SAE 3

JUNIOR DEV
NIANG SADIYA ET LINGET JULIEN
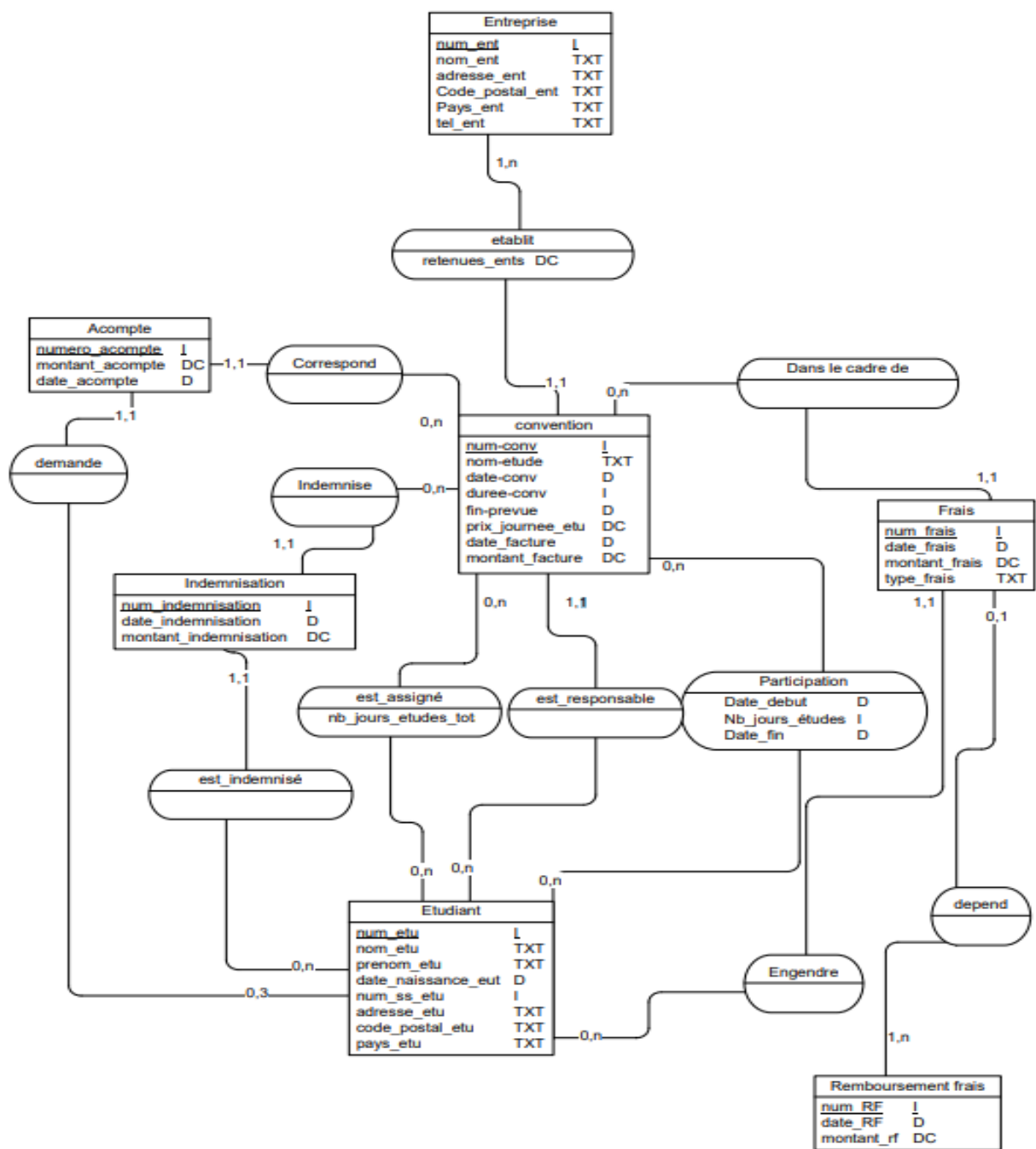
Table des matières

Partie 1

MCD

Notre MCD est le suivant :

MPD

Nous avons modifié le MPD correspondant MCD généré par l'AGL en ajoutant aux clés primaires de tables `GENERATED ALWAYS  AS IDENTITY` `START WITH 1 INCREMENT BY 1` ainsi qu'en modifiant tous les `CLOB` par `VARCHAR2(150)` pour les noms, prénoms et noms de projet, `VARCHAR2(300)` pour les adresses et `VARCHAR2(20)` pour le type de frais et les codes postaux.

```
alter table ACOMPTE
   drop constraint FK_ACOMPTE_CORRESPON_CONVENTI;

alter table ACOMPTE
   drop constraint FK_ACOMPTE_DEMANDE_ETUDIANT;

alter table CONVENTION
   drop constraint FK_CONVENTI_EST_RESPO_ETUDIANT;

alter table CONVENTION
   drop constraint FK_CONVENTI_ETABLIT_ENTREPRI;

alter table EST_ASSIGNE
   drop constraint FK_EST_ASSI_EST_ASSIG_CONVENTI;

alter table EST_ASSIGNE
   drop constraint FK_EST_ASSI_EST_ASSIG_ETUDIANT;

alter table FRAIS
   drop constraint FK_FRAIS_DANS_LE_C_CONVENTI;

alter table FRAIS
   drop constraint FK_FRAIS_DEPEND_REMBOURS;

alter table FRAIS
   drop constraint FK_FRAIS_ENGENDRE_ETUDIANT;

alter table INDEMNISATION
   drop constraint FK_INDEMNIS_EST_INDEM_ETUDIANT;

alter table INDEMNISATION
   drop constraint FK_INDEMNIS_INDEMNISE_CONVENTI;

alter table PARTICIPATION
   drop constraint FK_PARTICIP_PARTICIPA_ETUDIANT;

alter table PARTICIPATION
   drop constraint FK_PARTICIP_PARTICIPA_CONVENTI;
```

```
drop index CORRESPOND_FK;

drop index DEMANDE_FK;

drop table ACOMPTE cascade constraints;

drop index EST_RESPONSABLE_FK;

drop index ETABLIT_FK;

drop table CONVENTION cascade constraints;

drop table ENTREPRISE cascade constraints;

drop index EST_ASSIGNE2_FK;

drop index EST_ASSIGNE_FK;

drop table EST_ASSIGNE cascade constraints;

drop table ETUDIANT cascade constraints;

drop index DEPEND_FK;

drop index DANS_LE_CADRE_DE_FK;

drop index ENGENDRE_FK;

drop table FRAIS cascade constraints;

drop index EST_INDEMNISE_FK;

drop index INDEMNISE_FK;

drop table INDEMNISATION cascade constraints;

drop index PARTICIPATION2_FK;

drop index PARTICIPATION_FK;

drop table PARTICIPATION cascade constraints;

drop table REMBOURSEMENT_FRAIS cascade constraints;

/*==============================================================*/
/* Table : ACOMPTE                                              */
```

```
/*================================================================*/
create table ACOMPTE
(
   NUMERO_ACOMPTE       INTEGER              GENERATED ALWAYS  AS IDENTITY
START WITH 1 INCREMENT BY 1,
   NUM_CONV             INTEGER              not null,
   NUM_ETU              INTEGER              not null,
   MONTANT_ACOMPTE      NUMBER,
   DATE_ACOMPTE         DATE,
   constraint PK_ACOMPTE primary key (NUMERO_ACOMPTE)
);

/*================================================================*/
/* Index : DEMANDE_FK                                             */
/*================================================================*/
create index DEMANDE_FK on ACOMPTE (
   NUM_ETU ASC
);

/*================================================================*/
/* Index : CORRESPOND_FK                                          */
/*================================================================*/
create index CORRESPOND_FK on ACOMPTE (
   NUM_CONV ASC
);

/*================================================================*/
/* Table : CONVENTION                                             */
/*================================================================*/
create table CONVENTION
(
   NUM_CONV             INTEGER              GENERATED ALWAYS  AS IDENTITY
START WITH 1 INCREMENT BY 1,
   NUM_ETU              INTEGER              not null,
   NUM_ENT              INTEGER              not null,
   NOM_ETUDE            VARCHAR2(150),
   DATE_CONV            DATE,
   DUREE_CONV           INTEGER,
   FIN_PREVUE           DATE,
   PRIX_JOURNEE_ETU     NUMBER,
   DATE_FACTURE         DATE,
   MONTANT_FACTURE      NUMBER,
   RETENUES_ENTS        NUMBER,
   constraint PK_CONVENTION primary key (NUM_CONV)
);
```

```sql
/*==============================================================*/
/* Index : ETABLIT_FK                                           */
/*==============================================================*/
create index ETABLIT_FK on CONVENTION (
   NUM_ENT ASC
);


/*==============================================================*/
/* Index : EST_RESPONSABLE_FK                                   */
/*==============================================================*/
create index EST_RESPONSABLE_FK on CONVENTION (
   NUM_ETU ASC
);


/*==============================================================*/
/* Table : ENTREPRISE                                           */
/*==============================================================*/
create table ENTREPRISE
(
   NUM_ENT              INTEGER              GENERATED ALWAYS  AS IDENTITY
START WITH 1 INCREMENT BY 1,
   NOM_ENT              VARCHAR2(150),
   ADRESSE_ENT          VARCHAR2(300),
   CODE_POSTAL_ENT      VARCHAR2(20),
   PAYS_ENT             VARCHAR2(150),
   TEL_ENT              VARCHAR2(150),
   constraint PK_ENTREPRISE primary key (NUM_ENT)
);


/*==============================================================*/
/* Table : EST_ASSIGNE                                          */
/*==============================================================*/
create table EST_ASSIGNE
(
   NUM_CONV             INTEGER              not null,
   NUM_ETU              INTEGER              not null,
   NB_JOURS_ETUDES_TOTAL INTEGER             DEFAULT 0,
   constraint PK_EST_ASSIGNE primary key (NUM_CONV, NUM_ETU)
);


/*==============================================================*/
/* Index : EST_ASSIGNE_FK                                       */
/*==============================================================*/
create index EST_ASSIGNE_FK on EST_ASSIGNE (
   NUM_CONV ASC
);
```

```sql
/*==============================================================*/
/* Index : EST_ASSIGNE2_FK                                      */
/*==============================================================*/
create index EST_ASSIGNE2_FK on EST_ASSIGNE (
   NUM_ETU ASC
);


/*==============================================================*/
/* Table : ETUDIANT                                             */
/*==============================================================*/
create table ETUDIANT
(
   NUM_ETU              INTEGER                 GENERATED ALWAYS  AS IDENTITY
START WITH 1 INCREMENT BY 1,
   NOM_ETU              VARCHAR2(150),
   PRENOM_ETU           VARCHAR2(150),
   DATE_NAISSANCE_EUT   DATE,
   NUM_SS_ETU           INTEGER,
   ADRESSE_ETU          VARCHAR2(300),
   CODE_POSTAL_ETU      VARCHAR2(20),
   PAYS_ETU             VARCHAR2(150),
   constraint PK_ETUDIANT primary key (NUM_ETU)
);


/*==============================================================*/
/* Table : FRAIS                                                */
/*==============================================================*/
create table FRAIS
(
   NUM_FRAIS            INTEGER                 GENERATED ALWAYS  AS IDENTITY
START WITH 1 INCREMENT BY 1,
   NUM_RF               INTEGER                 DEFAULT NULL,
   NUM_CONV             INTEGER                 not null,
   NUM_ETU              INTEGER                 not null,
   DATE_FRAIS           DATE,
   MONTANT_FRAIS        NUMBER,
   TYPE_FRAIS           VARCHAR2(20),
   constraint PK_FRAIS primary key (NUM_FRAIS)
);


/*==============================================================*/
/* Index : ENGENDRE_FK                                          */
/*==============================================================*/
create index ENGENDRE_FK on FRAIS (
   NUM_ETU ASC
```

```sql
);

/*==================================================================*/
/* Index : DANS_LE_CADRE_DE_FK                                      */
/*==================================================================*/
create index DANS_LE_CADRE_DE_FK on FRAIS (
   NUM_CONV ASC
);

/*==================================================================*/
/* Index : DEPEND_FK                                                */
/*==================================================================*/
create index DEPEND_FK on FRAIS (
   NUM_RF ASC
);

/*==================================================================*/
/* Table : INDEMNISATION                                            */
/*==================================================================*/
create table INDEMNISATION
(
   NUM_INDEMNISATION    INTEGER              GENERATED ALWAYS  AS IDENTITY
START WITH 1 INCREMENT BY 1,
   NUM_CONV             INTEGER              not null,
   NUM_ETU              INTEGER              not null,
   DATE_INDEMNISATION   DATE,
   MONTANT_INDEMNISATION NUMBER,
   constraint PK_INDEMNISATION primary key (NUM_INDEMNISATION)
);

/*==================================================================*/
/* Index : INDEMNISE_FK                                             */
/*==================================================================*/
create index INDEMNISE_FK on INDEMNISATION (
   NUM_CONV ASC
);

/*==================================================================*/
/* Index : EST_INDEMNISE_FK                                         */
/*==================================================================*/
create index EST_INDEMNISE_FK on INDEMNISATION (
   NUM_ETU ASC
);

/*==================================================================*/
/* Table : PARTICIPATION                                            */
```

```sql
/*===============================================================*/
create table PARTICIPATION
(
   NUM_ETU              INTEGER               not null,
   NUM_CONV             INTEGER               not null,
   DATE_DEBUT           DATE,
   NB_JOURS_ETUDES      INTEGER               DEFAULT 0,
   DATE_FIN             DATE                  DEFAULT NULL,
   constraint PK_PARTICIPATION primary key (NUM_ETU, NUM_CONV, DATE_DEBUT)
);

/*===============================================================*/
/* Index : PARTICIPATION_FK                                      */
/*===============================================================*/
create index PARTICIPATION_FK on PARTICIPATION (
   NUM_ETU ASC
);

/*===============================================================*/
/* Index : PARTICIPATION2_FK                                     */
/*===============================================================*/
create index PARTICIPATION2_FK on PARTICIPATION (
   NUM_CONV ASC
);

/*===============================================================*/
/* Table : REMBOURSEMENT_FRAIS                                   */
/*===============================================================*/
create table REMBOURSEMENT_FRAIS
(
   NUM_RF               INTEGER               GENERATED ALWAYS  AS IDENTITY
START WITH 1 INCREMENT BY 1,
   DATE_RF              DATE,
   MONTANT_RF           NUMBER,
   constraint PK_REMBOURSEMENT_FRAIS primary key (NUM_RF)
);

alter table ACOMPTE
   add constraint FK_ACOMPTE_CORRESPON_CONVENTI foreign key (NUM_CONV)
      references CONVENTION (NUM_CONV);

alter table ACOMPTE
   add constraint FK_ACOMPTE_DEMANDE_ETUDIANT foreign key (NUM_ETU)
      references ETUDIANT (NUM_ETU);

alter table CONVENTION
```

```sql
   add constraint FK_CONVENTI_EST_RESPO_ETUDIANT foreign key (NUM_ETU)
      references ETUDIANT (NUM_ETU);

alter table CONVENTION
   add constraint FK_CONVENTI_ETABLIT_ENTREPRI foreign key (NUM_ENT)
      references ENTREPRISE (NUM_ENT);

alter table EST_ASSIGNE
   add constraint FK_EST_ASSI_EST_ASSIG_CONVENTI foreign key (NUM_CONV)
      references CONVENTION (NUM_CONV);

alter table EST_ASSIGNE
   add constraint FK_EST_ASSI_EST_ASSIG_ETUDIANT foreign key (NUM_ETU)
      references ETUDIANT (NUM_ETU);

alter table FRAIS
   add constraint FK_FRAIS_DANS_LE_C_CONVENTI foreign key (NUM_CONV)
      references CONVENTION (NUM_CONV);

alter table FRAIS
   add constraint FK_FRAIS_DEPEND_REMBOURS foreign key (NUM_RF)
      references REMBOURSEMENT_FRAIS (NUM_RF);

alter table FRAIS
   add constraint FK_FRAIS_ENGENDRE_ETUDIANT foreign key (NUM_ETU)
      references ETUDIANT (NUM_ETU);

alter table INDEMNISATION
   add constraint FK_INDEMNIS_EST_INDEM_ETUDIANT foreign key (NUM_ETU)
      references ETUDIANT (NUM_ETU);

alter table INDEMNISATION
   add constraint FK_INDEMNIS_INDEMNISE_CONVENTI foreign key (NUM_CONV)
      references CONVENTION (NUM_CONV);

alter table PARTICIPATION
   add constraint FK_PARTICIP_PARTICIPA_ETUDIANT foreign key (NUM_ETU)
      references ETUDIANT (NUM_ETU);

alter table PARTICIPATION
   add constraint FK_PARTICIP_PARTICIPA_CONVENTI foreign key (NUM_CONV)
      references CONVENTION (NUM_CONV);
```

Deuxième partie

Contraintes d'intégrité

Pour ajouter les contraintes d'intégrité non prises en charge par le MCD nous avons ajoutés les requêtes ci-dessous après le code généré par le MPD :

```sql
ALTER TABLE CONVENTION
ADD CONSTRAINT CK_CONVENTION_DUREE CHECK ( DUREE_CONV < 6);

ALTER TABLE CONVENTION
ADD CONSTRAINT CK_CONVENTION_PRIX_JOURNEE_ETU CHECK ( PRIX_JOURNEE_ETU > 10 );

ALTER TABLE CONVENTION
ADD CONSTRAINT CK_CONVENTION_RETENUES CHECK ( RETENUES_ENTS < 1 );

ALTER TABLE INDEMNISATION
ADD CONSTRAINT CK_INDEMNISATION_UNIQUE UNIQUE (NUM_CONV, NUM_ETU,
DATE_INDEMNISATION);

ALTER TABLE EST_ASSIGNE
ADD CONSTRAINT CK_EST_ASSIGNE_NB_JOURS_ETUDES CHECK ( NB_JOURS_ETUDES_TOTAL <=
30 );

ALTER TABLE PARTICIPATION
ADD CONSTRAINT CK_PARTICIPATION_NB_JOURS_ETUDES CHECK ( NB_JOURS_ETUDES <= 30
);

ALTER TABLE FRAIS
ADD CONSTRAINT CK_FRAIS_MONTANT CHECK ( MONTANT_FRAIS > 3 );

ALTER TABLE FRAIS
ADD CONSTRAINT CK_FRAIS_TYPE CHECK ( TYPE_FRAIS IN
('déplacement','séjour','autre') );
```

Scénarios pour exploiter la base de données :

Informations globables

Informations sur les entreprises :

Nom Entreprise	N°Entreprise	Adresse Entreprise	Code postal entreprise	Pays entreprise	Tel_ent
Sésame	1	26 rue Henry maréchal	69800	France	0472251022
Urgo	2	42 Rue de Longevic	21300	France	0474550283

Informations sur les étudiants :

Num étudiant	Nom étudiant	Prenom etudiant	Date naissance etudiant	Num Sécurité sociale etudiant	Adresse etudiant	Code postal etudiant	Pays etudiant
1	Rodriguez	Marie	14-03-2003	20148502672	23 Rue Merlin	69800	France
2	Meridi	Linda	23-07-2002	04805276642	4 Rue Mozart	69400	France
3	BoukarI	Hicham	29-06-2003	20004999200	14 Rue Mérieux	69003	France
4	Turoi	Clément	15-04-2001	NULL	126 Rue Jean Macé	69960	France
5	LaD	Koba	10-10-2010	NULL	3 Rue de la Drill	6900	France

Informations sur les projet et l'étudiant responsable :

Date de debut	Duree de l'étude	Nom du projet	Entreprise	Etudiant responsable	prix-journées
05-01-2022	2 mois	Gestion de base de donnée	Sésame	hicham Boukari	12, 0 euros
01-07-2022	3 mois	Promotion du Zolgensma	Urgo	Clément Turoi	18.72 euros
01-07-2022	2 mois	Promotion de	Urgo	Koba la D	12,0 euros

		l'Excilo			

Informations sur les rôles des étudiants dans les projets:

Projets	Etudiants	Nombre de jours de participation
Gestion de projets	Marie Rodriguez	15 jours
	Linda Meridi	30 jours
	BoukarI Hicham	6 jours
Promotion du Zolgensma	Marie Rodriguez	5 jours
	Linda Meridi	20 jours
	BoukarI Hicham	30 jours
	LaD Koba	30 jours
	Turin Clement	15 jours
Promotion de l'Excilo	Marie Rodriguez	20 jours
	Linda Meridi	3 jours
	LaD Koba	19 jours

Les requêtes suivantes sont exécutées dans l'ordre dans lequel elles apparaissent.

Pour pouvoir appliquer les scénarios, nous avons créé les deux entreprises :

```sql
INSERT INTO ENTREPRISE (NOM_ENT, ADRESSE_ENT, CODE_POSTAL_ENT, PAYS_ENT,
TEL_ENT)
VALUES ('Sésame', '26 Rue Henry-Maréchal', '69800', 'FRANCE', '0472251022');

INSERT INTO ENTREPRISE (NOM_ENT, ADRESSE_ENT, CODE_POSTAL_ENT, PAYS_ENT,
TEL_ENT)
VALUES ('Urgo', '42 Rue de longevic', '21300', 'FRANCE', '0474550283');
```

Ainsi que 5 étudiants :

```sql
INSERT INTO ETUDIANT (NOM_ETU, PRENOM_ETU, DATE_NAISSANCE_EUT, NUM_SS_ETU,
ADRESSE_ETU, CODE_POSTAL_ETU, PAYS_ETU)
VALUES ('Rodriguez', 'Marie', '14-03-2003', '20148502672', '23 Rue Merlin',
'69800', 'FRANCE');

INSERT INTO ETUDIANT (NOM_ETU, PRENOM_ETU, DATE_NAISSANCE_EUT, NUM_SS_ETU,
ADRESSE_ETU, CODE_POSTAL_ETU, PAYS_ETU)
VALUES ('Meridi', 'Linda', '23-07-2002', '04805276642', '4 Rue Mozart',
'69400', 'FRANCE');

INSERT INTO ETUDIANT (NOM_ETU, PRENOM_ETU, DATE_NAISSANCE_EUT, NUM_SS_ETU,
ADRESSE_ETU, CODE_POSTAL_ETU, PAYS_ETU)
VALUES ('Boukari', 'Hicham', '29-06-2003', '20004999200', '14 Rue Mérieux',
'69003', 'FRANCE');

INSERT INTO ETUDIANT (NOM_ETU, PRENOM_ETU, DATE_NAISSANCE_EUT, NUM_SS_ETU,
ADRESSE_ETU, CODE_POSTAL_ETU, PAYS_ETU)
VALUES ('Turoi', 'Clément', '15-04-2001', NULL, '126 Rue Jean Macé', '69960',
'FRANCE');

INSERT INTO ETUDIANT (NOM_ETU, PRENOM_ETU, DATE_NAISSANCE_EUT, NUM_SS_ETU,
ADRESSE_ETU, CODE_POSTAL_ETU, PAYS_ETU)
VALUES ('LaD', 'Koba', '10-10-2010', NULL, '3 Rue de la Drill', '69001',
'FRANCE');
```

Premier projet

Le premier est un projet donné par une entreprise nommée Sésame, qui demande à l'association Junior Dev de mener un projet de 2 mois "gestion de base de données".

Requêtes commentées du premier projet

La requête correspondant à la création de la convention est la suivante :

```
INSERT INTO CONVENTION (NUM_ENT, NOM_ETUDE, DATE_CONV, DUREE_CONV, FIN_PREVUE,
PRIX_JOURNEE_ETU, RETENUES_ENTS, NUM_ETU)
VALUES (1, 'Projet gestion de BD', '05-01-2022', 2, '05-03-2022', 12, 0.03, 3);
```

Nous avons ensuite assignés des étudiants à ce projet :

```
INSERT INTO EST_ASSIGNE (NUM_CONV, NUM_ETU, NB_JOURS_ETUDES_TOTAL)
VALUES (1, 1, 15);

INSERT INTO EST_ASSIGNE (NUM_ETU, NUM_CONV, NB_JOURS_ETUDES_TOTAL)
VALUES (2, 1, 30);

INSERT INTO EST_ASSIGNE (NUM_ETU, NUM_CONV, NB_JOURS_ETUDES_TOTAL)
VALUES (3, 1, 6);
```

Puis nous avons imaginé leurs nombres de jours de travail par mois :

```
INSERT INTO PARTICIPATION (NUM_ETU, NUM_CONV, DATE_DEBUT, NB_JOURS_ETUDES,
DATE_FIN)
VALUES (1, 1, '05-01-2022', 15, '04-02-2022');

INSERT INTO PARTICIPATION (NUM_ETU, NUM_CONV, DATE_DEBUT, NB_JOURS_ETUDES,
DATE_FIN)
VALUES (2, 1, '05-01-2022', 15, '04-02-2022');

INSERT INTO PARTICIPATION (NUM_ETU, NUM_CONV, DATE_DEBUT, NB_JOURS_ETUDES,
DATE_FIN)
VALUES (2, 1, '05-02-2022', 15, '05-03-2022');

INSERT INTO PARTICIPATION (NUM_ETU, NUM_CONV, DATE_DEBUT, NB_JOURS_ETUDES,
DATE_FIN)
VALUES (3, 1, '05-02-2022', 6, '05-03-2022');
```

Les requêtes suivantes ont été faites dans un ordre correspondant à l'ordre chronologique du projet.

Nous avons ensuite imaginé des frais et un acompte pour le premier mois :

```
INSERT INTO FRAIS (NUM_CONV, NUM_ETU, DATE_FRAIS, MONTANT_FRAIS, TYPE_FRAIS)
VALUES (1, 3, '07-01-2022', 200.50, 'séjour');
```

```
INSERT INTO FRAIS (NUM_CONV, NUM_ETU, DATE_FRAIS, MONTANT_FRAIS, TYPE_FRAIS)
VALUES (1, 3, '04-02-2022', 50.50, 'déplacement');

INSERT INTO ACOMPTE (NUM_CONV, NUM_ETU, MONTANT_ACOMPTE, DATE_ACOMPTE)
VALUES (1, 1, 50, '25-01-2022');
```

Une fois la fin du premier mois arrivé, voici la requête pour sauvegarder le montant de l'indemnisation pour chaque étudiant :

Voici la requête type :

```
INSERT INTO INDEMNISATION (NUM_CONV, NUM_ETU, DATE_INDEMNISATION,
MONTANT_INDEMNISATION)
VALUES (<numero de la convention>, <numero de l'étudiant>, '05-02-2022', (
        (
            SELECT (P.NB_JOURS_ETUDES * C2.PRIX_JOURNEE_ETU)
            FROM PARTICIPATION P
                    JOIN CONVENTION C2 on C2.NUM_CONV = P.NUM_CONV
            WHERE P.NUM_ETU = <numero de l'étudiant>
              AND P.DATE_FIN = '04-02-2022'
              AND C2.NUM_CONV = <numero de la convention>
        )
        -
        (
            SELECT COALESCE(SUM(A.MONTANT_ACOMPTE), 0)
            FROM ACOMPTE A
                    JOIN CONVENTION C3 on C3.NUM_CONV = A.NUM_CONV
                    JOIN ETUDIANT E on E.NUM_ETU = A.NUM_ETU
            WHERE E.NUM_ETU = <numero de l'étudiant>
              AND A.DATE_ACOMPTE >= '05-01-2022'
              AND C3.NUM_CONV = <numero de la convention>
        )
        -
        (
            SELECT (P.NB_JOURS_ETUDES * C2.PRIX_JOURNEE_ETU) * C2.RETENUES_ENTS
            FROM PARTICIPATION P
                    JOIN CONVENTION C2 on C2.NUM_CONV = P.NUM_CONV
            WHERE P.NUM_ETU = <numero de l'étudiant>
              AND P.DATE_FIN = '04-02-2022'
              AND C2.NUM_CONV = <numero de la convention>
        )
    ));
```

Appliqué sur l'étudiante "Rodriguez Marie" d'identifiant 1, la requête est :

```sql
INSERT INTO INDEMNISATION (NUM_CONV, NUM_ETU, DATE_INDEMNISATION,
MONTANT_INDEMNISATION)
VALUES (1, 1, '05-02-2022', (
        (
            SELECT (P.NB_JOURS_ETUDES * C2.PRIX_JOURNEE_ETU)
            FROM PARTICIPATION P
                    JOIN CONVENTION C2 on C2.NUM_CONV = P.NUM_CONV
            WHERE P.NUM_ETU = 1
              AND P.DATE_FIN = '04-02-2022'
              AND C2.NUM_CONV = 1
        )
        -
        (
            SELECT COALESCE(SUM(A.MONTANT_ACOMPTE), 0)
            FROM ACOMPTE A
                    JOIN CONVENTION C3 on C3.NUM_CONV = A.NUM_CONV
                    JOIN ETUDIANT E on E.NUM_ETU = A.NUM_ETU
            WHERE E.NUM_ETU = 1
              AND A.DATE_ACOMPTE >= '05-01-2022'
              AND C3.NUM_CONV = 1
        )
        -
        (
            SELECT (P.NB_JOURS_ETUDES * C2.PRIX_JOURNEE_ETU) * C2.RETENUES_ENTS
            FROM PARTICIPATION P
                    JOIN CONVENTION C2 on C2.NUM_CONV = P.NUM_CONV
            WHERE P.NUM_ETU = 1
              AND P.DATE_FIN = '04-02-2022'
              AND C2.NUM_CONV = 1
        )
    ));
```

Cette requête est ensuite répété pour les étudiants "Meridi Linda" et "Boukari Hicham" de numéros respectivement 2 et 3;

Nous avons ensuite enregistré le remboursement de frais engendré par :

```sql
INSERT INTO REMBOURSEMENT_FRAIS (DATE_RF, MONTANT_RF)
VALUES ('05-02-2022', (SELECT SUM(MONTANT_FRAIS) FROM FRAIS WHERE NUM_ETU = 3
AND NUM_CONV = 1 AND NUM_RF IS NULL));

UPDATE FRAIS
SET NUM_RF = (SELECT MAX(ROWNUM) FROM REMBOURSEMENT_FRAIS)
```

```
WHERE NUM_ETU = 3;
```

N'ayant pas généré de nouveaux frais ou d'acomptes pour le deuxième mois, en suivant l'ordre chronologique, on passe directement à l'enregistrement du montant de l'indemnisation du deuxième mois.

Enfin pour remplir la facture, nous utilisons la requête suivante :

```
UPDATE CONVENTION
SET DATE_FACTURE = '05-03-2022'
WHERE NUM_CONV = 1;

UPDATE CONVENTION
SET MONTANT_FACTURE = (
    SELECT SUM(A.NB_JOURS_ETUDES_TOTAL * C2.PRIX_JOURNEE_ETU)
    FROM EST_ASSIGNE A
            JOIN CONVENTION C2 on A.NUM_CONV = C2.NUM_CONV
    WHERE A.NUM_CONV = 1
)
WHERE NUM_CONV = 1;
```

Script complet pour le premier scénario

```
-- Ajouter l'entreprise
INSERT INTO ENTREPRISE (NOM_ENT, ADRESSE_ENT, CODE_POSTAL_ENT, PAYS_ENT,
TEL_ENT)
VALUES ('Sésame', '26 Rue Henry-Maréchal', '69800', 'FRANCE', '0472251022');

-- Création des étudiants
INSERT INTO ETUDIANT (NOM_ETU, PRENOM_ETU, DATE_NAISSANCE_EUT, NUM_SS_ETU,
ADRESSE_ETU, CODE_POSTAL_ETU, PAYS_ETU)
VALUES ('Rodriguez', 'Marie', '14-03-2003', '20148502672', '23 Rue Merlin',
'69800', 'FRANCE');

INSERT INTO ETUDIANT (NOM_ETU, PRENOM_ETU, DATE_NAISSANCE_EUT, NUM_SS_ETU,
ADRESSE_ETU, CODE_POSTAL_ETU, PAYS_ETU)
VALUES ('Meridi', 'Linda', '23-07-2002', '04805276642', '4 Rue Mozart',
'69400', 'FRANCE');

INSERT INTO ETUDIANT (NOM_ETU, PRENOM_ETU, DATE_NAISSANCE_EUT, NUM_SS_ETU,
ADRESSE_ETU, CODE_POSTAL_ETU, PAYS_ETU)
VALUES ('Boukari', 'Hicham', '29-06-2003', '20004999200', '14 Rue Mérieux',
'69003', 'FRANCE');

-- Ajouter la convention
INSERT INTO CONVENTION (NUM_ENT, NOM_ETUDE, DATE_CONV, DUREE_CONV, FIN_PREVUE,
```

```sql
                    PRIX_JOURNEE_ETU, RETENUES_ENTS,
                                    NUM_ETU)
VALUES (1, 'Projet gestion de BD', '05-01-2022', 2, '05-03-2022', 12, 0.03, 3);
-- Assignation des étudiants aux projets
INSERT INTO EST_ASSIGNE (NUM_CONV, NUM_ETU, NB_JOURS_ETUDES_TOTAL)
VALUES (1, 1, 15);

INSERT INTO EST_ASSIGNE (NUM_ETU, NUM_CONV, NB_JOURS_ETUDES_TOTAL)
VALUES (2, 1, 30);

INSERT INTO EST_ASSIGNE (NUM_ETU, NUM_CONV, NB_JOURS_ETUDES_TOTAL)
VALUES (3, 1, 6);

INSERT INTO PARTICIPATION (NUM_ETU, NUM_CONV, DATE_DEBUT, NB_JOURS_ETUDES,
DATE_FIN)
VALUES (1, 1, '05-01-2022', 15, '04-02-2022');

INSERT INTO PARTICIPATION (NUM_ETU, NUM_CONV, DATE_DEBUT, NB_JOURS_ETUDES,
DATE_FIN)
VALUES (2, 1, '05-01-2022', 15, '04-02-2022');

INSERT INTO PARTICIPATION (NUM_ETU, NUM_CONV, DATE_DEBUT, NB_JOURS_ETUDES,
DATE_FIN)
VALUES (2, 1, '05-02-2022', 15, '05-03-2022');

INSERT INTO PARTICIPATION (NUM_ETU, NUM_CONV, DATE_DEBUT, NB_JOURS_ETUDES,
DATE_FIN)
VALUES (3, 1, '05-02-2022', 6, '05-03-2022');

-- Ajout de frais
INSERT INTO FRAIS (NUM_CONV, NUM_ETU, DATE_FRAIS, MONTANT_FRAIS, TYPE_FRAIS)
VALUES (1, 3, '07-01-2022', 200.50, 'séjour');

INSERT INTO FRAIS (NUM_CONV, NUM_ETU, DATE_FRAIS, MONTANT_FRAIS, TYPE_FRAIS)
VALUES (1, 3, '04-02-2022', 50.50, 'déplacement');

-- Ajout d'acompte
INSERT INTO ACOMPTE (NUM_CONV, NUM_ETU, MONTANT_ACOMPTE, DATE_ACOMPTE)
VALUES (1, 1, 50, '25-01-2022');

-- Remplir fiche versement d'indémnité du 05/02
INSERT INTO INDEMNISATION (NUM_CONV, NUM_ETU, DATE_INDEMNISATION,
MONTANT_INDEMNISATION)
VALUES (1, 1, '05-02-2022', (
        (
            SELECT (P.NB_JOURS_ETUDES * C2.PRIX_JOURNEE_ETU)
```

```sql
            FROM PARTICIPATION P
                    JOIN CONVENTION C2 on C2.NUM_CONV = P.NUM_CONV
            WHERE P.NUM_ETU = 1
              AND P.DATE_FIN = '04-02-2022'
              AND C2.NUM_CONV = 1
        )
        -
        (
            SELECT COALESCE(SUM(A.MONTANT_ACOMPTE), 0)
            FROM ACOMPTE A
                    JOIN CONVENTION C3 on C3.NUM_CONV = A.NUM_CONV
                    JOIN ETUDIANT E on E.NUM_ETU = A.NUM_ETU
            WHERE E.NUM_ETU = 1
              AND A.DATE_ACOMPTE >= '05-01-2022'
              AND C3.NUM_CONV = 1
        )
        -
        (
            SELECT (P.NB_JOURS_ETUDES * C2.PRIX_JOURNEE_ETU) * C2.RETENUES_ENTS
            FROM PARTICIPATION P
                    JOIN CONVENTION C2 on C2.NUM_CONV = P.NUM_CONV
            WHERE P.NUM_ETU = 1
              AND P.DATE_FIN = '04-02-2022'
              AND C2.NUM_CONV = 1
        )
    ));

INSERT INTO INDEMNISATION (NUM_CONV, NUM_ETU, DATE_INDEMNISATION,
MONTANT_INDEMNISATION)
VALUES (1, 2, '05-02-2022', (
        (
            SELECT (P.NB_JOURS_ETUDES * C2.PRIX_JOURNEE_ETU)
            FROM PARTICIPATION P
                    JOIN CONVENTION C2 on C2.NUM_CONV = P.NUM_CONV
            WHERE P.NUM_ETU = 2
              AND P.DATE_FIN = '04-02-2022'
              AND C2.NUM_CONV = 1
        )
        -
        (
            SELECT COALESCE(SUM(A.MONTANT_ACOMPTE), 0)
            FROM ACOMPTE A
                    JOIN CONVENTION C3 on C3.NUM_CONV = A.NUM_CONV
                    JOIN ETUDIANT E on E.NUM_ETU = A.NUM_ETU
            WHERE E.NUM_ETU = 2
              AND A.DATE_ACOMPTE >= '05-01-2022'
```

```sql
                    AND C3.NUM_CONV = 1
            )
            -
            (
                SELECT (P.NB_JOURS_ETUDES * C2.PRIX_JOURNEE_ETU) * C2.RETENUES_ENTS
                FROM PARTICIPATION P
                        JOIN CONVENTION C2 on C2.NUM_CONV = P.NUM_CONV
                WHERE P.NUM_ETU = 2
                  AND P.DATE_FIN = '04-02-2022'
                  AND C2.NUM_CONV = 1
            )
        ));

INSERT INTO INDEMNISATION (NUM_CONV, NUM_ETU, DATE_INDEMNISATION,
MONTANT_INDEMNISATION)
VALUES (1, 3, '05-02-2022', (
            (
                SELECT (P.NB_JOURS_ETUDES * C2.PRIX_JOURNEE_ETU)
                FROM PARTICIPATION P
                        JOIN CONVENTION C2 on C2.NUM_CONV = P.NUM_CONV
                WHERE P.NUM_ETU = 3
                  AND P.DATE_FIN = '04-02-2022'
                  AND C2.NUM_CONV = 1
            )
            -
            (
                SELECT COALESCE(SUM(A.MONTANT_ACOMPTE), 0)
                FROM ACOMPTE A
                        JOIN CONVENTION C3 on C3.NUM_CONV = A.NUM_CONV
                        JOIN ETUDIANT E on E.NUM_ETU = A.NUM_ETU
                WHERE E.NUM_ETU = 3
                  AND A.DATE_ACOMPTE >= '05-01-2022'
                  AND C3.NUM_CONV = 1
            )
            -
            (
                SELECT (P.NB_JOURS_ETUDES * C2.PRIX_JOURNEE_ETU) * C2.RETENUES_ENTS
                FROM PARTICIPATION P
                        JOIN CONVENTION C2 on C2.NUM_CONV = P.NUM_CONV
                WHERE P.NUM_ETU = 3
                  AND P.DATE_FIN = '04-02-2022'
                  AND C2.NUM_CONV = 1
            )
        ));

-- Remplir fiche remboursement de frais du 05/02
```

```sql
INSERT INTO REMBOURSEMENT_FRAIS (DATE_RF, MONTANT_RF)
VALUES ('05-02-2022', (SELECT SUM(MONTANT_FRAIS) FROM FRAIS WHERE NUM_ETU = 3
AND NUM_CONV = 1 AND NUM_RF IS NULL));
UPDATE FRAIS
SET NUM_RF = (SELECT MAX(ROWNUM) FROM REMBOURSEMENT_FRAIS)
WHERE NUM_ETU = 3;

-- Remplir fiche versement d'indémnité du 05/03
INSERT INTO INDEMNISATION (NUM_CONV, NUM_ETU, DATE_INDEMNISATION,
MONTANT_INDEMNISATION)
VALUES (1, 1, '05-03-2022', (
        (
            SELECT (P.NB_JOURS_ETUDES * C2.PRIX_JOURNEE_ETU)
            FROM PARTICIPATION P
                    JOIN CONVENTION C2 on C2.NUM_CONV = P.NUM_CONV
            WHERE P.NUM_ETU = 1
              AND P.DATE_FIN = '05-03-2022'
              AND C2.NUM_CONV = 1
        )
        -
        (
            SELECT COALESCE(SUM(A.MONTANT_ACOMPTE), 0)
            FROM ACOMPTE A
                    JOIN CONVENTION C3 on C3.NUM_CONV = A.NUM_CONV
                    JOIN ETUDIANT E on E.NUM_ETU = A.NUM_ETU
            WHERE E.NUM_ETU = 1
              AND (A.DATE_ACOMPTE >= '05-02-2022')
              AND (C3.NUM_CONV = 1)
        )
        -
        (
            SELECT (P.NB_JOURS_ETUDES * C2.PRIX_JOURNEE_ETU) * C2.RETENUES_ENTS
            FROM PARTICIPATION P
                    JOIN CONVENTION C2 on C2.NUM_CONV = P.NUM_CONV
            WHERE P.NUM_ETU = 1
              AND P.DATE_FIN = '05-03-2022'
              AND C2.NUM_CONV = 1
        )
    ));

INSERT INTO INDEMNISATION (NUM_CONV, NUM_ETU, DATE_INDEMNISATION,
MONTANT_INDEMNISATION)
VALUES (1, 2, '05-03-2022', (
        (
            SELECT (P.NB_JOURS_ETUDES * C2.PRIX_JOURNEE_ETU)
            FROM PARTICIPATION P
```

```sql
                    JOIN CONVENTION C2 on C2.NUM_CONV = P.NUM_CONV
            WHERE P.NUM_ETU = 2
              AND P.DATE_FIN = '05-03-2022'
              AND C2.NUM_CONV = 1
        )
        -
        (
            SELECT COALESCE(SUM(A.MONTANT_ACOMPTE), 0)
            FROM ACOMPTE A
                     JOIN CONVENTION C3 on C3.NUM_CONV = A.NUM_CONV
                     JOIN ETUDIANT E on E.NUM_ETU = A.NUM_ETU
            WHERE E.NUM_ETU = 2
              AND (A.DATE_ACOMPTE >= '05-02-2022')
              AND C3.NUM_CONV = 1
        )
        -
        (
            SELECT (P.NB_JOURS_ETUDES * C2.PRIX_JOURNEE_ETU) * C2.RETENUES_ENTS
            FROM PARTICIPATION P
                     JOIN CONVENTION C2 on C2.NUM_CONV = P.NUM_CONV
            WHERE P.NUM_ETU = 2
              AND P.DATE_FIN = '05-03-2022'
              AND C2.NUM_CONV = 1
        )
    ));

INSERT INTO INDEMNISATION (NUM_CONV, NUM_ETU, DATE_INDEMNISATION,
MONTANT_INDEMNISATION)
VALUES (1, 3, '05-03-2022', (
        (
            SELECT (P.NB_JOURS_ETUDES * C2.PRIX_JOURNEE_ETU)
            FROM PARTICIPATION P
                     JOIN CONVENTION C2 on C2.NUM_CONV = P.NUM_CONV
            WHERE P.NUM_ETU = 3
              AND P.DATE_FIN = '05-03-2022'
              AND C2.NUM_CONV = 1
        )
        -
        (
            SELECT COALESCE(SUM(A.MONTANT_ACOMPTE), 0)
            FROM ACOMPTE A
                     JOIN CONVENTION C3 on C3.NUM_CONV = A.NUM_CONV
                     JOIN ETUDIANT E on E.NUM_ETU = A.NUM_ETU
            WHERE E.NUM_ETU = 3
              AND (A.DATE_ACOMPTE >= '05-02-2022')
              AND C3.NUM_CONV = 1
```

```
            )
            -
            (
                SELECT (P.NB_JOURS_ETUDES * C2.PRIX_JOURNEE_ETU) * C2.RETENUES_ENTS
                FROM PARTICIPATION P
                            JOIN CONVENTION C2 on C2.NUM_CONV = P.NUM_CONV
                WHERE P.NUM_ETU = 3
                    AND P.DATE_FIN = '05-03-2022'
                    AND C2.NUM_CONV = 1
            )
        ));


UPDATE INDEMNISATION
SET MONTANT_INDEMNISATION = 0
WHERE MONTANT_INDEMNISATION IS NULL;

-- Remplir Facture
UPDATE CONVENTION
SET DATE_FACTURE = '05-03-2022'
WHERE NUM_CONV = 1;

UPDATE CONVENTION
SET MONTANT_FACTURE = (
    SELECT SUM(A.NB_JOURS_ETUDES_TOTAL * C2.PRIX_JOURNEE_ETU)
    FROM EST_ASSIGNE A
            JOIN CONVENTION C2 on A.NUM_CONV = C2.NUM_CONV
    WHERE A.NUM_CONV = 1
)
WHERE NUM_CONV = 1;

commit;
```

Deuxième projet

Le deuxième projet nommé "Promotion du Zolgensma" a été commandé par l'entreprise Urgo. Ce projet dure 3 mois et n'est pas encore terminé.

Script complet pour le deuxième scénario

```
 -- Ajouter une entreprise
INSERT INTO ENTREPRISE (NOM_ENT, ADRESSE_ENT, CODE_POSTAL_ENT, PAYS_ENT,
TEL_ENT)
VALUES ('Urgo', '42 Rue de longevic', '21300', 'FRANCE', '0474550283');
```

```sql
-- Création d'étudiants
INSERT INTO ETUDIANT (NOM_ETU, PRENOM_ETU, DATE_NAISSANCE_EUT, NUM_SS_ETU,
ADRESSE_ETU, CODE_POSTAL_ETU, PAYS_ETU)
VALUES ('Turoi', 'Clément', '15-04-2001', NULL, '126 Rue Jean Macé', '69960',
'FRANCE');

INSERT INTO ETUDIANT (NOM_ETU, PRENOM_ETU, DATE_NAISSANCE_EUT, NUM_SS_ETU,
ADRESSE_ETU, CODE_POSTAL_ETU, PAYS_ETU)
VALUES ('LaD', 'Koba', '10-10-2010', NULL, '3 Rue de la Drill', '69001',
'FRANCE');
commit;

-- Création de convention
INSERT INTO CONVENTION (NUM_ETU, NUM_ENT, NOM_ETUDE, DATE_CONV, DUREE_CONV,
FIN_PREVUE, PRIX_JOURNEE_ETU,
                        RETENUES_ENTS)
VALUES (5, 2, 'Promotion du Zolgensma', '01-07-2022', 3, '01-10-2022', 18.72,
0.05);

-- Assignation des étudiants du premier projet
INSERT INTO EST_ASSIGNE (NUM_ETU, NUM_CONV, NB_JOURS_ETUDES_TOTAL)
VALUES (5, 2, 30);

INSERT INTO PARTICIPATION (NUM_ETU, NUM_CONV, DATE_DEBUT, NB_JOURS_ETUDES,
DATE_FIN)
VALUES (5, 2, '01-07-2022', 5, '31-07-2022');

INSERT INTO PARTICIPATION (NUM_ETU, NUM_CONV, DATE_DEBUT, NB_JOURS_ETUDES,
DATE_FIN)
VALUES (5, 2, '01-08-2022', 25, '31-08-2022');

INSERT INTO EST_ASSIGNE (NUM_ETU, NUM_CONV, NB_JOURS_ETUDES_TOTAL)
VALUES (1, 2, 5);

INSERT INTO PARTICIPATION (NUM_ETU, NUM_CONV, DATE_DEBUT, NB_JOURS_ETUDES,
DATE_FIN)
VALUES (1, 2, '01-08-2022', 5, '31-08-2022');

INSERT INTO EST_ASSIGNE (NUM_ETU, NUM_CONV, NB_JOURS_ETUDES_TOTAL)
VALUES (2, 2, 20);

INSERT INTO PARTICIPATION (NUM_ETU, NUM_CONV, DATE_DEBUT, NB_JOURS_ETUDES,
DATE_FIN)
VALUES (2, 2, '01-07-2022', 13, '31-07-2022');
```

```sql
INSERT INTO PARTICIPATION (NUM_ETU, NUM_CONV, DATE_DEBUT, NB_JOURS_ETUDES,
DATE_FIN)
VALUES (2, 2, '01-08-2022', 7, '31-08-2022');

INSERT INTO EST_ASSIGNE (NUM_ETU, NUM_CONV, NB_JOURS_ETUDES_TOTAL)
VALUES (3, 2, 30);

INSERT INTO PARTICIPATION (NUM_ETU, NUM_CONV, DATE_DEBUT, NB_JOURS_ETUDES,
DATE_FIN)
VALUES (3, 2, '01-08-2022', 30, '31-08-2022');

INSERT INTO EST_ASSIGNE (NUM_ETU, NUM_CONV, NB_JOURS_ETUDES_TOTAL)
VALUES (4, 2, 15);

INSERT INTO PARTICIPATION (NUM_ETU, NUM_CONV, DATE_DEBUT, NB_JOURS_ETUDES,
DATE_FIN)
VALUES (4, 2, '01-07-2022', 15, '31-07-2022');

-- Ajout de frais du premier mois du projet 1
INSERT INTO FRAIS (NUM_CONV, NUM_ETU, DATE_FRAIS, MONTANT_FRAIS, TYPE_FRAIS)
VALUES (2, 5, '24-07-2022', 100, 'séjour');

-- Remplir la fiche de versement d'indémnité du 01/08
INSERT INTO INDEMNISATION (NUM_CONV, NUM_ETU, DATE_INDEMNISATION,
MONTANT_INDEMNISATION)
VALUES (2, 5, '01-08-2022',
        (
            (
                SELECT (P.NB_JOURS_ETUDES * C2.PRIX_JOURNEE_ETU)
                FROM PARTICIPATION P
                        JOIN CONVENTION C2 on C2.NUM_CONV = P.NUM_CONV
                WHERE P.NUM_CONV = 2
                  AND P.DATE_FIN = '31-07-2022'
                  AND P.NUM_ETU = 5
            )
            -
            (
                SELECT COALESCE(SUM(A.MONTANT_ACOMPTE), 0)
                FROM ACOMPTE A
                        JOIN CONVENTION C3 on C3.NUM_CONV = A.NUM_CONV
                        JOIN ETUDIANT E on E.NUM_ETU = A.NUM_ETU
                WHERE C3.NUM_CONV = 2
                  AND E.NUM_ETU = 5
                  AND A.DATE_ACOMPTE >= '01-07-2022'
            )
            -
```

```sql
                (
                    SELECT (P.NB_JOURS_ETUDES * C2.PRIX_JOURNEE_ETU) *
C2.RETENUES_ENTS
                    FROM PARTICIPATION P
                            JOIN CONVENTION C2 on C2.NUM_CONV = P.NUM_CONV
                    WHERE P.NUM_CONV = 2
                      AND P.DATE_FIN = '31-07-2022'
                      AND P.NUM_ETU = 5
                )
            ));

INSERT INTO INDEMNISATION (NUM_CONV, NUM_ETU, DATE_INDEMNISATION,
MONTANT_INDEMNISATION)
VALUES (2, 4, '01-08-2022',
        (
            (
                SELECT (P.NB_JOURS_ETUDES * C2.PRIX_JOURNEE_ETU)
                FROM PARTICIPATION P
                        JOIN CONVENTION C2 on C2.NUM_CONV = P.NUM_CONV
                WHERE P.NUM_CONV = 2
                  AND P.DATE_FIN = '31-07-2022'
                  AND P.NUM_ETU = 4
            )
            -
            (
                SELECT COALESCE(SUM(A.MONTANT_ACOMPTE), 0)
                FROM ACOMPTE A
                        JOIN CONVENTION C3 on C3.NUM_CONV = A.NUM_CONV
                        JOIN ETUDIANT E on E.NUM_ETU = A.NUM_ETU
                WHERE C3.NUM_CONV = 2
                  AND E.NUM_ETU = 4
                  AND A.DATE_ACOMPTE >= '01-07-2022'
            )
            -
            (
                SELECT (P.NB_JOURS_ETUDES * C2.PRIX_JOURNEE_ETU) *
C2.RETENUES_ENTS
                FROM PARTICIPATION P
                        JOIN CONVENTION C2 on C2.NUM_CONV = P.NUM_CONV
                WHERE P.NUM_CONV = 2
                  AND P.DATE_FIN = '31-07-2022'
                  AND P.NUM_ETU = 4
            )
        ));

INSERT INTO INDEMNISATION (NUM_CONV, NUM_ETU, DATE_INDEMNISATION,
```

```sql
MONTANT_INDEMNISATION)
VALUES (2, 3, '01-08-2022',
        (
            (
                SELECT (P.NB_JOURS_ETUDES * C2.PRIX_JOURNEE_ETU)
                FROM PARTICIPATION P
                        JOIN CONVENTION C2 on C2.NUM_CONV = P.NUM_CONV
                WHERE P.NUM_CONV = 2
                  AND P.DATE_FIN = '31-07-2022'
                  AND P.NUM_ETU = 3
            )
            -
            (
                SELECT COALESCE(SUM(A.MONTANT_ACOMPTE), 0)
                FROM ACOMPTE A
                        JOIN CONVENTION C3 on C3.NUM_CONV = A.NUM_CONV
                        JOIN ETUDIANT E on E.NUM_ETU = A.NUM_ETU
                WHERE C3.NUM_CONV = 2
                  AND E.NUM_ETU = 3
                  AND A.DATE_ACOMPTE >= '01-07-2022'
            )
            -
            (
                SELECT (P.NB_JOURS_ETUDES * C2.PRIX_JOURNEE_ETU) *
C2.RETENUES_ENTS
                FROM PARTICIPATION P
                        JOIN CONVENTION C2 on C2.NUM_CONV = P.NUM_CONV
                WHERE P.NUM_CONV = 2
                  AND P.DATE_FIN = '31-07-2022'
                  AND P.NUM_ETU = 3
            )
        ));

INSERT INTO INDEMNISATION (NUM_CONV, NUM_ETU, DATE_INDEMNISATION,
MONTANT_INDEMNISATION)
VALUES (2, 2, '01-08-2022',
        (
            (
                SELECT (P.NB_JOURS_ETUDES * C2.PRIX_JOURNEE_ETU)
                FROM PARTICIPATION P
                        JOIN CONVENTION C2 on C2.NUM_CONV = P.NUM_CONV
                WHERE P.NUM_CONV = 2
                  AND P.DATE_FIN = '31-07-2022'
                  AND P.NUM_ETU = 2
            )
            -
```

```sql
            (
                SELECT COALESCE(SUM(A.MONTANT_ACOMPTE), 0)
                FROM ACOMPTE A
                        JOIN CONVENTION C3 on C3.NUM_CONV = A.NUM_CONV
                        JOIN ETUDIANT E on E.NUM_ETU = A.NUM_ETU
                WHERE C3.NUM_CONV = 2
                  AND E.NUM_ETU = 2
                  AND A.DATE_ACOMPTE >= '01-07-2022'
            )
            -
            (
                SELECT (P.NB_JOURS_ETUDES * C2.PRIX_JOURNEE_ETU) *
C2.RETENUES_ENTS
                FROM PARTICIPATION P
                        JOIN CONVENTION C2 on C2.NUM_CONV = P.NUM_CONV
                WHERE P.NUM_CONV = 2
                  AND P.DATE_FIN = '31-07-2022'
                  AND P.NUM_ETU = 2
            )
        ));

INSERT INTO INDEMNISATION (NUM_CONV, NUM_ETU, DATE_INDEMNISATION,
MONTANT_INDEMNISATION)
VALUES (2, 1, '01-08-2022',
        (
            (
                SELECT (P.NB_JOURS_ETUDES * C2.PRIX_JOURNEE_ETU)
                FROM PARTICIPATION P
                        JOIN CONVENTION C2 on C2.NUM_CONV = P.NUM_CONV
                WHERE P.NUM_CONV = 2
                  AND P.DATE_FIN = '31-07-2022'
                  AND P.NUM_ETU = 1
            )
            -
            (
                SELECT COALESCE(SUM(A.MONTANT_ACOMPTE), 0)
                FROM ACOMPTE A
                        JOIN CONVENTION C3 on C3.NUM_CONV = A.NUM_CONV
                        JOIN ETUDIANT E on E.NUM_ETU = A.NUM_ETU
                WHERE C3.NUM_CONV = 2
                  AND E.NUM_ETU = 1
                  AND A.DATE_ACOMPTE >= '01-07-2022'
            )
            -
            (
                SELECT (P.NB_JOURS_ETUDES * C2.PRIX_JOURNEE_ETU) *
```

```sql
                C2.RETENUES_ENTS
                        FROM PARTICIPATION P
                                JOIN CONVENTION C2 on C2.NUM_CONV = P.NUM_CONV
                        WHERE P.NUM_CONV = 2
                          AND P.DATE_FIN = '31-07-2022'
                          AND P.NUM_ETU = 1
                    )
                ));

UPDATE INDEMNISATION
SET MONTANT_INDEMNISATION = 0
WHERE MONTANT_INDEMNISATION IS NULL;

-- Remplir Remboursement des frais 01/08
INSERT INTO REMBOURSEMENT_FRAIS (DATE_RF, MONTANT_RF)
VALUES ('01-08-2022', (SELECT SUM(F.MONTANT_FRAIS)
                        FROM FRAIS F
                        WHERE F.NUM_ETU = 5
                          AND F.NUM_CONV = 2
                          AND F.NUM_RF IS NULL));

UPDATE FRAIS
SET NUM_RF = (SELECT MAX(NUM_RF) FROM REMBOURSEMENT_FRAIS)
WHERE NUM_ETU = 5
  AND NUM_CONV = 2
  AND NUM_RF IS NULL;

-- Ajout de frais du deuxième mois
INSERT INTO FRAIS (NUM_CONV, NUM_ETU, DATE_FRAIS, MONTANT_FRAIS, TYPE_FRAIS)
VALUES (2, 1, '05-08-2022', 50, 'séjour');

INSERT INTO FRAIS (NUM_CONV, NUM_ETU, DATE_FRAIS, MONTANT_FRAIS, TYPE_FRAIS)
VALUES (2, 2, '10-08-2022', 162, 'autre');

-- Ajout des acomptes du deuxième mois
INSERT INTO ACOMPTE (NUM_CONV, NUM_ETU, MONTANT_ACOMPTE, DATE_ACOMPTE)
VALUES (2, 4, 207, '27-08-2022');

-- Remplir la fiche de versement d'indémnité du 01/09
INSERT INTO INDEMNISATION (NUM_CONV, NUM_ETU, DATE_INDEMNISATION,
MONTANT_INDEMNISATION)
VALUES (2, 5, '01-09-2022',
        (
            (
                SELECT (P.NB_JOURS_ETUDES * C2.PRIX_JOURNEE_ETU)
                FROM PARTICIPATION P
```

```sql
                        JOIN CONVENTION C2 on C2.NUM_CONV = P.NUM_CONV
                WHERE P.NUM_CONV = 2
                  AND P.DATE_FIN = '31-08-2022'
                  AND P.NUM_ETU = 5
            )
            -
            (
                SELECT COALESCE(SUM(A.MONTANT_ACOMPTE), 0)
                FROM ACOMPTE A
                        JOIN CONVENTION C3 on C3.NUM_CONV = A.NUM_CONV
                        JOIN ETUDIANT E on E.NUM_ETU = A.NUM_ETU
                WHERE C3.NUM_CONV = 2
                  AND E.NUM_ETU = 5
                  AND A.DATE_ACOMPTE >= '01-08-2022'
            )
            -
            (
                SELECT (P.NB_JOURS_ETUDES * C2.PRIX_JOURNEE_ETU) *
C2.RETENUES_ENTS
                FROM PARTICIPATION P
                        JOIN CONVENTION C2 on C2.NUM_CONV = P.NUM_CONV
                WHERE P.NUM_CONV = 2
                  AND P.DATE_FIN = '31-08-2022'
                  AND P.NUM_ETU = 5
            )
        ));

INSERT INTO INDEMNISATION (NUM_CONV, NUM_ETU, DATE_INDEMNISATION,
MONTANT_INDEMNISATION)
VALUES (2, 4, '01-09-2022',
        (
            (
                SELECT (P.NB_JOURS_ETUDES * C2.PRIX_JOURNEE_ETU)
                FROM PARTICIPATION P
                        JOIN CONVENTION C2 on C2.NUM_CONV = P.NUM_CONV
                WHERE P.NUM_CONV = 2
                  AND P.DATE_FIN = '31-08-2022'
                  AND P.NUM_ETU = 4
            )
            -
            (
                SELECT COALESCE(SUM(A.MONTANT_ACOMPTE), 0)
                FROM ACOMPTE A
                        JOIN CONVENTION C3 on C3.NUM_CONV = A.NUM_CONV
                        JOIN ETUDIANT E on E.NUM_ETU = A.NUM_ETU
                WHERE C3.NUM_CONV = 2
```

```sql
                    AND E.NUM_ETU = 4
                    AND A.DATE_ACOMPTE >= '01-08-2022'
                )
                -
                (
                    SELECT (P.NB_JOURS_ETUDES * C2.PRIX_JOURNEE_ETU) *
C2.RETENUES_ENTS
                    FROM PARTICIPATION P
                            JOIN CONVENTION C2 on C2.NUM_CONV = P.NUM_CONV
                    WHERE P.NUM_CONV = 2
                      AND P.DATE_FIN = '31-08-2022'
                      AND P.NUM_ETU = 4
                )
            ));

INSERT INTO INDEMNISATION (NUM_CONV, NUM_ETU, DATE_INDEMNISATION,
MONTANT_INDEMNISATION)
VALUES (2, 3, '01-09-2022',
        (
            (
                SELECT (P.NB_JOURS_ETUDES * C2.PRIX_JOURNEE_ETU)
                FROM PARTICIPATION P
                        JOIN CONVENTION C2 on C2.NUM_CONV = P.NUM_CONV
                WHERE P.NUM_CONV = 2
                  AND P.DATE_FIN = '31-08-2022'
                  AND P.NUM_ETU = 3
            )
            -
            (
                SELECT COALESCE(SUM(A.MONTANT_ACOMPTE), 0)
                FROM ACOMPTE A
                        JOIN CONVENTION C3 on C3.NUM_CONV = A.NUM_CONV
                        JOIN ETUDIANT E on E.NUM_ETU = A.NUM_ETU
                WHERE C3.NUM_CONV = 2
                  AND E.NUM_ETU = 3
                  AND A.DATE_ACOMPTE >= '01-08-2022'
            )
            -
            (
                SELECT (P.NB_JOURS_ETUDES * C2.PRIX_JOURNEE_ETU) *
C2.RETENUES_ENTS
                FROM PARTICIPATION P
                        JOIN CONVENTION C2 on C2.NUM_CONV = P.NUM_CONV
                WHERE P.NUM_CONV = 2
                  AND P.DATE_FIN = '31-08-2022'
                  AND P.NUM_ETU = 3
```

```sql
            )
        ));

INSERT INTO INDEMNISATION (NUM_CONV, NUM_ETU, DATE_INDEMNISATION,
MONTANT_INDEMNISATION)
VALUES (2, 2, '01-09-2022',
        (
            (
                SELECT (P.NB_JOURS_ETUDES * C2.PRIX_JOURNEE_ETU)
                FROM PARTICIPATION P
                        JOIN CONVENTION C2 on C2.NUM_CONV = P.NUM_CONV
                WHERE P.NUM_CONV = 2
                  AND P.DATE_FIN = '31-08-2022'
                  AND P.NUM_ETU = 2
            )
            -
            (
                SELECT COALESCE(SUM(A.MONTANT_ACOMPTE), 0)
                FROM ACOMPTE A
                        JOIN CONVENTION C3 on C3.NUM_CONV = A.NUM_CONV
                        JOIN ETUDIANT E on E.NUM_ETU = A.NUM_ETU
                WHERE C3.NUM_CONV = 2
                  AND E.NUM_ETU = 2
                  AND A.DATE_ACOMPTE >= '01-08-2022'
            )
            -
            (
                SELECT (P.NB_JOURS_ETUDES * C2.PRIX_JOURNEE_ETU) *
C2.RETENUES_ENTS
                FROM PARTICIPATION P
                        JOIN CONVENTION C2 on C2.NUM_CONV = P.NUM_CONV
                WHERE P.NUM_CONV = 2
                  AND P.DATE_FIN = '31-08-2022'
                  AND P.NUM_ETU = 2
            )
        ));

INSERT INTO INDEMNISATION (NUM_CONV, NUM_ETU, DATE_INDEMNISATION,
MONTANT_INDEMNISATION)
VALUES (2, 1, '01-09-2022',
        (
            (
                SELECT (P.NB_JOURS_ETUDES * C2.PRIX_JOURNEE_ETU)
                FROM PARTICIPATION P
                        JOIN CONVENTION C2 on C2.NUM_CONV = P.NUM_CONV
                WHERE P.NUM_CONV = 2
```

```sql
                            AND P.DATE_FIN = '31-08-2022'
                            AND P.NUM_ETU = 1
                )
                -
                (
                    SELECT COALESCE(SUM(A.MONTANT_ACOMPTE), 0)
                    FROM ACOMPTE A
                            JOIN CONVENTION C3 on C3.NUM_CONV = A.NUM_CONV
                            JOIN ETUDIANT E on E.NUM_ETU = A.NUM_ETU
                    WHERE C3.NUM_CONV = 2
                      AND E.NUM_ETU = 1
                      AND A.DATE_ACOMPTE >= '01-08-2022'
                )
                -
                (
                    SELECT (P.NB_JOURS_ETUDES * C2.PRIX_JOURNEE_ETU) *
C2.RETENUES_ENTS
                    FROM PARTICIPATION P
                            JOIN CONVENTION C2 on C2.NUM_CONV = P.NUM_CONV
                    WHERE P.NUM_CONV = 2
                      AND P.DATE_FIN = '31-08-2022'
                      AND P.NUM_ETU = 1
                )
            ));

UPDATE INDEMNISATION
SET MONTANT_INDEMNISATION = 0
WHERE MONTANT_INDEMNISATION IS NULL;

-- Remplir Remboursement des frais 01/09
INSERT INTO REMBOURSEMENT_FRAIS (DATE_RF, MONTANT_RF)
VALUES ('01-09-2022', (SELECT SUM(F.MONTANT_FRAIS)
                       FROM FRAIS F
                       WHERE F.NUM_ETU = 1
                         AND F.NUM_CONV = 2
                         AND F.NUM_RF IS NULL));

UPDATE FRAIS
SET NUM_RF = (SELECT MAX(NUM_RF) FROM REMBOURSEMENT_FRAIS)
WHERE NUM_ETU = 1
  AND NUM_CONV = 2
  AND NUM_RF IS NULL;

INSERT INTO REMBOURSEMENT_FRAIS (DATE_RF, MONTANT_RF)
VALUES ('01-09-2022', (SELECT SUM(F.MONTANT_FRAIS)
                       FROM FRAIS F
```

```
                              WHERE F.NUM_ETU = 2
                                AND F.NUM_CONV = 2
                                AND F.NUM_RF IS NULL));

UPDATE FRAIS
SET NUM_RF = (SELECT MAX(NUM_RF) FROM REMBOURSEMENT_FRAIS)
WHERE NUM_ETU = 2
  AND NUM_CONV = 2
  AND NUM_RF IS NULL;



-- Ajout de frais pour le 3ème mois
INSERT INTO FRAIS (NUM_CONV, NUM_ETU, DATE_FRAIS, MONTANT_FRAIS, TYPE_FRAIS)
VALUES (2, 1, '10-09-2022', 100, 'déplacement');

INSERT INTO FRAIS (NUM_CONV, NUM_ETU, DATE_FRAIS, MONTANT_FRAIS, TYPE_FRAIS)
VALUES (2, 1, '10-09-2022', 50, 'séjour');

INSERT INTO ACOMPTE (NUM_CONV, NUM_ETU, MONTANT_ACOMPTE, DATE_ACOMPTE)
VALUES (2, 2, 120,'21-09-2022');
commit;
```

Troisième projet

Le troisième projet nommé "Promotion de l'Excilor" a été commandé par l'entreprise Urgo. Ce projet dure 2 mois et se passe en même temps que le deuxième projet

Script complet du troisième projet

```
-- Création de la convention
INSERT INTO CONVENTION (NUM_ETU, NUM_ENT, NOM_ETUDE, DATE_CONV, DUREE_CONV,
FIN_PREVUE, PRIX_JOURNEE_ETU, RETENUES_ENTS)
VALUES (4,2,'Promotion de l''Excilor','01-07-2022',2,'01-09-2022',12,0.01);

-- Assignation des étudiants du deuxième projet
INSERT INTO EST_ASSIGNE (NUM_ETU, NUM_CONV, NB_JOURS_ETUDES_TOTAL)
VALUES (4,3,30);

INSERT INTO PARTICIPATION (NUM_ETU, NUM_CONV, DATE_DEBUT, NB_JOURS_ETUDES,
DATE_FIN)
VALUES (4, 3, '01-07-2022', 30, '31-07-2022');

INSERT INTO EST_ASSIGNE (NUM_ETU, NUM_CONV, NB_JOURS_ETUDES_TOTAL)
VALUES (1,3,20);
```

```sql
INSERT INTO PARTICIPATION (NUM_ETU, NUM_CONV, DATE_DEBUT, NB_JOURS_ETUDES,
DATE_FIN)
VALUES (1,3, '01-07-2022', 20, '31-07-2022');

INSERT INTO EST_ASSIGNE (NUM_ETU, NUM_CONV, NB_JOURS_ETUDES_TOTAL)
VALUES (2,3,3);

INSERT INTO PARTICIPATION (NUM_ETU, NUM_CONV, DATE_DEBUT, NB_JOURS_ETUDES,
DATE_FIN)
VALUES (2,3, '01-08-2022', 3, '31-08-2022');

INSERT INTO EST_ASSIGNE (NUM_ETU, NUM_CONV, NB_JOURS_ETUDES_TOTAL)
VALUES (5,3,19);

INSERT INTO PARTICIPATION (NUM_ETU, NUM_CONV, DATE_DEBUT, NB_JOURS_ETUDES,
DATE_FIN)
VALUES (5,3, '01-08-2022', 3, '31-08-2022');

-- Ajout des frais du premier mois
INSERT INTO FRAIS (NUM_CONV, NUM_ETU, DATE_FRAIS, MONTANT_FRAIS, TYPE_FRAIS)
VALUES (3, 3, '14-07-2022', 110.75, 'séjour');

-- Remplir les fiches d'indémnités 01/08
INSERT INTO INDEMNISATION (NUM_CONV, NUM_ETU, DATE_INDEMNISATION,
MONTANT_INDEMNISATION)
VALUES (3, 1, '01-08-2022',
        (
            (
                SELECT (P.NB_JOURS_ETUDES * C2.PRIX_JOURNEE_ETU)
                FROM PARTICIPATION P
                        JOIN CONVENTION C2 on C2.NUM_CONV = P.NUM_CONV
                WHERE P.NUM_CONV = 3 AND P.NUM_ETU = 1 AND P.DATE_FIN =
'31-07-2022'
            )
            -
            (
                SELECT COALESCE(SUM(A.MONTANT_ACOMPTE), 0)
                FROM ACOMPTE A
                        JOIN CONVENTION C3 on C3.NUM_CONV = A.NUM_CONV
                        JOIN ETUDIANT E on E.NUM_ETU = A.NUM_ETU
                WHERE C3.NUM_CONV = 3 AND E.NUM_ETU = 1 AND A.DATE_ACOMPTE
>= '01-07-2022'
            )
            -
            (
                SELECT (P.NB_JOURS_ETUDES * C2.PRIX_JOURNEE_ETU) *
```

```sql
C2.RETENUES_ENTS
                    FROM PARTICIPATION P
                            JOIN CONVENTION C2 on C2.NUM_CONV = P.NUM_CONV
                    WHERE P.NUM_CONV = 3 AND P.NUM_ETU = 1 AND P.DATE_FIN =
'31-07-2022'
                )
        ));

INSERT INTO INDEMNISATION (NUM_CONV, NUM_ETU, DATE_INDEMNISATION,
MONTANT_INDEMNISATION)
VALUES (3, 4, '01-08-2022',
        (
                (
                SELECT (P.NB_JOURS_ETUDES * C2.PRIX_JOURNEE_ETU)
                FROM PARTICIPATION P
                            JOIN CONVENTION C2 on C2.NUM_CONV = P.NUM_CONV
                WHERE P.NUM_CONV = 3 AND P.NUM_ETU = 4 AND P.DATE_FIN =
'31-07-2022'
                )
                -
                (
                SELECT COALESCE(SUM(A.MONTANT_ACOMPTE), 0)
                FROM ACOMPTE A
                            JOIN CONVENTION C3 on C3.NUM_CONV = A.NUM_CONV
                            JOIN ETUDIANT E on E.NUM_ETU = A.NUM_ETU
                WHERE C3.NUM_CONV = 3 AND E.NUM_ETU = 4 AND A.DATE_ACOMPTE
>= '01-07-2022'
                )
                -
                (
                SELECT (P.NB_JOURS_ETUDES * C2.PRIX_JOURNEE_ETU) *
C2.RETENUES_ENTS
                FROM PARTICIPATION P
                            JOIN CONVENTION C2 on C2.NUM_CONV = P.NUM_CONV
                WHERE P.NUM_CONV = 3 AND P.NUM_ETU = 4 AND P.DATE_FIN =
'31-07-2022'
                )
        ));

INSERT INTO INDEMNISATION (NUM_CONV, NUM_ETU, DATE_INDEMNISATION,
MONTANT_INDEMNISATION)
VALUES (3, 2, '01-08-2022',
        (
                (
                SELECT (P.NB_JOURS_ETUDES * C2.PRIX_JOURNEE_ETU)
                FROM PARTICIPATION P
```

```sql
                        JOIN CONVENTION C2 on C2.NUM_CONV = P.NUM_CONV
                    WHERE P.NUM_CONV = 3 AND P.NUM_ETU = 2 AND P.DATE_FIN =
'31-07-2022'
                )
                -
                (
                    SELECT COALESCE(SUM(A.MONTANT_ACOMPTE), 0)
                    FROM ACOMPTE A
                        JOIN CONVENTION C3 on C3.NUM_CONV = A.NUM_CONV
                        JOIN ETUDIANT E on E.NUM_ETU = A.NUM_ETU
                    WHERE C3.NUM_CONV = 3 AND E.NUM_ETU = 2 AND A.DATE_ACOMPTE
>= '01-07-2022'
                )
                -
                (
                    SELECT (P.NB_JOURS_ETUDES * C2.PRIX_JOURNEE_ETU) *
C2.RETENUES_ENTS
                    FROM PARTICIPATION P
                        JOIN CONVENTION C2 on C2.NUM_CONV = P.NUM_CONV
                    WHERE P.NUM_CONV = 3 AND P.NUM_ETU = 2 AND P.DATE_FIN =
'31-07-2022'
                )
            ));

INSERT INTO INDEMNISATION (NUM_CONV, NUM_ETU, DATE_INDEMNISATION,
MONTANT_INDEMNISATION)
VALUES (3, 5, '01-08-2022',
        (
            (
                SELECT (P.NB_JOURS_ETUDES * C2.PRIX_JOURNEE_ETU)
                FROM PARTICIPATION P
                    JOIN CONVENTION C2 on C2.NUM_CONV = P.NUM_CONV
                WHERE P.NUM_CONV = 3 AND P.NUM_ETU = 5 AND P.DATE_FIN =
'31-07-2022'
            )
            -
            (
                SELECT COALESCE(SUM(A.MONTANT_ACOMPTE), 0)
                FROM ACOMPTE A
                    JOIN CONVENTION C3 on C3.NUM_CONV = A.NUM_CONV
                    JOIN ETUDIANT E on E.NUM_ETU = A.NUM_ETU
                WHERE C3.NUM_CONV = 3 AND E.NUM_ETU = 5 AND A.DATE_ACOMPTE
>= '01-07-2022'
            )
            -
            (
```

```sql
                    SELECT (P.NB_JOURS_ETUDES * C2.PRIX_JOURNEE_ETU) *
C2.RETENUES_ENTS
                    FROM PARTICIPATION P
                            JOIN CONVENTION C2 on C2.NUM_CONV = P.NUM_CONV
                    WHERE P.NUM_CONV = 3 AND P.NUM_ETU = 5 AND P.DATE_FIN =
'31-07-2022'
            )
        ));

UPDATE INDEMNISATION
SET MONTANT_INDEMNISATION = 0
WHERE MONTANT_INDEMNISATION IS NULL;

-- Remplir Remboursement des frais 01/08
INSERT INTO REMBOURSEMENT_FRAIS (DATE_RF, MONTANT_RF)
VALUES ('01-08-2022', (SELECT SUM(F.MONTANT_FRAIS)
                        FROM FRAIS F
                        WHERE F.NUM_ETU = 3
                          AND F.NUM_CONV = 3
                          AND F.NUM_RF IS NULL));

UPDATE FRAIS
SET NUM_RF = (SELECT MAX(NUM_RF) FROM REMBOURSEMENT_FRAIS)
WHERE NUM_ETU = 3
  AND NUM_CONV = 3
  AND NUM_RF IS NULL;

-- Ajout de frais pour le deuxième mois
INSERT INTO FRAIS (NUM_CONV, NUM_ETU, DATE_FRAIS, MONTANT_FRAIS, TYPE_FRAIS)
VALUES (3, 5, '03-08-2022', 300, 'déplacement');

-- Ajout d'accomptes pour le deuxième mois
INSERT INTO ACOMPTE (NUM_CONV, NUM_ETU, MONTANT_ACOMPTE, DATE_ACOMPTE)
VALUES (3, 4, 150, '09-08-2022');

INSERT INTO ACOMPTE (NUM_CONV, NUM_ETU, MONTANT_ACOMPTE, DATE_ACOMPTE)
VALUES (3, 1, 220, '15-08-2022');

-- Remplir les fiches d'indémnités 01/09
INSERT INTO INDEMNISATION (NUM_CONV, NUM_ETU, DATE_INDEMNISATION,
MONTANT_INDEMNISATION)
VALUES (3, 1, '01-09-2022',
        (
            (
                SELECT (P.NB_JOURS_ETUDES * C2.PRIX_JOURNEE_ETU)
                FROM PARTICIPATION P
```

```sql
                            JOIN CONVENTION C2 on C2.NUM_CONV = P.NUM_CONV
                    WHERE P.NUM_CONV = 3 AND P.NUM_ETU = 1 AND P.DATE_FIN =
'31-08-2022'
                )
                -
                (
                    SELECT COALESCE(SUM(A.MONTANT_ACOMPTE), 0)
                    FROM ACOMPTE A
                            JOIN CONVENTION C3 on C3.NUM_CONV = A.NUM_CONV
                            JOIN ETUDIANT E on E.NUM_ETU = A.NUM_ETU
                    WHERE C3.NUM_CONV = 3 AND E.NUM_ETU = 1 AND A.DATE_ACOMPTE
>= '01-08-2022'
                )
                -
                (
                    SELECT (P.NB_JOURS_ETUDES * C2.PRIX_JOURNEE_ETU) *
C2.RETENUES_ENTS
                    FROM PARTICIPATION P
                            JOIN CONVENTION C2 on C2.NUM_CONV = P.NUM_CONV
                    WHERE P.NUM_CONV = 3 AND P.NUM_ETU = 1 AND P.DATE_FIN =
'31-08-2022'
                )
            ));

INSERT INTO INDEMNISATION (NUM_CONV, NUM_ETU, DATE_INDEMNISATION,
MONTANT_INDEMNISATION)
VALUES (3, 4, '01-09-2022',
        (
                (
                    SELECT (P.NB_JOURS_ETUDES * C2.PRIX_JOURNEE_ETU)
                    FROM PARTICIPATION P
                            JOIN CONVENTION C2 on C2.NUM_CONV = P.NUM_CONV
                    WHERE P.NUM_CONV = 3 AND P.NUM_ETU = 4 AND P.DATE_FIN =
'31-08-2022'
                )
                -
                (
                    SELECT COALESCE(SUM(A.MONTANT_ACOMPTE), 0)
                    FROM ACOMPTE A
                            JOIN CONVENTION C3 on C3.NUM_CONV = A.NUM_CONV
                            JOIN ETUDIANT E on E.NUM_ETU = A.NUM_ETU
                    WHERE C3.NUM_CONV = 3 AND E.NUM_ETU = 4 AND A.DATE_ACOMPTE
>= '01-08-2022'
                )
                -
                (
```

```sql
                            SELECT (P.NB_JOURS_ETUDES * C2.PRIX_JOURNEE_ETU) *
C2.RETENUES_ENTS
                            FROM PARTICIPATION P
                                    JOIN CONVENTION C2 on C2.NUM_CONV = P.NUM_CONV
                            WHERE P.NUM_CONV = 3 AND P.NUM_ETU = 4 AND P.DATE_FIN =
'31-08-2022'
                    )
            ));

INSERT INTO INDEMNISATION (NUM_CONV, NUM_ETU, DATE_INDEMNISATION,
MONTANT_INDEMNISATION)
VALUES (3, 2, '01-09-2022',
        (
                (
                    SELECT (P.NB_JOURS_ETUDES * C2.PRIX_JOURNEE_ETU)
                    FROM PARTICIPATION P
                            JOIN CONVENTION C2 on C2.NUM_CONV = P.NUM_CONV
                    WHERE P.NUM_CONV = 3 AND P.NUM_ETU = 2 AND P.DATE_FIN =
'31-08-2022'
                )
                -
                (
                    SELECT COALESCE(SUM(A.MONTANT_ACOMPTE), 0)
                    FROM ACOMPTE A
                            JOIN CONVENTION C3 on C3.NUM_CONV = A.NUM_CONV
                            JOIN ETUDIANT E on E.NUM_ETU = A.NUM_ETU
                    WHERE C3.NUM_CONV = 3 AND E.NUM_ETU = 2 AND A.DATE_ACOMPTE
>= '01-08-2022'
                )
                -
                (
                    SELECT (P.NB_JOURS_ETUDES * C2.PRIX_JOURNEE_ETU) *
C2.RETENUES_ENTS
                    FROM PARTICIPATION P
                            JOIN CONVENTION C2 on C2.NUM_CONV = P.NUM_CONV
                    WHERE P.NUM_CONV = 3 AND P.NUM_ETU = 2 AND P.DATE_FIN =
'31-08-2022'
                )
            ));

INSERT INTO INDEMNISATION (NUM_CONV, NUM_ETU, DATE_INDEMNISATION,
MONTANT_INDEMNISATION)
VALUES (3, 5, '01-09-2022',
        (
                (
                    SELECT (P.NB_JOURS_ETUDES * C2.PRIX_JOURNEE_ETU)
```

```sql
                    FROM PARTICIPATION P
                            JOIN CONVENTION C2 on C2.NUM_CONV = P.NUM_CONV
                    WHERE P.NUM_CONV = 3 AND P.NUM_ETU = 5 AND P.DATE_FIN =
'31-08-2022'
                )
                -
                (
                    SELECT COALESCE(SUM(A.MONTANT_ACOMPTE), 0)
                    FROM ACOMPTE A
                            JOIN CONVENTION C3 on C3.NUM_CONV = A.NUM_CONV
                            JOIN ETUDIANT E on E.NUM_ETU = A.NUM_ETU
                    WHERE C3.NUM_CONV = 3 AND E.NUM_ETU = 5 AND A.DATE_ACOMPTE
>= '01-08-2022'
                )
                -
                (
                    SELECT (P.NB_JOURS_ETUDES * C2.PRIX_JOURNEE_ETU) *
C2.RETENUES_ENTS
                    FROM PARTICIPATION P
                            JOIN CONVENTION C2 on C2.NUM_CONV = P.NUM_CONV
                    WHERE P.NUM_CONV = 3 AND P.NUM_ETU = 5 AND P.DATE_FIN =
'31-08-2022'
                )
            ));

UPDATE INDEMNISATION
SET MONTANT_INDEMNISATION = 0
WHERE MONTANT_INDEMNISATION IS NULL;

-- Remplir Remboursement des frais 01/09
INSERT INTO REMBOURSEMENT_FRAIS (DATE_RF, MONTANT_RF)
VALUES ('01-09-2022', (SELECT SUM(F.MONTANT_FRAIS)
                       FROM FRAIS F
                       WHERE F.NUM_ETU = 5
                         AND F.NUM_CONV = 3
                         AND F.NUM_RF IS NULL));

UPDATE FRAIS
SET NUM_RF = (SELECT MAX(NUM_RF) FROM REMBOURSEMENT_FRAIS)
WHERE NUM_ETU = 5
  AND NUM_CONV = 3
  AND NUM_RF IS NULL;

-- Remplir la facture
UPDATE CONVENTION
SET DATE_FACTURE = '01-09-2022'
```

```sql
WHERE NUM_CONV = 3;


UPDATE CONVENTION
SET MONTANT_FACTURE = (
    SELECT SUM(A.NB_JOURS_ETUDES_TOTAL * C2.PRIX_JOURNEE_ETU)
    FROM EST_ASSIGNE A
             JOIN CONVENTION C2 on A.NUM_CONV = C2.NUM_CONV
    WHERE A.NUM_CONV = 3
    )
WHERE NUM_CONV = 3;

commit;
```

Requêtes SQL

Voici quelques requêtes pour :

Lister les projets terminés

```sql
SELECT NOM_ETUDE
FROM CONVENTION
WHERE DATE_FACTURE IS NOT NULL;
```

Lister les projets en cours

```sql
SELECT NOM_ETUDE
FROM CONVENTION
WHERE DATE_FACTURE IS NULL;
```

Lister les projets dans lesquels l'étudiante Rodriguez Marie a participé

```sql
SELECT NOM_ETUDE
FROM EST_ASSIGNE A
    JOIN CONVENTION C2 on C2.NUM_CONV = A.NUM_CONV
    JOIN ETUDIANT E on A.NUM_ETU = E.NUM_ETU
WHERE UPPER(NOM_ETU) = 'RODRIGUEZ' AND UPPER(PRENOM_ETU) = 'MARIE';
```

Lister le nombre de projet par étudiant

```sql
SELECT NUM_ETU, COUNT(NUM_CONV)
FROM EST_ASSIGNE
GROUP BY NUM_ETU
```