

Système De Gestion Des Dons De Sang

**Titre de cours : TP Base De Données
Avancées.**

Professeur De Cours : Debbabi Fadoua

Projet réalisé par : Rami Drira

**Programme d'études : Deuxième
Préparatoire Intègre.**

**Université : Institut Supérieur
d'Informatique et de Multimédia de
Sfax.**

Année Universitaire : 2022-2023

Sommaire

Sujet	Page No
Introduction	3
Modèle Conceptuel Des Donnes	4
Informations Sur Les Tables	5 - 6
Relations Entre Les Table s	7 - 8
Schémas Relationnel	9 - 12
Les donnees de tables	13 – 15
Création de Table s	16 - 22
Les Requêtes PLSQL	23 - 32

Introduction

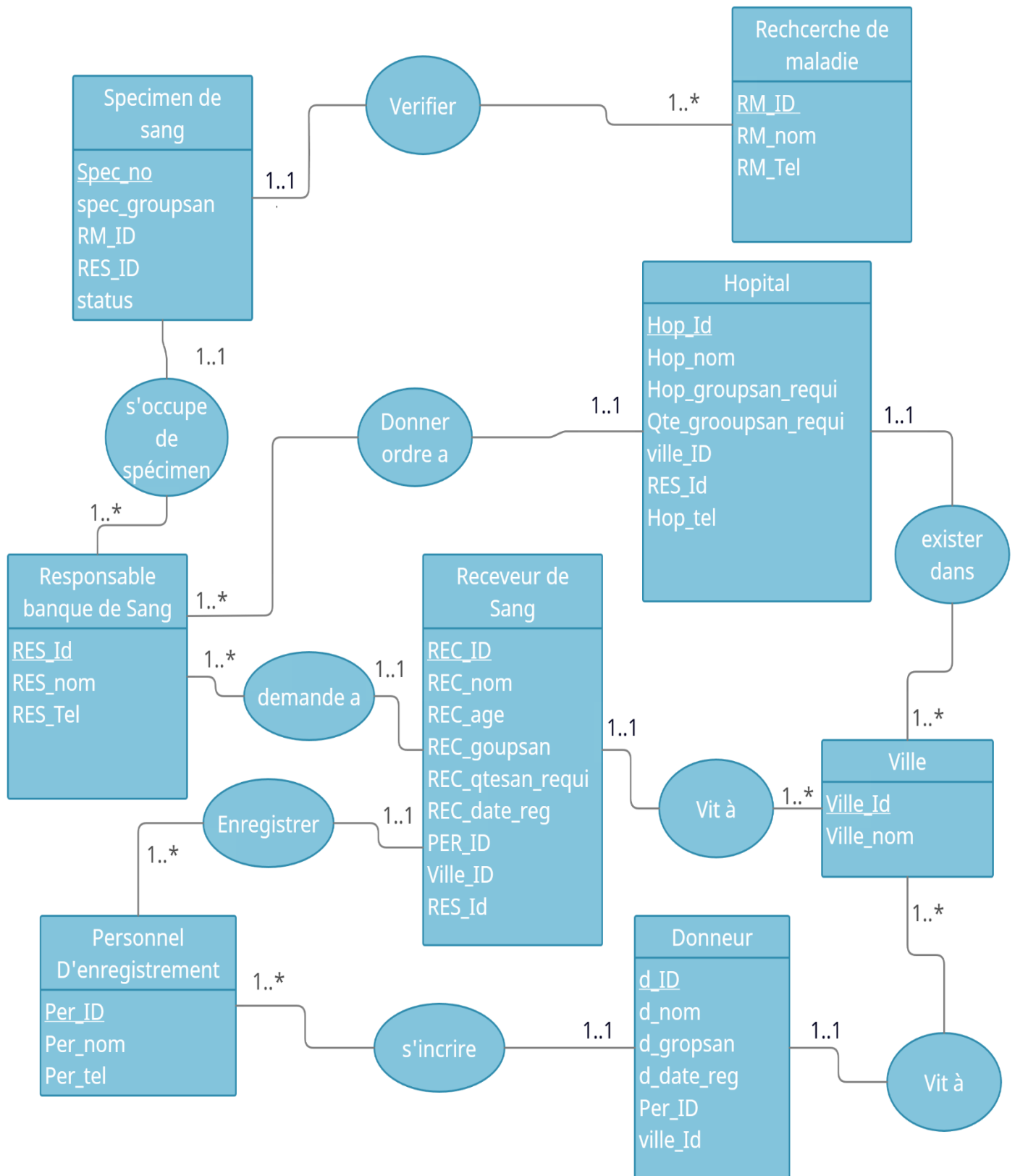
Les banques de sang collectent, stockent et fournissent le sang collecté aux patients qui en ont besoin. Les personnes qui donnent du sang sont appelées « donneurs ». Les banques regroupent alors le sang qu'elles reçoivent selon les groupes sanguins. Ils s'assurent également que le sang n'est pas contaminé. La mission principale de la banque de sang est de fournir le sang aux hôpitaux et aux soins de santé systèmes qui sauvent la vie du patient. Aucun hôpital ne peut maintenir le système de soins de santé sans sang pur et suffisant.

La principale préoccupation de chaque banque de sang est de surveiller la qualité de du sang et surveiller les personnes qui donnent le sang, c'est-à-dire les « donneurs ». Mais c'est un travail difficile. Le système existant ne satisfera pas le besoin de maintenir un sang de qualité et de suivre les donneurs. Pour surmonter toutes ces limitations, nous avons introduit un nouveau système appelé "Don de sang Système de gestion".

Le Système de gestion des banques de sang nous permet de suivre la qualité de sang et assure également le suivi du sang disponible à la demande de l'accepteur. L'existant Les systèmes sont des systèmes manuels qui prennent du temps et ne sont pas si efficaces. 'Banque du sang Le système de gestion' automatise la distribution du sang. Cette base de données contient des milliers des dossiers de chaque banque de sang.

En utilisant ce système, la recherche du sang disponible devient facile et permet d'économiser beaucoup de temps que le système manuel. Il thésaurisera, exploitera, récupérera et analysera les informations concerne la gestion administrative et des stocks au sein d'une banque de sang. Ce système est développé de manière à ce qu'il soit gérable, rapide, rentable, flexible et beaucoup la main-d'œuvre n'est pas nécessaire.

système de gestion des dons de sang



Informations Sur Les Tables

Au total, nous avons huit entités et les informations de chaque entité sont mentionnées ci-dessous :

Donneur : (d_ID, d_nom, d_gropsan, d_date_reg, Per_ID, Ville_ID)

Le donneur est la personne qui donne du sang, lors du don, un identifiant de donneur (d_ID) est généré et utilisé comme clé primaire pour identifier les informations du donneur. Autre que ce nom, groupe sanguin, date de registration seront stockés dans la base de données sous le table Donneur.

Receveur De Sang : (Rec_ID, Rec_nom, rec_age, Rec_gropsan, Rec_qtesan_requi, Rec_date_reg, Per_Id, Ville_ID, Res_ID)

Le receveur est la personne qui reçoit le sang de la banque de sang, lorsque le sang est donné à un destinataire un identifiant de destinataire (Rec_ID) est généré et utilisé comme clé primaire pour l'entité destinataire pour identifier les informations sur les receveurs de sang. Avec son nom, son âge, son groupe sanguin (nécessaire), la quantité de sang (nécessaire) et les dates d'enregistrement sont également stockés dans la base de données sous l'entité bénéficiaire.

Responsable banque de sang : (Res_ID, Res_nom, Res_tel)

Le responsable de la banque de sang est la personne qui s'occupe des prélèvements sanguins disponibles dans la banque de sang, il est également responsable du traitement des demandes de sang des receveurs et des hôpitaux. Le responsable a un numéro d'identification unique (Res_ID) utilisé comme clé primaire avec le nom et le numéro de téléphone du responsable de la banque de sang sera stocké dans la base de données sous l'entité Responsable banque de sang.

Personnel d'enregistrement: (Per_ID, Per_nom, Per_tel)

Le personnel d'enregistrement est une personne qui enregistre le donneur de sang et les receveurs et le L'entité Personnel d'enregistrement a Per_ID qui est la clé primaire avec le nom de l'enregistreur et le numéro de téléphone de l'enregistreur sera également stocké dans la base de données sous l'entité Personnel d'enregistrement.

Spécimen de sang : (Spec_no, RM_ID, Spec_gropsan, Res_ID, status)

Dans la base de données, sous l'entité Spécimen de sang, nous stockerons les informations des échantillons de sang qui sont disponibles à la banque de sang. Dans cette entité Spec_no Spec_gropsan ensemble seront clé primaire avec attribut d'état qui indiquera si le sang est contaminé ou non.

Recherche de maladie : (RM_Id, RM_nom, RM_tel)

Dans la base de données, sous l'entité Recherche de maladie, nous stockerons les informations du médecin qui vérifie le sang pour tout type de contaminations. Pour stocker ces informations, nous avons des Numéro d'identification (RM_Id) comme clé primaire. Avec le nom et le numéro de téléphone du médecin sera également stocké sous la même entité.

Hôpital: (Hop_ID, Hop_nom, Hop_gropsan_requi, Qte_gropsan_requi, Ville_ID, Res_ID, Hop_tel)

Dans la base de données, sous l'entité Hôpital, nous stockerons les informations des hôpitaux. Dans ce Hop_ID et Hop_gropsan_requi forment ensemble la clé primaire. Nous stockerons le nom de l'hôpital et la quantité de sang requise à l'hôpital.

Ville : (Ville_Id, Ville_nom)

Cette entité stockera les informations des villes où se trouvent les donneurs, les receveurs et les hôpitaux sont présents. Un numéro d'identification unique (Ville_Id) sera utilisé comme clé primaire pour définir les informations sur la ville. Avec l'ID, les noms de ville seront également stockés sous cette entité.

Relations Entre Les Tables

1.Ville et hôpital :

Relation=exister dans

Type de relation :1 a plusieurs

Explication : Une Ville peut avoir beaucoup des hôpitaux mais un hôpital existe dans une seule ville.

2.Ville et Donneur :

Relation : Vit a

Type de relation : 1 a plusieurs

Explication : Dans une ville, de nombreux donneurs peuvent vivre. Un donateur appartiendra à une ville.

3.Ville et Receveur de sang :

Relation : Vit a

Type de relation : 1 a plusieurs

Explication : Dans une ville, de nombreux receveur de sang peuvent vivre. Un receveur de sang appartiendra à une ville.

4.Personnel d'enregistrement et donneur :

Relation : s'inscrire

Type de relation :1 a plusieurs

Explication : Un personnel d'enregistrement peut enregistrer plusieurs donneurs. Un donateur s'inscrira auprès d'un agent d'enregistrement.

5.Personnel d'enregistrement et receveur de sang :

Relation : Enregistrer

Type de relation :1 a plusieurs

Explication : Un personnel d'enregistrement peut enregistrer plusieurs destinataires. Un destinataire sera enregistré par un agent d'enregistrement.

6. Hopital et Responsable banque de sang :

Relation : Donner ordre a

Type de Relation : 1 a plusieurs

Explication : Un seul gestionnaire de banque de sang peut gérer et traiter les demandes de plusieurs hôpitaux. Un hôpital enverra une demande au responsable de la banque de sang.

7. Responsable banque de sang et Spécimen de sang :

Relation : s'occupe de spécimen

Type de relation : 1 a plusieurs

Explication : Un gestionnaire de banque de sang peut gérer plusieurs échantillons de sang et un spécimen sera géré par un gestionnaire.

8. Receveur de sang et Responsable banque de sang :

Relation : demande a

Type de relation : 1 a plusieurs

Explication : Un receveur peut demander du sang à un gestionnaire et un gestionnaire peut gérer les demandes de nombreux destinataires.

9. Rechercheur de maladie et Spécimen de sang :

Relation : vérifier

Type de relation : 1 a plusieurs

Explication : Un détecteur de maladie peut vérifier de nombreux échantillons de sang. Un échantillon de sang est vérifié par un chercheur de maladie.

SCHÉMAS RELATIONNELS

Donneur :

Attributs	Description	Type
d_ID (Primary key)	Donneur de sang ID	Number
d_nom	Nom du donneur de sang	Varchar
d_groupsan	Group sanguin du donneur de sang	Varchar
d_date_reg	Date de registration du donneur de sang	Date
Per_Id (Foreign Key)	ID du personnel d'enregistrement	Number
Ville_ID (Foreign Key)	ID du ville	Number

- ⇒ La Relation entre Personnel d'enregistrement et donneur est 1 a plusieurs donc on prend Per_Id clé étrangère.
- ⇒ La Relation entre ville et Donneur est 1 a plusieurs donc on prend Ville_Id clé étrangère.

Ville :

Attributs	Description	Type
Ville_ID (Primary key)	ID du ville	Number
Ville_nom	Nom du ville	varchar

- ⇒ La relation entre Ville et Receveur de sang, Donneur, Hôpital est 1 à plusieurs c'est pourquoi la clé primaire de Ville est pris comme une clé étrangère dans Receveur de sang, Donneur et Hôpital.

Personnel d'enregistrement :

Attributs	Description	Type
Per_Id (Primary key)	ID du personnel d'enregistrement	Number
Per_nom	Nom du personnel d'enregistrement	Varchar
Per_tel	Téléphone du personnel d'enregistrement	Number

⇒ La Relation entre Personnel d'enregistrement et Donneur, Receveur de sang est 1 à plusieurs c'est pourquoi on prend la clé primaire de Personnel d'enregistrement comme un clé étrangers dans Donneur et Receveur de sang.

Receveur de sang :

Attributs	Description	Type
Rec_Id (Primary key)	ID du receveur de sang	Number
Rec_nom	Nom du receveur de sang	Varchar
Rec_age	Age du receveur de sang	Number
Rec_groupsan	Group sanguin du receveur de sang	Varchar
Rec_qtesan_requi	Quantité de sang requis par le récepteur	Number
Rec_date_reg	Date de registration du receveur de sang	Date
Per_ID (Foreign Key)	ID du personnel d'enregistrement	Number
Ville_ID (Foreign Key)	ID du ville	Number
Res_Id (Foreign Key)	ID du Responsable banque de sang	Number

⇒ La relation entre Personnel d'enregistrement et Receveur de sang est 1 à plusieurs c'est pourquoi on prend Per_ID comme clé étrangère.

- ⇒ La Relation entre Ville et Receveur de sang est 1 à plusieurs c'est pourquoi on prend Ville_Id comme clé étrangère.
- ⇒ La Relation entre Responsable banque de sang et Receveur de sang est 1 à plusieurs c'est pourquoi on prend Res_ID comme clé étrangère.

Spécimen de sang :

Attributs	Description	Type
Spec_no (Primary Key)	Numéro unique de Spécimen de sang	Number
RM_Id (Foreign Key)	ID du Recherche maladie	Number
Spec_groupsan	Group sanguin du Specimen de Sang	Varchar
Res_Id (Foreign Key)	ID du responsable banque de sang	Number
Status	Le sang est-il pur ou non ?	Number

- ⇒ La relation entre Responsable banque de sang et Spécimen de sang est 1 à plusieurs c'est pourquoi on prend Res_ID comme un clé étrangère.
- ⇒ La Relation entre Recherche maladie et Spécimen de sang est 1 à plusieurs c'est pourquoi on prend RM_ID comme clé étrangère.

Recherche Maladie :

Attributs	Description	Type
RM_ID (Primary key)	ID du recherche de maladie	Number
RM_nom	Le nom de recherche de maladie	Varchar
RM_tel	Le numéro de téléphone de recherche de maladie	Number

- ⇒ La relation entre Recherche de maladie et Spécimen de sang est 1 à plusieurs c'est pourquoi on prend RM_ID comme clé étranger pour Spécimen de sang.

Responsable banque de sang :

Attributs	Description	Type
Res_ID	ID du responsable banque de sang	Number
Res_nom	Nom du Responsable de sang	Varchar
Res_tel	Téléphone du Responsable de sang	Number

- ⇒ La relation entre Responsable de sang et Spécimen de sang , Receveur de sang , hôpital sont tous 1 à plusieurs c'est pourquoi on prend Res_Id comme clé étrangère dans Spécimen de sang , receveur de sang et hôpital.

Hôpital :

Attributs	Description	Type
Hop_ID (primary key)	ID du Hopital	Number
Hop_nom	Nom du Hopital	Varchar
Hop_groupsan_requi (Primary key)	Group sanguin requis par l'hôpital	Varchar
Qte_groupsan_requi	Quantité du groupe sanguin requis	Number
Ville_ID (Foreign Key)	ID du Ville	Number
Res_ID (Foreign Key)	ID du responsable banque de sang	Number
Hop_tel	Numéro de téléphone d'hôpital	Number

- ⇒ La relation entre Hôpital et Ville est 1 à plusieurs donc on prend Ville_ID comme clé étrangère.
- ⇒ La relation enter Hôpital et Responsable banque de sang est 1 à plusieurs donc on prend Res_ID comme clé étrangère.

Les Données des tables

Ville :

Ville_ID	Ville_nom
1200	sfax
1300	tunis
1400	sousse
1500	Gabes
1600	Jandouba
1700	Nabeul

Responsable banque de sang (RB_SANG) :

Res_ID	Res_nom	Res_tel
101	Mariem	93549022
103	Mahdi	94045983
104	Mohamed	52393324
105	Rami	22394850
106	Ahmed	98342012
107	Soulaima	99129323
108	Nour	52982138

Donneur :

D_ID	D_nom	D_groupsan	D_date_reg	Per_ID	Ville_ID
150011	Malik	A-	2022-07-19	101412	1200
150021	Yessine	AB+	2018-12-06	101212	1200
150045	Mohamed	A-	2022-08-15	101312	1200
160091	Mokhtar	B-	2018-06-14	101312	1400
160401	Mohsin	O-	2016-08-22	102342	1500
150121	Asma	A+	2022-10-17	101312	1600
180301	Ichrak	O+	2017-04-16	101617	1300
180563	Housin	B+	2017-02-24	102349	1300

Spécimen de sang (specsan) :

Spec_Id	RM_ID	Spec_groupsan	Res_ID	status
1001	11	B+	101	1
1002	12	O+	106	1
1003	11	AB+	106	1
1004	13	O-	103	0
1005	14	A+	105	1
1006	13	AB-	107	1
1007	15	B+	104	0
1008	11	B-	105	1

Recherche Maladie (RMAL) :

RM_ID	RM_nom	RM_tel
11	mohamed	99123456
12	Yessine	52948351
13	Mahdi	56938429
14	Oussema	26893932
15	Ala	25992321
16	Achref	98372432
17	Ons	92323929

Personnel d'enregistrement (PERSEN):

Per_ID	Per_nom	Per_tel
101212	Nour	99232455
102342	Melek	54329031
102349	Rami	53899123
101312	Salim	98384241
101412	Dhia	93131822
101617	Bilel	22381239

Hôpital :

Hop_ID	Hop_nom	Hop_groupsan_requi	Qte_groupsan_requi	Ville_ID	Res_ID	Hop_tel
1	Charles-Nicolle	A+	20	1300	101	99232435
2	Taher-Maamouri	A-	40	1700	103	53241454
3	Farhat-Hached	AB+	0	1400	104	93821283
4	Sahloul	B-	20	1400	103	59124249
5	HR MOHAMED BEN SASSI	AB-	10	1500	107	93423423
6	Hôpital Habib Bourguiba	B+	30	1200	106	23845342
7	HR DE JENDOUBA	A+	0	1600	104	58383249

Receveur de sang (REC_SAN) :

Rec_ID	Rec_nom	Rec_age	Rec_groupsan	Rec_qtesan_requi	Rec_date_reg	Per_ID	Ville_Id	Res_ID
10001	Anas	34	B+	1.5	2016-12-17	101212	1200	101
10002	Mahmoud	43	A-	1	2018-10-17	102342	1200	106
10003	Maki	23	AB+	0.5	2022-06-23	102349	1300	104
10004	Achref	56	O+	1	2018-08-27	101412	1400	103
1005	Aymen	32	B-	1	2017-03-22	101617	1600	108
1006	Racem	29	A+	3.5	2017-10-17	101312	1300	106
1007	Anis	56	B+	2	2016-12-13	101617	1200	101

Création Des Tables

```
create table VILLE(  
    VILLE_ID number(8) primary key,  
    VILLE_NOM varchar2(20)  
);  
  
Insert into VILLE(VILLE_ID,VILLE_nom)  
Values(1200,'sfax');  
  
Insert into VILLE(VILLE_ID,VILLE_nom)  
Values(1300,'tunis');  
  
Insert into VILLE(VILLE_ID,VILLE_nom)  
Values(1400,'sousse');  
  
Insert into VILLE(VILLE_ID,VILLE_nom)  
Values(1500,'gabes');  
  
Insert into VILLE(VILLE_ID,VILLE_nom)  
Values(1600,'jerba');  
  
Insert into VILLE(VILLE_ID,VILLE_nom)  
Values(1700,'nabeul');  
  
create table RB_SANG (  
    Res_ID number(3) primary key,  
    Res_nom varchar2(20),  
    Res_tel number(8)  
);  
  
insert into RB_SANG  
values(101,'Mariem',93549022);  
  
insert into RB_SANG  
values(103,'Mahdi',94045983);
```



```

insert into RB_SANG
values(104,'Mohamed',52393324);
insert into RB_SANG
values(105,'Rami',22394850);
insert into RB_SANG
values(106,'Ahmed',98342012);
insert into RB_SANG
Values(107,'Soulaima',99129323) ;
Insert into RB_SANG
Values(108,'Nour',52982138) ;
create table PERSEN(
    Per_ID number primary key,
    Per_nom varchar2(20),
    Per_tel number(8)
);
insert into PERSEN
values(101212,'Nour',99232455);
insert into PERSEN
values(102342,'Melek',54329031);
insert into PERSEN
values(102349,'Rami',53899123);
insert into PERSEN
values(101312,'Salim',98384241);
insert into PERSEN
values(101412,'Dhia',93131822);
insert into PERSEN
values(101617,'Bilel',22381239);

```

```

create table DONNEUR (
    D_ID number (6) primary key,
    D_nom varchar2(20),
    D_groupsan varchar2(6),
    D_date_reg date,
    Per_ID number,
    VILLE_ID number (8),
    constraint fk_villeid foreign key (VILLE_ID) references
    VILLE(VILLE_ID),
    constraint fk_perid foreign key (Per_ID) references PERSEN(Per_ID)
);

insert into DONNEUR
values(150011,'Malik','A-','19/07/2022',101412,1200);

insert into DONNEUR
values(150021,'Yessine','AB+','06/12/2018',101212,1200);

insert into DONNEUR
values(160091,'Mokhtar','B-','14/06/2018',101312,1400);

insert into DONNEUR
values(160401,'Mohsin','O-','22/08/2016',102342,1500);

insert into DONNEUR
values(150121,'Asma','A+','17/10/2022',101312,1600);

insert into DONNEUR
values(180563,'Housin','B+','24/02/2017',102349,1300);

insert into DONNEUR
values(150045,'Mohamed','A-','15/08/2022',101312,1200);

```

```

create table RMAL (

```

```
RM_ID number primary key,  
RM_nom varchar2(20),  
RM_tel number (8)  
);  
  
insert into RMAL  
values(11,'mohamed',99123456);  
  
insert into RMAL  
values(12,'Yessine',52948351);  
  
insert into RMAL  
values(13,'Mahdi',56938429);  
  
insert into RMAL  
values(14,'Oussema',26893932);  
  
insert into RMAL  
values(15,'Ala',25992321);  
  
insert into RMAL  
values(16,'Achref',98372432);  
  
insert into RMAL  
values(17,'Ons',92323929);
```

```

create table SPECSAN(
    Spec_Id number primary key,
    RM_ID number,
    Spec_groupsan varchar2(6),
    Res_ID number(3),
    status number(1),
    constraint fk_rmid foreign key (RM_ID) references RMAL(RM_ID),
    constraint fk_resid foreign key (Res_ID) references
    RB_SANG(Res_ID)
);

insert into SPECSAN
values(1001,11,'B+',101,1);
insert into SPECSAN
values(1002,12,'O+',106,1);
insert into SPECSAN
values(1003,11,'AB+',106,1);
insert into SPECSAN
values(1004,13,'O-',103,0);
insert into SPECSAN
values(1005,14,'A+',105,1);
insert into SPECSAN
values(1006,13,'AB-',107,1);
insert into SPECSAN
values(1008,11,'B-',105,1);
create table HOPITAL (
    Hop_ID number,
    Hop_nom varchar2(20),

```

```

Hop_groupsan_requi varchar2(9),
Qte_groupsan_requi number,
Ville_ID number,
RES_ID number (3),
Hop_tel number (8),
constraint pk_hoppk primary key(Hop_ID,Hop_groupsan_requi),
constraint fk_villeidd foreign key (Ville_ID) VILLE(VILLE_ID),
constraint fk_residd foreign key (RES_ID) references
RB_SANG(RES_ID)
);

insert into HOPITAL
values(1,'Charles-Nicolle','A+',20,1300,101,99232435);

insert into HOPITAL
values(2,'Taher-Maamouri','A-',40,1700,103,53241454);

insert into HOPITAL
values(3,'Farhat-Hached','AB+',0,1400,104,93821283);

insert into HOPITAL
values(4,'Sahloul','B-',20,1400,103,59124249);

insert into HOPITAL
values (5,'HR MOHAMED BEN SASSI','AB-',10,1500,107,93423423);

insert into HOPITAL
values (6,'Habib Bourguiba','B+',30,1200,106,23845342);

insert into HOPITAL
values (7,'HR DE JENDOUBA','A+',0,1600,104,58383249);

create table REC_SAN(
    Rec_ID number primary key,
    Rec_nom varchar2(20),

```

```

Rec_age number,
Rec_groupsan varchar2(20),
Rec_qtesan_requi number,
Rec_date_reg date,
Per_ID number,
Ville_Id number,
Res_ID number ,

constraint fk_peridd foreign key(PER_ID) references
PERSEN(PER_ID),

constraint fk_ville foreign key(VILLE_ID) references
VILLE(VILLE_ID),

constraint fk_res foreign key(RES_Id) references
RB_SANG(RES_ID)
);

insert into REC_SAN
values(10001,'Anas',34,'B+',1.5,'17/12/2016',101212,1200,101);

insert into REC_SAN
values(10002,'Mahmoud',43,'A-',1,'17/10/2018',102342,1200,106);

insert into REC_SAN
values(10003,'Maki',23,'AB+',0.5,'23/06/2022',102349,1300,104);

insert into REC_SAN
values(10004,'Achref',56,'O++',1,'27/08/2018',101412,1400,103);

insert into REC_SAN
values(10005,'Aymen',32,'B-',1,'22/03/2017',101617,1600,108);

insert into REC_SAN
values(10006,'Racem',29,'A+',3.5,'17/10/2017',101312,1300,106);

```

Les Requêtes PLSQL

R1 : Affichez les informations de Hospital géré par responsable de banque de sang dont l'ID est 103 :

```
declare
cursor c1 is
select Hop_ID , Hop_nom , Ville_ID, h.RES_ID as res
from HOPITAL h, RB_SANG r
where h.RES_ID=r.RES_ID and r.RES_ID=103;
begin
for wc1 in c1 loop
dbms_output.put_line('le responsable d id '||wc1.res||' gere l hopital d
id'||wc1.Hop_ID||' et de nom '||wc1.Hop_nom||' qui se trouve dans la
ville d id '||wc1.VILLE_ID);
End loop ;
End ;
```

```
le responsable d id 103 gere l hopital d id4 et de nom Sahloul qui se trouve dans la ville d id 1400
le responsable d id 103 gere l hopital d id2 et de nom Taher-Maamouri qui se trouve dans la ville d id 1700
Instruction traitée.
```

R2 : Afficher les IDS et noms des donneurs qui registre entre '01/01/2017' et '23/07/2022' et qui habitent dans Sfax (VILLE_ID=1200)

```
declare
cursor c1
is select D_ID,D_nom
from DONNEUR d
where d.VILLE_ID=1200 and D_date_reg between ('01/01/2017') and
('23/07/2022');
begin
for wc1 in c1 loop
```

```

dbms_output.put_line('ID du donneur '||wc1.D_ID||' nom du donneur
'||wc1.D_nom);
end loop;
exception
when no_data_found then
dbms_output.put_line('il n y a pas de donneur avec date de registraion
entre 01/01/2017 et 23/07/2022 qui se trouve dans safx');
end;

```

```

ID du donneur 150011 nom du donneur Malik
ID du donneur 150021 nom du donneur Yessine

Instruction traitée.

```

R3 : Ecrire un programme en PL/SQL permettant la saisie des informations d'un nouvel donneur et de faire le traitement suivant :

-si le groupe sanguin est nulle, traiter une exception et afficher un Message d'erreur « Un donneur doit avoir un groupe sanguin ».

Sinon, insérer les données relatives à ce nouvel donneur dans la table donneur.

```

declare
idd DONNEUR.D_ID%type;
nom DONNEUR.D_nom%type;
groupsanguin DONNEUR.D_groupsan%type;
registration DONNEUR.D_date_reg%type;
personnel DONNEUR.Per_ID%type;
villeid DONNEUR.Ville_ID%type;
no_groupsan exception;
begin
idd:=:idd;
nom:=:nom;
groupsanguin:=:groupsanguin;

```



```

registration:=:registration;
personnel:=:personnel;
villeid:=:villeid;
if groupsanguin=null then
raise
no_groupsan;
else
insert into DONNEUR
values(idd,nom,groupsanguin,registration,personnel,villeid);
end if;
exception
when no_groupsan then
dbms_output.put_line('Un donneur doit avoir un groupe sanguin');
end;

```

1 ligne(s) insérée(s).

Soumettre	
Variable attachée	Valeur
:IDD	190563
:NOM	maki
:GROUPSANGUIN	A+
:REGISTRATION	19/4/2017
:PERSONNEL	101412
:VILLEID	1300

R4 : Montrer l'échantillon de sang pur manipulé par le Responsable banque de sang qui s'occupe également d'un receveur ayant besoin

Le même groupe sanguin ainsi que les coordonnées du Responsable banque de sang et du receveur de sang.

```
declare
```

```
cursor c1 is
```

```
select rb.RES_ID as res, RES_nom , r.REC_nom as rec ,  
r.REC_groupsan as san, SPEC_GROUPSAN
```

```
from RB_SANG rb , REC_SAN r , SPECSAN s
```

```
where r.RES_ID=s.RES_ID and r.REC_groupsan=s.SPEC_groupsan  
and s.RES_ID = rb.RES_ID and status=1;
```

```
begin
```

```
for wc1 in C1 loop
```

```
dbms_output.put_line('Le responsable '||wc1.RES_nom||' de Banque de  
sang d id '||wc1.res||' prend soin de specimen sanguin du groupe  
'||wc1.san||' pour un receveur de meme group sanguin '||wc1.rec);
```

```
end loop;
```

```
exception
```

```
when no_data_found then
```

```
dbms_output.put_line('aucun responsable prend soin d un specimen de  
sang qui a le meme groupe sanguin de receveur');
```

```
end;
```

```
Le responsable Mariem de Banque de sang d id 101 prend soin de specimen sanguin du groupe B+ pour un receveur de meme group sanguin Anas
```

```
Instruction traitée.
```

R5 : sélectionner l'identifiant et le nom du donneur qui a le même groupe sanguin taper par l'utilisateur :

```
declare
blood DONNEUR.D_groupsan%type;
cursor c1 (sang DONNEUR.D_groupsan%type)
is
select D_ID,D_nom
from DONNEUR d
where d.D_groupsan=sang;
begin
blood:=:blood;
for wc1 in c1(blood) loop
dbms_output.put_line('IDentifiant du donneur est '||wc1.D_ID||' nom du
donneur '||wc1.D_nom);
end loop;
end;
```

Si on prend blood='A+' =>

```
IDentifiant du donneur est 150121 nom du donneur Asma
Instruction traitée.
0,26 secondes
```

R6 : Donner le pourcentage de chaque groupe sanguin qui se trouve dans la ville de Sfax (VILLE_ID=1200)

```
declare
cursor c1
is select D_groupsan, count(D_groupsan) as nb
from DONNEUR d
where d.VILLE_ID=1200
group by D_groupsan;
```

```

s number;
n number (6,2);
begin
s:=0;
for wc1 in c1 loop
s:=s+wc1.nb;
end loop;
for wc1 in c1 loop
n:=(wc1.nb*100)/s;
dbms_output.put_line ('le pourcentage de groupe sanguin
'||wc1.D_groupsan||' est '||n||' %');
End Loop ;
End ;

```

```

le pourcentage de groupe sanguin AB+ est 33,33 %
le pourcentage de groupe sanguin A- est 66,67 %

Instruction traitée.

0,24 secondes

```

R7 : afficher les receveurs de sangs et les donneurs qui ont le même groupe sanguin et qui vit dans la même ville.

```

declare
cursor c1
is select REC_ID , REC_nom , REC_age
,D_ID,REC_groupsan,d.VILLE_ID as VILLE_ID
from REC_SAN r , DONNEUR d
where d.D_GROUPSAN=r.REC_GROUPSAN
and r.VILLE_ID=d.VILLE_ID;
cursor c2(iden VILLE.VILLE_ID%type)
is select VILLE_nom
from VILLE

```

```

where VILLE_ID=iden;
vill VILLE.VILLE_nom%type;
begin
for wc1 in c1 loop
for wc2 in c2(wc1.VILLE_ID) loop
vill:=wc2.VILLE_nom;
end loop;

dbms_output.put_line('le receveur de sang d ID '||wc1.REC_ID||' et le
donneur d ID '||wc1.D_ID||' vit dans la meme ville '||vill||' Ils ont le meme
groupe sanguin '||wc1.REC_groupsan);
end loop;
exception
when No_data_found then
    dbms_output.put_line('Il n existe pas un donneur et un receveur de
sang qui ont le meme groupe sanguin et qui vit dans la meme ville');
end;

```

```

le receveur de sang d ID 10002 et le donneur d ID 150045 vit dans la meme ville sfax Ils ont le meme groupe sanguin A-
le receveur de sang d ID 10002 et le donneur d ID 150011 vit dans la meme ville sfax Ils ont le meme groupe sanguin A-

Instruction traitée.

0,03 secondes

```

R8 : Afficher les informations des chercheurs qui trouvent une maladie dans le spécimen de sangs et les supprimer de table specsan :

```

declare
cursor c1
is select r.RM_ID as rmid , RM_nom
from RMAL r,specsan s
where r.RM_ID=s.RM_ID and status=0;
begin
for wc1 in c1 loop

```

```
dbms_output.put_line('le chercheur d id '||wc1.rmid||' et de nom '||wc1.RM_nom||'a trouve un specimen de sang non pure (avec maladie)');
```








```
delete from specsan where RM_ID=wc1.rmid;
```






```
end loop;
```

```
end;
```

```
le chercheur d id 15 et de nom Alaa trouve un specimen de sang non pure (avec maladie)
le chercheur d id 13 et de nom Mahdia trouve un specimen de sang non pure (avec maladie)

1 ligne(s) supprimée(s)
```

MODIFIER	SPEC_ID	RM_ID	SPEC_GROUPSAN	RES_ID	STATUS
	1002	12	O+	106	1
	1008	11	B-	105	1
	1004	13	O-	103	0
	1003	11	AB+	106	1
	1001	11	B+	101	1
	1005	14	A+	105	1
	1007	15	B+	104	0

MODIFIER	SPEC_ID	RM_ID	SPEC_GROUPSAN	RES_ID	STATUS
	1002	12	O+	106	1
	1008	11	B-	105	1
	1003	11	AB+	106	1
	1001	11	B+	101	1
	1005	14	A+	105	1

R9 : Pour chaque ville afficher la quantité de sang qui va avoir de la banque de sang :

```
declare
```

```
cursor c1 is
```

```
select qte_groupsan_requi as sang , VILLE_ID
```

```
from HOPITAL;
```

```
cursor c2 is
```

```
select VILLE_ID ,VILLE_NOM as nom
```

```
from VILLE;
```

```
nb number;
```

```

begin
for wc2 in c2 loop
nb:=0;
for wc1 in c1 loop
if (wc1.VILLE_ID=wc2.VILLE_ID) then
nb:=nb+wc1.sang;
end if;
end loop;
dbms_output.put_line('La quantite de sang pour la ville
'||wc2.VILLE_ID||' est '||nb);
end loop;
end;

```

```

La quantite de sang pour la ville 1600 est 0
La quantite de sang pour la ville 1700 est 40
La quantite de sang pour la ville 1300 est 20
La quantite de sang pour la ville 1200 est 30
La quantite de sang pour la ville 1400 est 20
La quantite de sang pour la ville 1500 est 10

Instruction traitée.

```

R10 : Afficher le nombre d'hôpital par responsable :

```

declare
cursor c1 is
select RES_ID ,RES_nom as nom
from RB_SANG;
cursor c2 is
select RES_ID as res
from HOPITAL;
nb number;
begin
for wc1 in c1 loop
nb:=0;

```

```
for wc2 in c2 loop
if (wc1.RES_ID=wc2.res) then
nb:=nb+1;
end if;
end loop;
dbms_output.put_line('le responsable d id '||wc1.RES_ID||' et de nom
'||wc1.nom||' est responsable de '||nb||' Hopital');
end loop;
end;
```

```
le responsable d id 103 et de nom Mahdi est responsable de 2 Hopital
le responsable d id 104 et de nom Mohamed est responsable de 2 Hopital
le responsable d id 105 et de nom Rami est responsable de 0 Hopital
le responsable d id 107 et de nom Soulaïma est responsable de 1 Hopital
le responsable d id 108 et de nom Nour est responsable de 0 Hopital
le responsable d id 101 et de nom Mariem est responsable de 1 Hopital
le responsable d id 106 et de nom Ahmed est responsable de 1 Hopital
```