
7	4
7	5

Table PRODUIT :

```
INSERT INTO produit (ID_produit, stock, localisation, nom, ID_s1, ID_s2, ID_s3,  
ID_s4, ID_s5, ID_lab, ID_chercheur) VALUES  
(1, 14, 'rayon A', 'Acide Aminé ', 1, 0, 0, 0, 0, 1, 2);
```

```
INSERT INTO produit (ID_produit, stock, localisation, nom, ID_s1, ID_s2, ID_s3,  
ID_s4, ID_s5, ID_lab, ID_chercheur) VALUES (2, 15, 'rayon B', 'Reactif HCL', 0,  
2, 0, 0, 0, 1, 3);
```

```
INSERT INTO Produit (ID_produit, stock, localisation, nom, ID_s1, ID_s2,  
ID_s3, ID_s4, ID_s5, ID_lab, ID_chercheur)  
VALUES (3, 7, 'Rayon B', 'Acide sulfurique', 0, 2, 0, 0, 0, 1, 3);
```

```
INSERT INTO Produit (ID_produit, stock, localisation, nom, ID_s1, ID_s2,  
ID_s3, ID_s4, ID_s5, ID_lab, ID_chercheur)  
VALUES (4, 15, 'Rayon F', 'Enzyme ZN', 0, 0, 0, 0, 5, 3, 6);
```

```
INSERT INTO Produit (ID_produit, stock, localisation, nom, ID_s1, ID_s2,  
ID_s3, ID_s4, ID_s5, ID_lab, ID_chercheur)  
VALUES (5, 30, 'Rayon B', 'Réactif H3', 0, 2, 0, 0, 0, 2, 5);
```

```
INSERT INTO Produit (ID_produit, stock, localisation, nom, ID_s1, ID_s2,  
ID_s3, ID_s4, ID_s5, ID_lab, ID_chercheur)
```

Timofte B & Bouchhioua W
INFO0505 - RAPPORT

```
VALUES (6, 12, 'Rayon C', 'Milieu de culture', 1, 0, 0, 0, 0, 3, 7);
```

```
INSERT INTO Produit (ID_produit, stock, localisation, nom, ID_s1, ID_s2,  
ID_s3, ID_s4, ID_s5, ID_lab, ID_chercheur)  
VALUES (7, 25, 'Rayon A', 'Liquide de lavage', 0, 0, 2, 0, 0, 2, 4);
```

```
INSERT INTO Produit (ID_produit, stock, localisation, nom, ID_s1, ID_s2,  
ID_s3, ID_s4, ID_s5, ID_lab, ID_chercheur)  
VALUES (9, 22, 'Rayon G', 'Réactif B', 0, 0, 0, 1, 0, 1, 2);
```

```
INSERT INTO Produit (ID_produit, stock, localisation, nom, ID_s1, ID_s2,  
ID_s3, ID_s4, ID_s5, ID_lab, ID_chercheur)  
VALUES (10, 10, 'Rayon A', 'Milieu sélectif', 0, 0, 0, 1, 0, 2, 5);
```

```
INSERT INTO Produit (ID_produit, stock, localisation, nom, ID_s1, ID_s2,  
ID_s3, ID_s4, ID_s5, ID_lab, ID_chercheur)  
VALUES (11, 28, 'Rayon F', 'Lysat cellulaire', 1, 0, 0, 0, 0, 3, 6);
```

```
INSERT INTO Produit (ID_produit, stock, localisation, nom, ID_s1, ID_s2,  
ID_s3, ID_s4, ID_s5, ID_lab, ID_chercheur)  
VALUES (12, 15, 'Rayon C', 'Amplificateur ADN', 0, 2, 0, 0, 0, 1, 1);
```

```
INSERT INTO Produit (ID_produit, stock, localisation, nom, ID_s1, ID_s2,  
ID_s3, ID_s4, ID_s5, ID_lab, ID_chercheur)  
VALUES (13, 20, 'Rayon E', 'Solution tampon', 0, 2, 0, 0, 0, 2, 5);
```

```
INSERT INTO Produit (ID_produit, stock, localisation, nom, ID_s1, ID_s2,  
ID_s3, ID_s4, ID_s5, ID_lab, ID_chercheur)  
VALUES (14, 200, 'Rayon E', 'Gantes médicales', 0, 0, 2, 0, 0, 3, 7);
```

```
INSERT INTO Produit (ID_produit, stock, localisation, nom, ID_s1, ID_s2, ID_s3,  
ID_s4, ID_s5, ID_lab, ID_chercheur)  
VALUES (15, 230, 'Rayon K', 'CO2', 0, 2, 0, 0, 0, 3, 7)
```

Timofte B & Bouchhioua W
INFO0505 - RAPPORT

ID_PRODUIT	STOCK	LOCALISATION	NOM	ID_S1	ID_S2	ID_S3	ID_S4	ID_S5	ID_LAB	ID_CHERCHEUR
2	15	rayon B	Reactif HCL	0	2	0	0	0	1	3
3	7	Rayon B	Acide sulfurique	0	2	0	0	0	1	3
1	14	rayon A	Acide Aminé	1	0	0	0	0	1	2
4	15	Rayon F	Enzyme ZN	0	0	0	0	5	3	6
5	30	Rayon B	Réactif H3	0	2	0	0	0	2	5
6	12	Rayon C	Milieu de culture	1	0	0	0	0	3	7
7	25	Rayon A	Liquide de lavage	0	0	2	0	0	2	4
9	22	Rayon G	Réactif B	0	0	0	1	0	1	2
10	10	Rayon A	Milieu sélectif	0	0	0	1	0	2	5
11	28	Rayon F	Lysat cellulaire	1	0	0	0	0	3	6
12	15	Rayon C	Amplificateur ADN	0	2	0	0	0	1	1
13	20	Rayon E	Solution tampon	0	2	0	0	0	2	5
14	200	Rayon E	Gantes médicales	0	0	2	0	0	3	7

Table SE_SITUER :

INSERT INTO se_situer (ID_adresse, ID_fournisseur) VALUES (4, 1);

INSERT INTO se_situer (ID_adresse, ID_fournisseur) VALUES (5, 2);

INSERT INTO se_situer (ID_adresse, ID_fournisseur) VALUES (6, 3);

INSERT INTO se_situer (ID_adresse, ID_fournisseur) VALUES (7, 4);

INSERT INTO se_situer (ID_adresse, ID_fournisseur) VALUES (8, 5);

Timofte B & Bouchhioua W
INFO0505 - RAPPORT

ID_ADRESSE	ID_FOURNISSEUR
4	1
5	2
6	3
7	4
8	5

Table RSERVATION :

INSERT INTO Reservation_ (ID_rsv, date_reservation, duree,
ID_lab, ID_chercheur)

Timofte B & Bouchhioua W
INFO0505 - RAPPORT

```
VALUES (1, TO_TIMESTAMP('2023-06-05 02:00:00', 'YYYY-MM-DD HH24:MI:SS'), 2, 1, 1);
```

```
INSERT INTO Reservation_ (ID_rsv, date_reservation, duree, ID_lab, ID_chercheur)  
VALUES (2, TO_TIMESTAMP('2023-06-12 02:30:00', 'YYYY-MM-DD HH24:MI:SS'), 2, 1, 2);
```

```
INSERT INTO Reservation_ (ID_rsv, date_reservation, duree, ID_lab, ID_chercheur)  
VALUES (3, TO_TIMESTAMP('2023-07-14 01:30:00', 'YYYY-MM-DD HH24:MI:SS'), 1, 1, 3);
```

```
INSERT INTO Reservation_ (ID_rsv, date_reservation, duree, ID_lab, ID_chercheur)  
VALUES (4, TO_TIMESTAMP('2023-11-13 02:15:00', 'YYYY-MM-DD HH24:MI:SS'), 2, 2, 4);
```

```
INSERT INTO Reservation_ (ID_rsv, date_reservation, duree, ID_lab, ID_chercheur)  
VALUES (5, TO_TIMESTAMP('2023-12-06 03:30:00', 'YYYY-MM-DD HH24:MI:SS'), 3, 3, 5);
```

Table RESERVATION_

ID_RSV	DATE_RESERVATION	DUREE	ID_LAB	ID_CHERCHEUR
1	05-JUN-23	2	1	1
2	12-JUN-23	2	1	2
3	14-JUL-23	1	1	3
4	13-NOV-23	2	2	4
5	06-DEC-23	3	3	5

Table UTILISER_MATERIEL :

```
INSERT INTO utiliser_materiel (ID_produit, ID_rsv)  
VALUES (1, 1);
```


Timofte B & Bouchhioua W
INFO0505 - RAPPORT

```
INSERT INTO utiliser_materiel (ID_produit, ID_rsv)  
VALUES (2, 2);
```

```
INSERT INTO utiliser_materiel (ID_produit, ID_rsv)  
VALUES (3, 3);
```

```
INSERT INTO utiliser_materiel (ID_produit, ID_rsv)  
VALUES (4, 4);
```

ID_PRODUIT	ID_RSV
1	1
2	2
3	3
4	4

Table UTILISER_EXPERIENCE :

```
INSERT INTO utilis_experience (ID_exp, ID_produit)  
VALUES (1, 1);
```

```
INSERT INTO utilis_experience (ID_exp, ID_produit)  
VALUES (1, 2);
```

```
INSERT INTO utilis_experience (ID_exp, ID_produit)  
VALUES (2, 2);
```

```
INSERT INTO utilis_experience (ID_exp, ID_produit)  
VALUES (2, 3);
```

```
INSERT INTO utilis_experience (ID_exp, ID_produit)  
VALUES (3, 4);
```

```
INSERT INTO utilis_experience (ID_exp, ID_produit)  
VALUES (3, 5);
```

ID_EXP	ID_PRODUIT
1	1
1	2
2	2
2	3
3	4
3	5

Table PROJET_EXPERIENCE :

```
INSERT INTO Projet_Experience (ID_Projet, ID_exp)  
VALUES (1, 1);
```

```
INSERT INTO Projet_Experience (ID_Projet, ID_exp)  
VALUES (2, 2);
```

```
INSERT INTO Projet_Experience (ID_Projet, ID_exp)  
VALUES (3, 3);
```

ID_PROJET	ID_EXP
1	1
2	2
3	3

Table COMMANDER :

```
INSERT INTO commander (ID_produit, ID_commande, quantite) VALUES (1,  
1, 6);
```

```
INSERT INTO commander (ID_produit, ID_commande, quantite) VALUES  
(4,2,12);
```

```
INSERT INTO commander (ID_produit, ID_commande, quantite) VALUES  
(2,1,23);
```

```
INSERT INTO commander (ID_produit, ID_commande, quantite) VALUES  
(1,4,12);
```

```
INSERT INTO commander (ID_produit, ID_commande, quantite) VALUES  
(5,5,10);
```

ID_PRODUIT	ID_COMMANDE	QUANTITE
1	1	6
4	2	12
2	1	23
1	4	12
5	5	10

4. Partie SQL : Requêtes

1. Le nombre des chercheurs, ainsi que leur nom et leur prénom, qui sont des Doctorants qui font partie de l'équipe qui travaille sur un projet qui comporte une expérience qui utilise un produit qui est organisé dans le sous-ensemble des cultures cellulaires et qui est commandé lors d'une commande avec la date de commande entre le mois de mars 2023 et le mois de juin 2023 :




```
SELECT c.nom, c.prenom, COUNT(c.ID_chercheur) AS  
nombre_doctorants FROM Chercheur c  
WHERE c.ID_doctorant IS NOT NULL  
AND c.ID_equipe IN (  
    SELECT e.ID_equipe  
    FROM Equipe e  
    WHERE e.ID_projet IN (  
        SELECT DISTINCT p.ID_projet  
        FROM Projet p
```

Timofte B & Bouchhioua W
INFO0505 - RAPPORT

```
WHERE EXISTS (  
    SELECT 1  
    FROM COMMANDER cm  
    JOIN Produit pr ON cm.ID_produit = pr.ID_produit  
    JOIN utiliser_experience ue ON pr.ID_produit = ue.ID_produit  
    JOIN Commande co ON cm.ID_commande = co.ID_commande  
    JOIN S1_culture_cellulaire s1 ON pr.ID_s1 = s1.ID_s1  
    WHERE p.ID_projet = e.ID_projet  
    AND TO_CHAR(co.date_commade, 'YYYYMM') BETWEEN  
'202303' AND '202306'  
    )  
    )  
    )  
GROUP BY c.nom, c.prenom, c.ID_chercheur;
```


Le rendu sur ORACLE Live SQL de cette requête :

Statement 300



```
SELECT c.nom, c.prenom, COUNT(c.ID_chercheur) AS nombre_doctorants
FROM Chercheur c
WHERE c.ID_doctorant IS NOT NULL
AND c.ID_equipe IN (
  SELECT e.ID_equipe
  FROM Equipe e
  WHERE e.ID_projet IN (
    SELECT DISTINCT p.ID_projet
    FROM Projet p
    WHERE EXISTS (
      SELECT 1
      FROM COMMANDER cm
      JOIN Produit pr ON cm.ID_produit = pr.ID_produit
      JOIN utiliser_experience ue ON pr.ID_produit = ue.ID_produit
      JOIN Commande co ON cm.ID_commande = co.ID_commande
      JOIN S1_culture_cellulaire s1 ON pr.ID_s1 = s1.ID_s1
      WHERE p.ID_projet = e.ID_projet
      AND TO_CHAR(co.date_commade, 'YYYYMM') BETWEEN '202303' AND '202306'
    )
  )
)
GROUP BY c.nom, c.prenom, c.ID_chercheur
```

NOM	PRENOM	NOMBRE_DOCTORANTS
Duval	Jean-Philippe	1

2. Tous les chercheurs qui appartient au laboratoire qui possède une animalerie dont le nombre des souris est plus petit que 100 et cette animalerie est caractérisé par un numero PEA qui a comme mention 'réussi' et le laboratoire est identifié par une adresse qui est parmi les 5 premieres adresses en ordre alphabétique de leur nom de ville :

```
SELECT c.nom, c.prenom
FROM Chercheur c
WHERE c.ID_lab IN (
  SELECT l.ID_lab
  FROM Laboratoire l
  WHERE l.ID_lab IN (
    SELECT a.ID_lab
    FROM Animalerie a
    WHERE a.nb_souris < 100
    AND a.Numero_PEA IN (
      SELECT pea.Numero_PEA
      FROM PEA pea
      WHERE pea.mention = 'reussi'
    )
  )
)
AND l.ID_adresse IN (
  SELECT a.ID_adresse
  FROM Adresse a
  WHERE a.ID_adresse IN (
    SELECT ID_adresse
    FROM (
      SELECT ID_adresse
      FROM Adresse
      ORDER BY ville
    )
    WHERE ROWNUM <= 5
  )
)
);
```

3. La salle de la réunion à laquelle il se présente le chercheur qui est l'enseignant chercheur avec le plus grand nombre d'heures dans l'établissement et qui fait partie de l'équipe qui travaille sur le projet qui comporte l'expérience avec la plus longue durée de préparation :

```
SELECT r.salle
FROM Reunion r
WHERE r.ID_lab = (
    SELECT c.ID_lab
```

```
FROM Chercheur c
JOIN Enseignant_chercheur ec ON c.ID_ensg = ec.ID_ensg
WHERE ec.nb_heures = (
    SELECT MAX(ec.nb_heures)
    FROM Enseignant_chercheur ec
)
AND c.ID_equipe = (
    SELECT ID_equipe
    FROM Projet p
    JOIN Projet_Experience pe ON p.ID_projet = pe.ID_projet
    JOIN Experience e ON pe.ID_exp = e.ID_exp
    WHERE e.duree_preparation = (
        SELECT MAX(e.duree_preparation)
        FROM Experience e
    )
)
);
```

4. Le projet avec la plus grande équipe des chercheurs qui sont responsables d'une tâche dont la description ne dépasse pas 40 caractères, avec la date de début du projet et l'âge maximum des chercheurs qui travaillent sur ce projet :

```
SELECT
    p.sujet AS sujet_projet,
    p.date_debut AS date_debut_projet,
    COUNT(c.ID_chercheur) AS
    nombre_chercheurs, MAX(c.age) AS age_max
FROM
```

```
Projet p
JOIN
  Equipe e ON p.ID_projet = e.ID_projet
JOIN
  Chercheur c ON e.ID_equipe = c.ID_equipe
JOIN
  Tache t ON c.ID_tache = t.ID_tache
WHERE
  LENGTH(t.description_tache) <= 40
GROUP BY
  p.sujet, p.date_debut
ORDER BY
  COUNT(c.ID_chercheur) DESC
FETCH FIRST 1 ROW ONLY;
```

Le rendu sur ORACLE Live SQL de cette requête :

Statement 306



```
SELECT
  p.sujet AS sujet_projet,
  p.date_debut AS date_debut_projet,
  COUNT(c.ID_chercheur) AS nombre_chercheurs,
  MAX(c.age) AS age_max
FROM
  Projet p
JOIN
  Equipe e ON p.ID_projet = e.ID_projet
JOIN
  Chercheur c ON e.ID_equipe = c.ID_equipe
JOIN
  Tache t ON c.ID_tache = t.ID_tache
WHERE
  LENGTH(t.description_tache) <= 40
GROUP BY
  p.sujet, p.date_debut
ORDER BY
  COUNT(c.ID_chercheur) DESC
FETCH FIRST 1 ROW ONLY
```

SUJET_PROJET	DATE_DEBUT_PROJET	NOMBRE_CHERCHEURS	AGE_MAX
Recherche sur le cancer	30-NOV-23	3	37

5. Le nom et l'adresse des fournisseurs qui fournissent plus de commandes que la moyenne de commandes par fournisseurs.

```
SELECT f.nom_fournisseur,a.ville
FROM Fournisseur f

JOIN
    se_situer s ON f.ID_fournisseur = s.ID_fournisseur
JOIN
    Adresse a ON s.ID_adresse = a.ID_adresse
WHERE f.ID_fournisseur IN

    (SELECT ID_fournisseur FROM (
        (SELECT ID_fournisseur, Count(*) AS Nbc FROM
            Commande GROUP BY ID_fournisseur HAVING COUNT(*) >
                (SELECT AVG(Nb) FROM (
                    SELECT ID_fournisseur, Count(*) AS Nb FROM
                        Commande
                        GROUP BY ID_fournisseur ) ) ) ) )
```

```
SELECT f.nom_fournisseur,a.ville FROM Fournisseur f
JOIN se_situer s ON f.ID_fournisseur = s.ID_fournisseur
JOIN Adresse a ON s.ID_adresse = a.ID_adresse
WHERE f.ID_fournisseur IN
    (SELECT ID_fournisseur FROM (
        (SELECT ID_fournisseur, Count(*) AS Nbc FROM Commande GROUP BY ID_fournisseur HAVING COUNT(*) >
            (SELECT AVG(Nb) FROM (
                SELECT ID_fournisseur, Count(*) AS Nb FROM Commande GROUP BY ID_fournisseur ) ) ) ) )
```

NOM_FOURNISSEUR	VILLE
ServicepourLabo	Lyon

6. La localisation des produits appartenant aux sous ensemble S2_produits chimiques fournit par des fournisseurs situés à Paris utilisé par le projet dont la date de début > '2018-01-01'

```
SELECT p.localisation FROM Produit p
WHERE p.ID_s2 <> 0 AND p.ID_produit IN
    (SELECT c.ID_produit FROM commander c
        WHERE c.ID_commande IN

            (SELECT cmd.ID_commande FROM Commande cmd
                WHERE cmd.ID_fournisseur IN

                    ( SELECT f.ID_fournisseur FROM Fournisseur f
```

```

WHERE f.ID_fournisseur IN

        ( SELECT s.ID_fournisseur FROM se_situer s
          WHERE s.ID_adresse IN
            (SELECT a.ID_adresse FROM Adresse a
             WHERE a.ville ='Paris')
          )
      )
  )
)
AND p.ID_produit IN
  (SELECT u.ID_produit FROM utiliser_experience u
   WHERE u.ID_exp IN

     (SELECT e.ID_exp FROM Experience e
      WHERE e.ID_exp IN
        (SELECT pj.ID_exp FROM Projet pj
         WHERE
           TO_CHAR (pj.date_debut,'YYYYMMDD')>'2018-01-01')
        )
     )
  )

```

```

1 ✓ SELECT p.localisation FROM Produit p WHERE p.ID_s2 <> 0
2 AND p.ID_produit IN
3   (SELECT c.ID_produit FROM commander c WHERE c.ID_commande IN
4     (SELECT cmd.ID_commande FROM Commande cmd WHERE cmd.ID_fournisseur IN
5       ( SELECT f.ID_fournisseur FROM Fournisseur f WHERE f.ID_fournisseur IN
6         ( SELECT s.ID_fournisseur FROM se_situer s WHERE s.ID_adresse IN
7           (SELECT a.ID_adresse FROM Adresse a WHERE a.ville ='Paris')
8         )
9       )
10    )
11  )
12 AND p.ID_produit IN
13   (SELECT u.ID_produit FROM utiliser_experience u WHERE u.ID_exp IN
14     (SELECT e.ID_exp FROM Experience e WHERE e.ID_exp IN
15       (SELECT pj.ID_exp FROM Projet pj WHERE TO_CHAR(pj.date_debut, 'YYYYMMDD')>'2018-01-01')
16     )
17   )

```

LOCALISATION

Rayon K

7. La concentration des produits appartenant au sous-ensembles S5_enzymes dont la température est supérieure à 20 commandé par le laboratoire qui contient moins de 5 chercheurs.

```

SELECT s5.concentration FROM S5_enzymes s5
WHERE s5.temperature> 20 AND s5.ID_s5 IN

```

```

  (SELECT p.ID_s5 FROM Produit p
   WHERE p.ID_produit IN
     (SELECT c.ID_produit FROM commander c
      WHERE c.ID_commande IN

```

```
SELECT s5.concentration FROM S5_enzymes s5 WHERE s5.temperature> 20 AND s5.ID_s5 IN
  (SELECT p.ID_s5 FROM Produit p WHERE p.ID_produit IN
    (SELECT c.ID_produit FROM commander c WHERE c.ID_commande IN
      (SELECT cmd.ID_commande FROM Commande cmd WHERE cmd.ID_lab IN
        (SELECT l.ID_lab FROM Laboratoire l WHERE l.nb_chercheurs<5)
      )
    )
  )
)
```

CONCENTRATION
77
Download CSV

```
(SELECT cmd.ID_commande FROM Commande
cmd
WHERE cmd.ID_lab IN
  (SELECT l.ID_lab FROM Laboratoire l
WHERE l.nb_chercheurs<5)
)
)
)
```

8. Les produits qui ont été utilisé dans toutes les expériences dont le projet est dirigé par une équipe dont le responsable est doctorant. (DIVISION)

```
SELECT DISTINCT p.nom
FROM Produit p
JOIN Utiliser_Experience ue ON p.ID_produit = ue.ID_produit
JOIN Experience e ON ue.ID_exp = e.ID_exp
JOIN Projet proj ON e.ID_exp = proj.ID_exp
JOIN Equipe eq ON proj.ID_projet = eq.ID_projet
JOIN etre_composee ec ON eq.ID_equipe =ec.ID_equipe
JOIN Chercheur doc ON ec.ID_chercheur = doc.ID_chercheur AND doc.ID_doctorant IS NOT NULL
WHERE NOT EXISTS(
  SELECT ue.ID_exp
```



```
FROM Utiliser_Experience ue

WHERE ue.ID_produit = p.ID_produit

AND NOT EXISTS (

    SELECT ue.ID_exp

    FROM Utiliser_Experience ue

    WHERE ue.ID_exp = e.ID_exp

)

)
```

```
SELECT DISTINCT p.nom
FROM Produit p
JOIN Utiliser_Experience ue ON p.ID_produit = ue.ID_produit
JOIN Experience e ON ue.ID_exp = e.ID_exp
JOIN Projet proj ON e.ID_exp = proj.ID_exp
JOIN Equipe eq ON proj.ID_projet = eq.ID_projet
JOIN etre_composee ec ON eq.ID_equipe = ec.ID_equipe
JOIN Chercheur doc ON ec.ID_chercheur = doc.ID_chercheur AND doc.ID_doctorant IS NOT NULL
WHERE NOT EXISTS(
    SELECT ue.ID_exp
    FROM Utiliser_Experience ue
    WHERE ue.ID_produit = p.ID_produit
    AND NOT EXISTS (
        SELECT ue.ID_exp
        FROM Utiliser_Experience ue
        WHERE ue.ID_exp = e.ID_exp
    )
)
```

NOM
Reactif HCL
Acide Aminé

5. Partie PL/SQL Programmes

1. Programme PL/SQL avec Procédure Stockée . Objectif : Mettre à jour le nombre de chercheurs dans une équipe donnée :

```
CREATE OR REPLACE PROCEDURE maj_nb_chercheurs_equipe(equipe_id
IN INT) AS
  v_nb_chercheurs INT;
BEGIN
  SELECT COUNT(*)
  INTO v_nb_chercheurs
  FROM Chercheur
  WHERE ID_equipe = equipe_id;

  UPDATE Equipe
  SET nb_chercheur = v_nb_chercheurs
  WHERE ID_equipe = equipe_id;

  COMMIT;
END maj_nb_chercheurs_equipe;
```

Pour exécuter la procédure stockée :

```
EXEC maj_nb_chercheurs_equipe(1)
```

Pour vérifier les résultats :

```
SELECT nb_chercheur
FROM Equipe
WHERE ID_equipe = 1;
```

2. Programme PL/SQL avec Trigger . Objectif : Mettre à jour le sujet
d'un projet lorsque le but de l'expérience associée est modifiée :

```
CREATE OR REPLACE TRIGGER maj_sujet_projet
BEFORE UPDATE OF but ON Experience
FOR EACH ROW
BEGIN
  IF :NEW.but <> :OLD.but THEN
    UPDATE Projet
    SET sujet = :NEW.but
    WHERE ID_exp = :NEW.ID_exp;
  END IF;
END maj_sujet_projet;
```

Pour utiliser le Trigger :

* Mise à jour de la colonne 'but' dans la table
'Experience' UPDATE Experience
SET but = 'Nouveau but'
WHERE ID_exp = 123; -- Remplacez 123 par l'ID_exp que vous souhaitez
mettre à jour

* Vérification des modifications dans la table 'Projet' :
SELECT sujet
FROM Projet
WHERE ID_exp = 123; -- Remplacez 123 par l'ID_exp que vous avez mis à jour

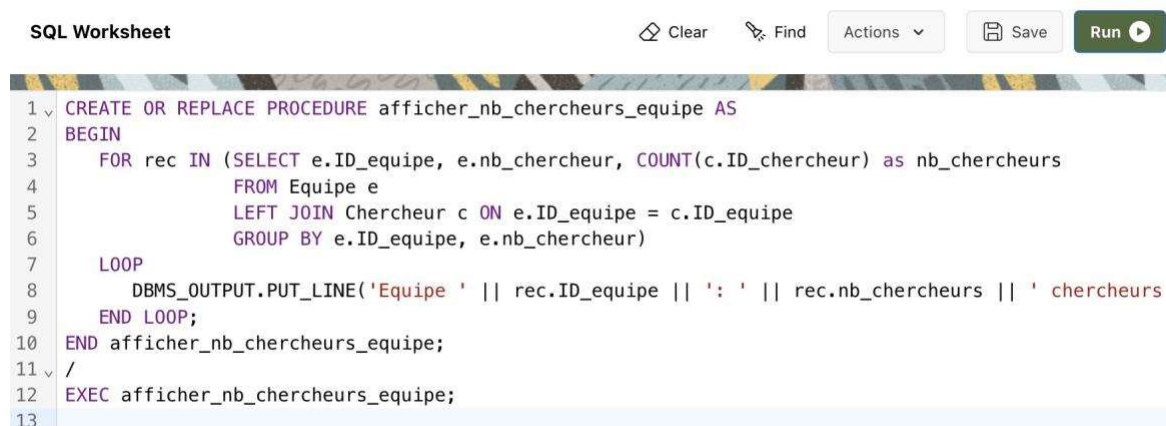
3. Programme PL/SQL avec Curseur. Objectif : Afficher le nombre de chercheurs dans chaque équipe :

```
CREATE OR REPLACE PROCEDURE afficher_nb_chercheurs_equipe
AS BEGIN
  FOR rec IN (SELECT e.ID_equipe, e.nb_chercheur, COUNT(c.ID_chercheur)
as nb_chercheurs
    FROM Equipe e
    LEFT JOIN Chercheur c ON e.ID_equipe =
    c.ID_equipe GROUP BY e.ID_equipe, e.nb_chercheur)
  LOOP
    DBMS_OUTPUT.PUT_LINE('Equipe ' || rec.ID_equipe || ': '
|| rec.nb_chercheurs || ' chercheurs sur ' || rec.nb_chercheur);
  END LOOP;
END afficher_nb_chercheurs_equipe;
```

Pour exécuter la procédure stockée :

```
EXEC afficher_nb_chercheurs_equipe;
```

Le résultat de cette procédure dans le SQL Worksheet :



The screenshot shows a SQL Worksheet interface with a toolbar at the top containing 'Clear', 'Find', 'Actions', 'Save', and 'Run' buttons. The code area displays the following SQL code:

```
1 CREATE OR REPLACE PROCEDURE afficher_nb_chercheurs_equipe AS
2 BEGIN
3   FOR rec IN (SELECT e.ID_equipe, e.nb_chercheur, COUNT(c.ID_chercheur) as nb_chercheurs
4               FROM Equipe e
5               LEFT JOIN Chercheur c ON e.ID_equipe = c.ID_equipe
6               GROUP BY e.ID_equipe, e.nb_chercheur)
7   LOOP
8     DBMS_OUTPUT.PUT_LINE('Equipe ' || rec.ID_equipe || ': ' || rec.nb_chercheurs || ' chercheurs
9   END LOOP;
10 END afficher_nb_chercheurs_equipe;
11 /
12 EXEC afficher_nb_chercheurs_equipe;
13
```

4.

Objectif :

Afficher les produits en
stock

```
CREATE OR REPLACE PROCEDURE afficher_produits_en_stock AS
CURSOR cur_produits_en_stock IS
SELECT nom, stock
FROM Produit
WHERE stock > 0;

nom_produit Produit.nom%TYPE;
stock_produit Produit.stock%TYPE;
BEGIN
OPEN cur_produits_en_stock;
LOOP
FETCH cur_produits_en_stock INTO nom_produit, stock_produit;
EXIT WHEN cur_produits_en_stock%NOTFOUND;
DBMS_OUTPUT.PUT_LINE('Produit : ' || nom_produit || ', Stock : ' ||
stock_produit);
END LOOP;
CLOSE cur_produits_en_stock;
END afficher_produits_en_stock;
```

```
1 EXEC afficher_produits_en_stock;
```

```
Statement processed.  
Produit : Acide Aminé , Stock : 14  
Produit : Réactif HCL, Stock : 15  
Produit : Acide sulfurique, Stock : 7  
Produit : Enzyme ZN, Stock : 15  
Produit : Réactif H3, Stock : 30  
Produit : Milieu de culture, Stock : 12  
Produit : Liquide de lavage, Stock : 25  
Produit : Réactif B, Stock : 22  
Produit : Milieu sélectif, Stock : 10  
Produit : Lysat cellulaire, Stock : 28  
Produit : Amplificateur ADN, Stock : 15  
Produit : Solution tampon, Stock : 20  
Produit : Gantes medicales, Stock : 200  
Produit : CO2, Stock : 230
```

5. Objectif : Trigger pour empêcher la suppression d'un chercheur ayant une réservation en cours

```
CREATE OR REPLACE TRIGGER empecher_suppression_chercheur  
BEFORE DELETE ON Chercheur  
FOR EACH ROW  
  DECLARE nb_reserv INT;  
BEGIN  
  SELECT COUNT(*) INTO nb_reserv  
  FROM Reservation_  
  WHERE ID_chercheur = :OLD.ID_chercheur;  
  IF nb_reserv > 0 THEN  
    RAISE_APPLICATION_ERROR(-20001, 'Impossible de supprimer un  
    chercheur avec une réservation en cours.');
```

```
  END IF;
```

```
END empecher_suppression_chercheur;
```

```
1 DELETE FROM Chercheur WHERE ID_chercheur = 1;
```

ORA-20001: Impossible de supprimer un chercheur avec une réservation en cours. ORA-06512: at
"SQL_ETMRPLNWBVBFPHCHPATSAJXEY.EMPECHER_SUPPRESSION_CHERCHEUR", line 7
ORA-06512: at "SYS.DBMS_SQL", line 1721

More Details: <https://docs.oracle.com/error-help/db/ora-20001>

6. Objectif : Afficher les fournisseur pour chaque laboratoire

CREATE OR REPLACE PROCEDURE

afficher_fournisseurs_par_laboratoire(p_laboratoire_id IN
Laboratoire.ID_lab%TYPE) AS

BEGIN

FOR i IN (SELECT f.nom_fournisseur, f.num_telephone
FROM Fournisseur f
JOIN Commande c ON f.ID_fournisseur = c.ID_fournisseur
JOIN Laboratoire l ON c.ID_lab = l.ID_lab
WHERE l.ID_lab = p_laboratoire_id) LOOP

DBMS_OUTPUT.PUT_LINE('Fournisseur : ' || i.nom_fournisseur || ', Numéro
telephone : ' || i.num_telephone);

END LOOP;

END afficher_fournisseurs_par_laboratoire;

```
1 EXEC afficher_fournisseurs_par_laboratoire(1);
```

Statement processed.

Fournisseur : ServicepourLabo, Numéro telephone : +33658100923

Fournisseur : ServicepourLabo, Numéro telephone : +33658100923