

3ème étape SAE BD

Sarah EL MCHANTEF & Maylee PEIRO TOMAS

Table des matières

Introduction	2
Tests Réalisés :	3
Insertion dans Conference	3
Mise à jour de Conference :	5
Test du type de données pour la table Conference:	6
Insertion dans Participant :	6
Mise à jour de Participant :	8
Insertion dans Inscription :	9
Insertion pour SessionConf :	12
Mise à jour pour SessionConf :	13
Insertion pour ArticleAccept :	13
Les suppressions :	14
Conclusion :	15

Introduction

L'objectif de ce rapport est de vérifier les contraintes de clé, les contraintes référentielles, ainsi que les types des attributs en faisant des mises à jour (requêtes INSERT, UPDATE et DELETE). Les tests incluent des cas valides et invalides, illustrés par des exemples de requêtes SQL et des copies d'écran pour les erreurs rencontrées.

Les relations suivantes ont été créées :

CREATE TABLE Conference (

IdConf INT Primary Key,

Acronyme varchar2(100),

DateC date,

Lieu varchar2(100),

montantFrais INT);

CREATE TABLE Participant (

idPersonne INT Primary Key,

typeP varchar2(50));

CREATE TABLE Inscription (

idPersonne INT REFERENCES Participant,

idConf INT REFERENCES Conference,

dateInscription Date,

typePaiement varchar2(50),

PRIMARY KEY (idPersonne,idConf));

CREATE TABLE SessionConf (

idConf INT REFERENCES conference,

nomSession varchar2(50),

numSalle INT,

idPersonne INT,
idConf2 INT,
Primary Key (idConf,nomSession),
Foreign Key (idPersonne,idConf2) REFERENCES Inscription);

CREATE TABLE ArticleAccept (
idArticle INT Primary Key,
categorie varchar2(50),
nbPages INT,
texteReviser varchar2(50),
idConf INT,
nomSession varchar2(50),
Foreign Key (idConf,nomSession) REFERENCES SessionConf);

Tests Réalisés :

Insertion dans Conference

1) Insertion valide dans Conférence :

```
INSERT INTO Conference (IdConf, Acronyme, DateC, Lieu, montantFrais)VALUES  
(1, 'CLIMAT', TO_DATE('2023-07-20', 'YYYY-MM-DD'), 'Paris', 500);  
  
commit ;
```

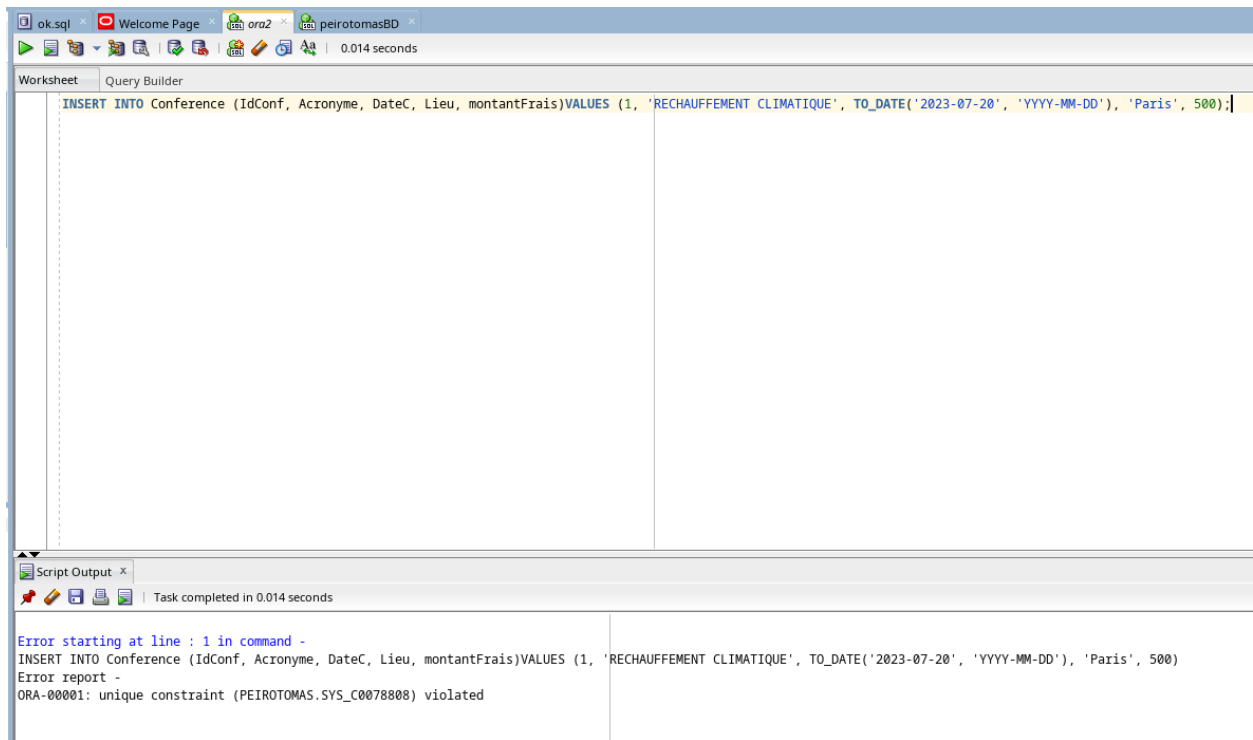
Cette insertion est réussie, car IdConf = 1 n'existe pas encore dans la table.

2) Insertion invalide dans Conference : (car duplication de clé primaire) :

```
INSERT INTO Conference (IdConf, Acronyme, DateC, Lieu, montantFrais)VALUES  
(1, 'RECHAUFFEMENT CLIMATIQUE', TO_DATE('2023-07-20', 'YYYY-MM-DD'),  
'Paris', 500) ;  
  
commit ;
```

Cette insertion est invalide, il y a un message d'échec affiché :

ORA-00001: unique constraint violated



Cette erreur survient parce que la valeur de IdConf =1 existe déjà dans la table, il y a ainsi une violation de la contrainte de clé primaire qui exige des valeurs uniques sur l'attribut IdConf.

3) Insertion d'un autre exemple :

INSERT INTO Conference (IdConf, Acronyme, DateC, Lieu, montantFrais)VALUES (2, 'Intelligence artificielle', TO_DATE('2023-07-20', 'YYYY-MM-DD'), 'Paris', 500);

commit;

Cette insertion ne pose pas de problème car la clé primaire est différente de la première insertion.

Mise à jour de Conference :

1) Mise à jour qui fonctionne :

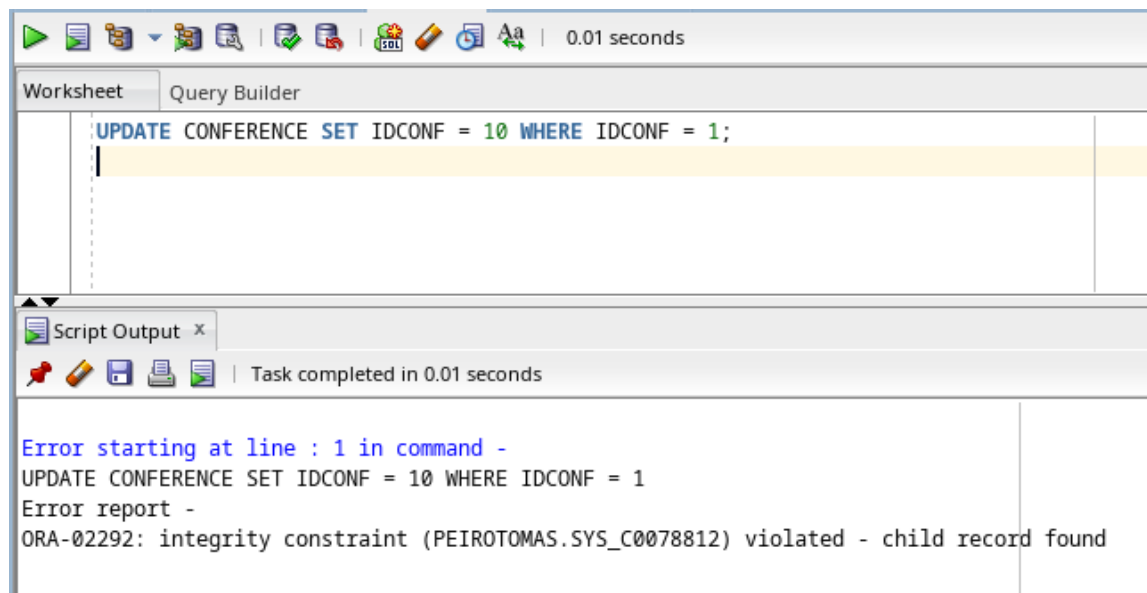
```
update Conference set ACRONYME='RECHAUFFEMENT CLIMATIQUE' where  
IDconf=1;  
  
commit;
```

La mise à jour est réussie. Nous avons renommé l'acronyme de la conférence (qui ne fait pas partie de la clés primaire) 'réchauffement climatique' au lieu de 'climat'. Ainsi, la contrainte de clé primaire n'est pas affectée, IdConf a été utilisé comme un critère : IdConf=1.

2) Mise à jour qui ne fonctionne pas :

```
update Conference set idconf=10 where idconf=1
```

ORA-02292: integrity constraint violated - child record found



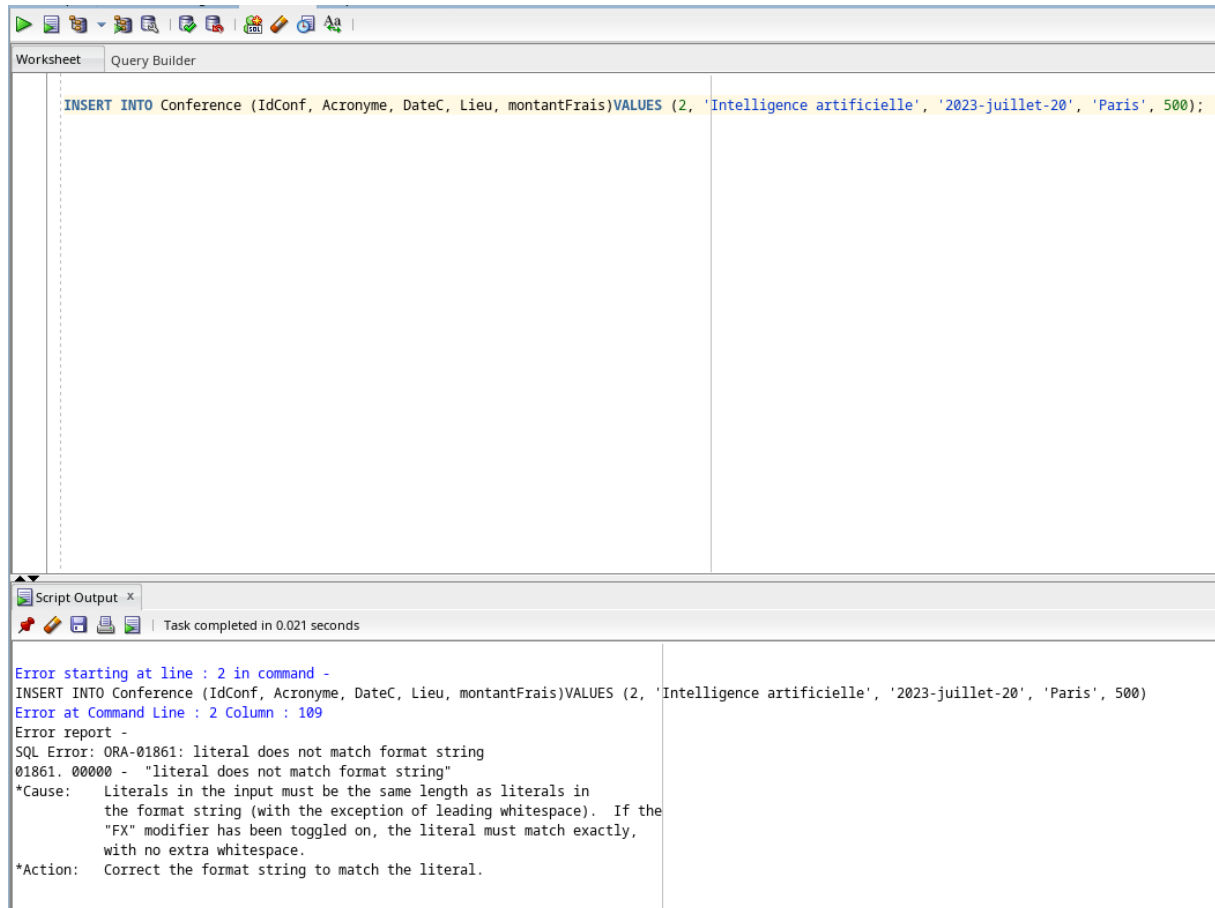
Cette mise à jour ne peut pas se réaliser parce qu'il y a une violation de contrainte d'intégrité.

Test du type de données pour la table Conference:

Nous allons essayer d'insérer une valeur de type : VARCHAR au lieu d'un type DATE :

Exemple :

```
INSERT INTO Conference (IdConf, Acronyme, DateC, Lieu, montantFrais)VALUES  
(2, 'Intelligence artificielle', '2023-juillet-20', 'Paris', 500);
```



Ça ne fonctionne pas, puisque que pour insérer un format date, on doit utiliser la fonction TO_DATE et donner le format souhaité. Dans notre cas 'YYYY-mon-DD'.

Insertion dans Participant :

1) Insertion qui fonctionne :

```
INSERT INTO Participant (idPersonne, typeP)VALUES (102, 'SARAH ELMCHANTEF');
```

```
INSERT INTO Participant (idPersonne, typeP)VALUES (103, 'MAYLEE PEIRO-TOMAS');
```

```
INSERT INTO Participant (idPersonne, typeP)VALUES (104, 'AMINE AYSSINE');
```

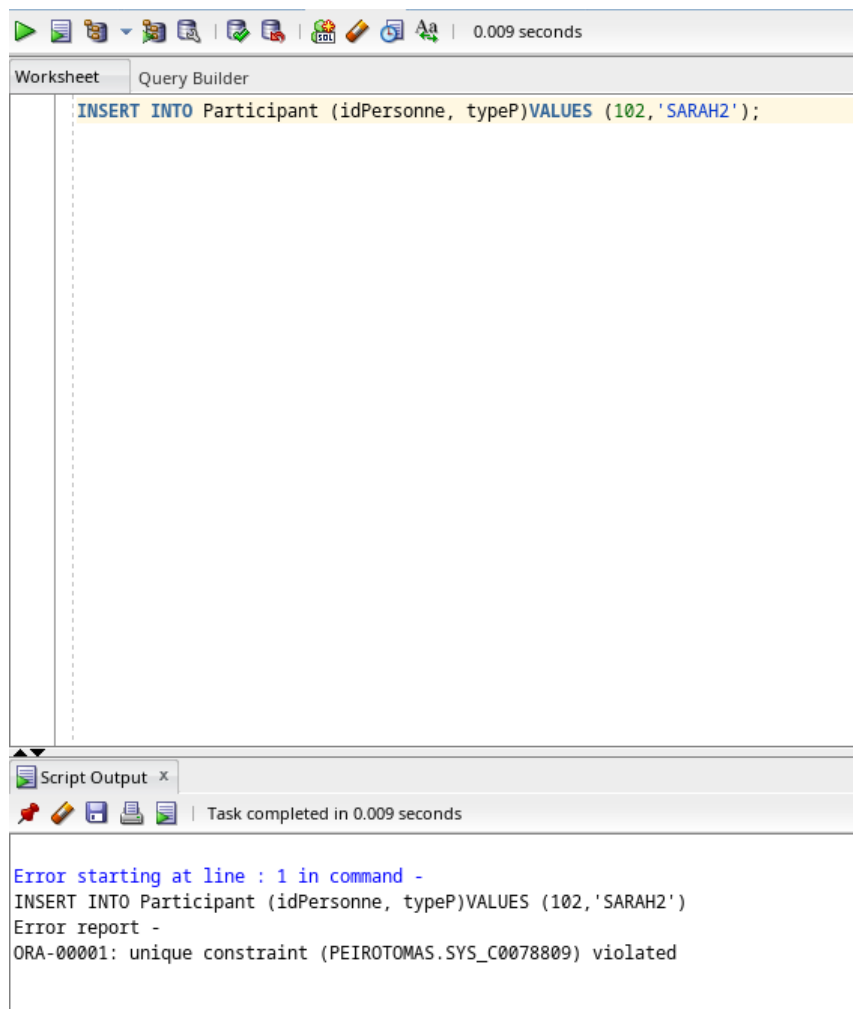
```
commit;
```

Nous avons inséré avec succès les 3 nouveaux enregistrements dans la table participant.

2) Insertion qui ne fonctionne pas :

```
INSERT INTO Participant (idPersonne, typeP)VALUES (102, 'SARAH2');
```

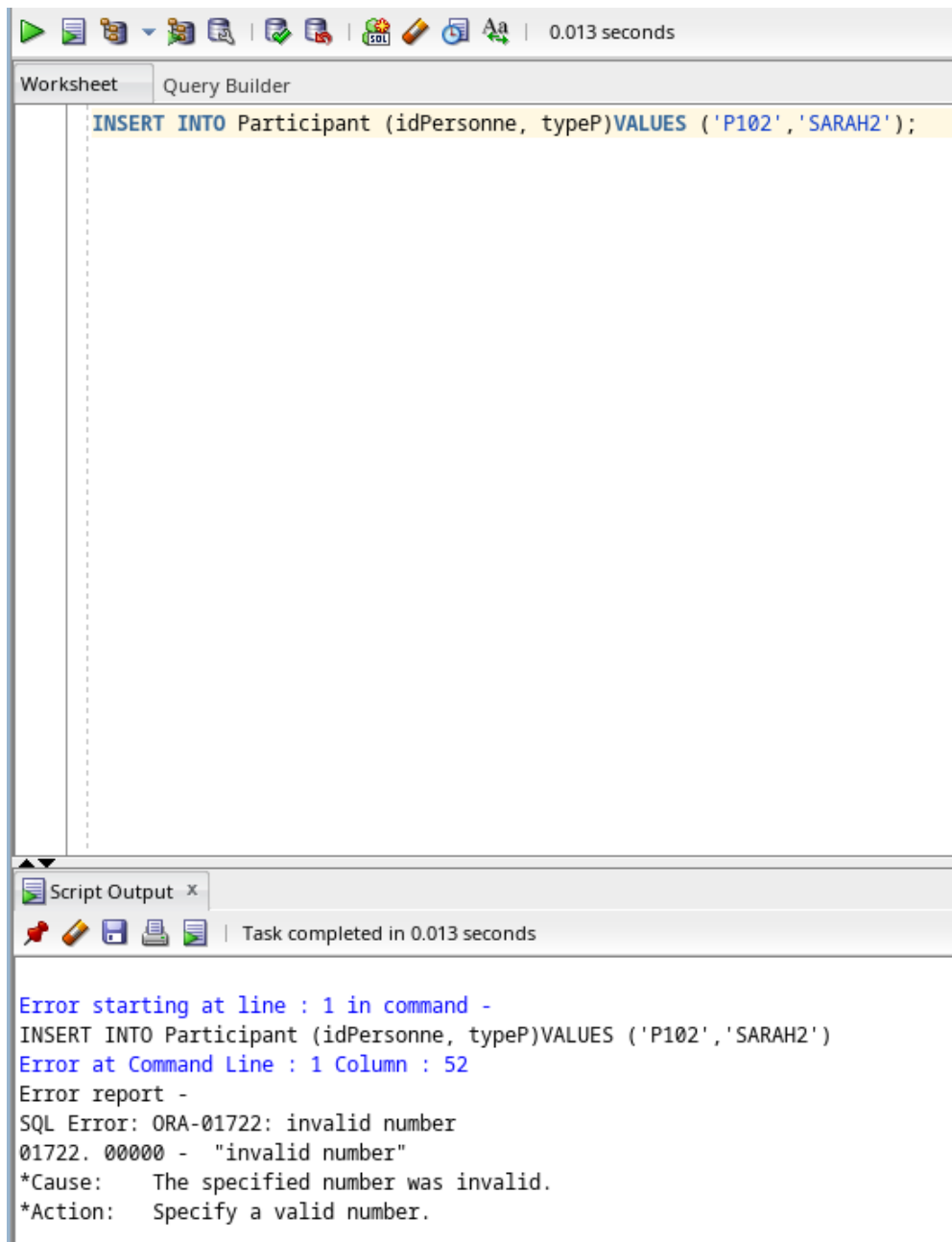
ORA-00001: unique constraint violated



Cette erreur survient parce que la valeur de `IdPersonne` = 102 existe déjà dans la table, il y a ainsi une violation ainsi la contrainte de clé primaire qui exige des valeurs uniques sur l'attribut `IdPersonne`.

3) Autre exemple :

INSERT INTO Participant (idPersonne, typeP)VALUES ('P102', 'SARAH2');



Cette insertion ne fonctionnera pas non plus, car l'IdPersonne doit être un INT et non pas un VARCHAR.

Mise à jour de Participant :

1) UPDATE qui fonctionne :

update Participant set typeP='REGINE LALEAU' where idPersonne=104 ;

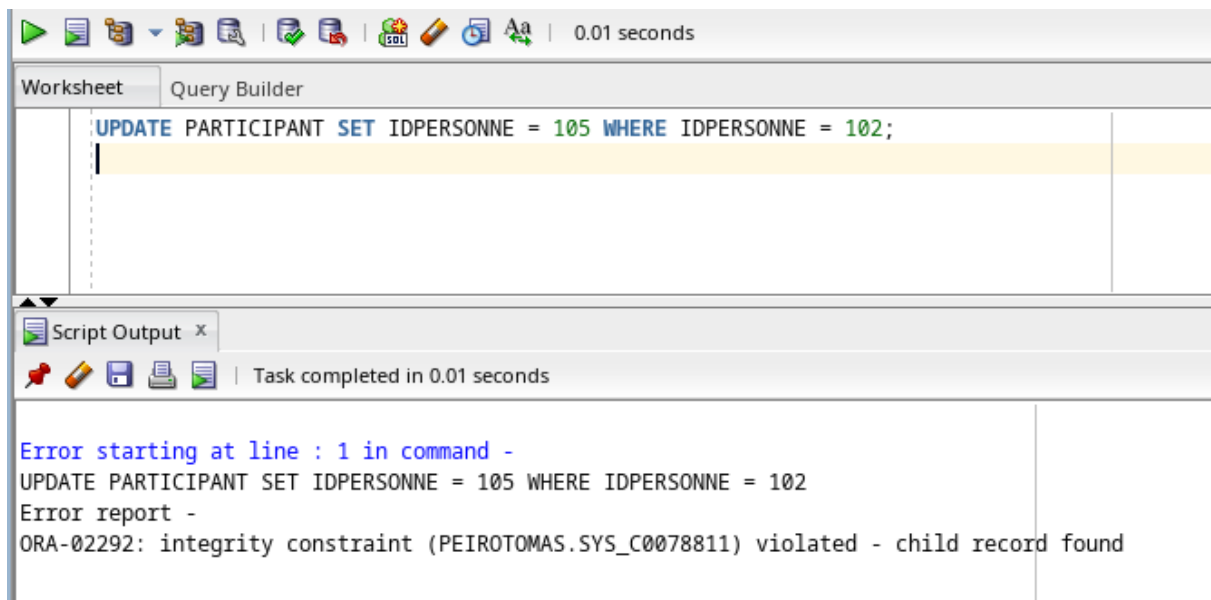
commit ;

Ici nous avons modifié le nom de la personne numéro 104, cette update fonctionne en utilisant la primary key comme clause where.

2) UPDATE qui ne fonctionne pas :

update Participant set idPeronne=105 where idPersonne=102;

ORA-02292: integrity constraint violated - child record found



Cette update génère une erreur parce que nous essayons de modifier la valeur de la primary key.

Insertion dans Inscription :

1) Insertions qui fonctionnent :

INSERT INTO Inscription (idPersonne, idConf, dateInscription, typePaiement) VALUES (102, 1, TO_DATE('2025-01-23', 'YYYY-MM-DD'), 'PAYPAL');

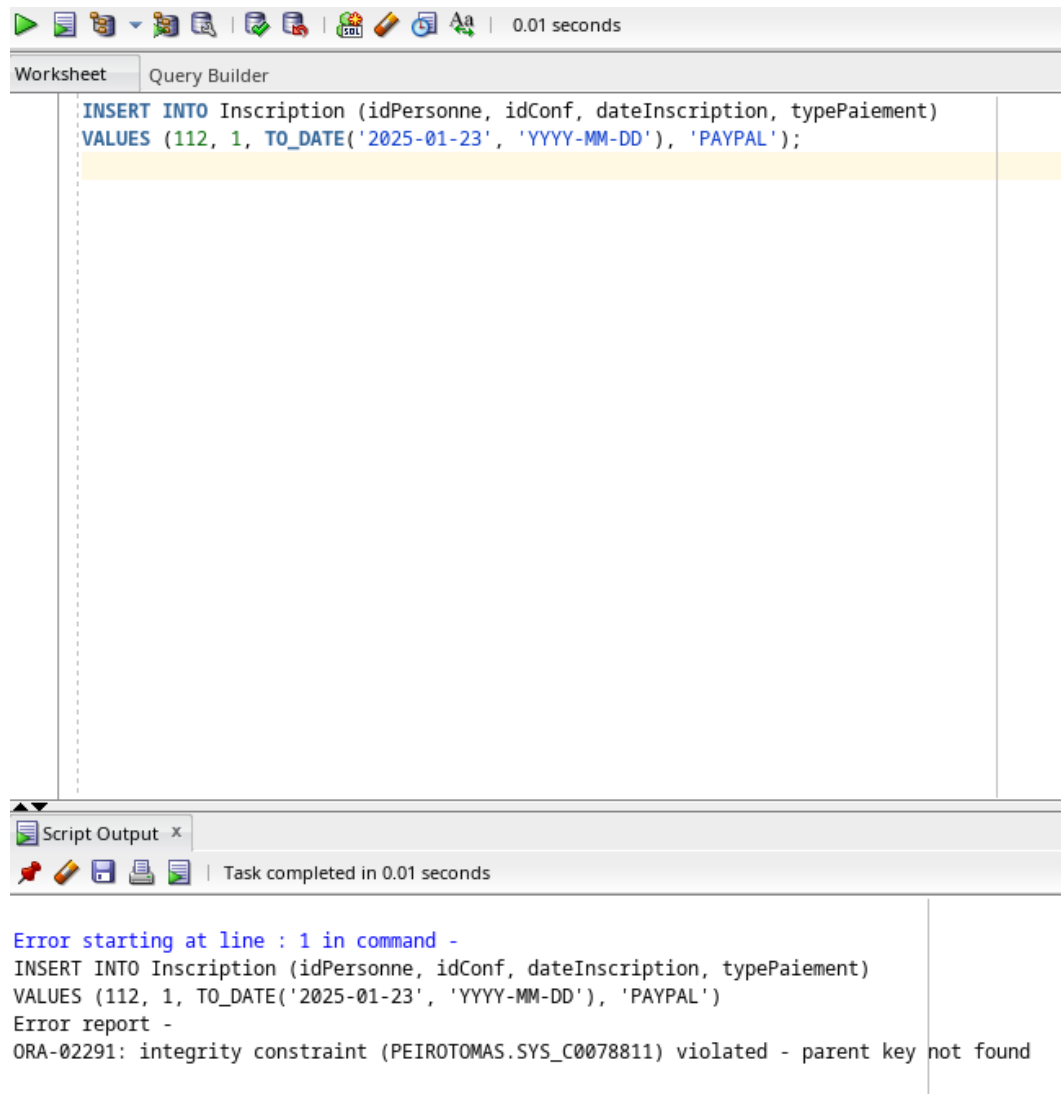
INSERT INTO Inscription (idPersonne, idConf, dateInscription, typePaiement) VALUES (104, 1, TO_DATE('2025-01-23', 'YYYY-MM-DD'), 'CARTE BANCAIRE');

```
INSERT INTO Inscription (idPersonne, idConf, dateInscription,  
typePaiement)VALUES (103, 1, TO_DATE('2025-01-23', 'YYYY-MM-DD'), 'PAYPAL');  
  
commit;
```

Nous avons inséré avec succès les 3 enregistrements dans la table inscription.

2) Insertion qui ne fonctionne pas :

```
INSERT INTO Inscription (idPersonne, idConf, dateInscription,  
typePaiement)VALUES (112, 1, TO_DATE('2025-01-23', 'YYYY-MM-DD'), 'PAYPAL');
```

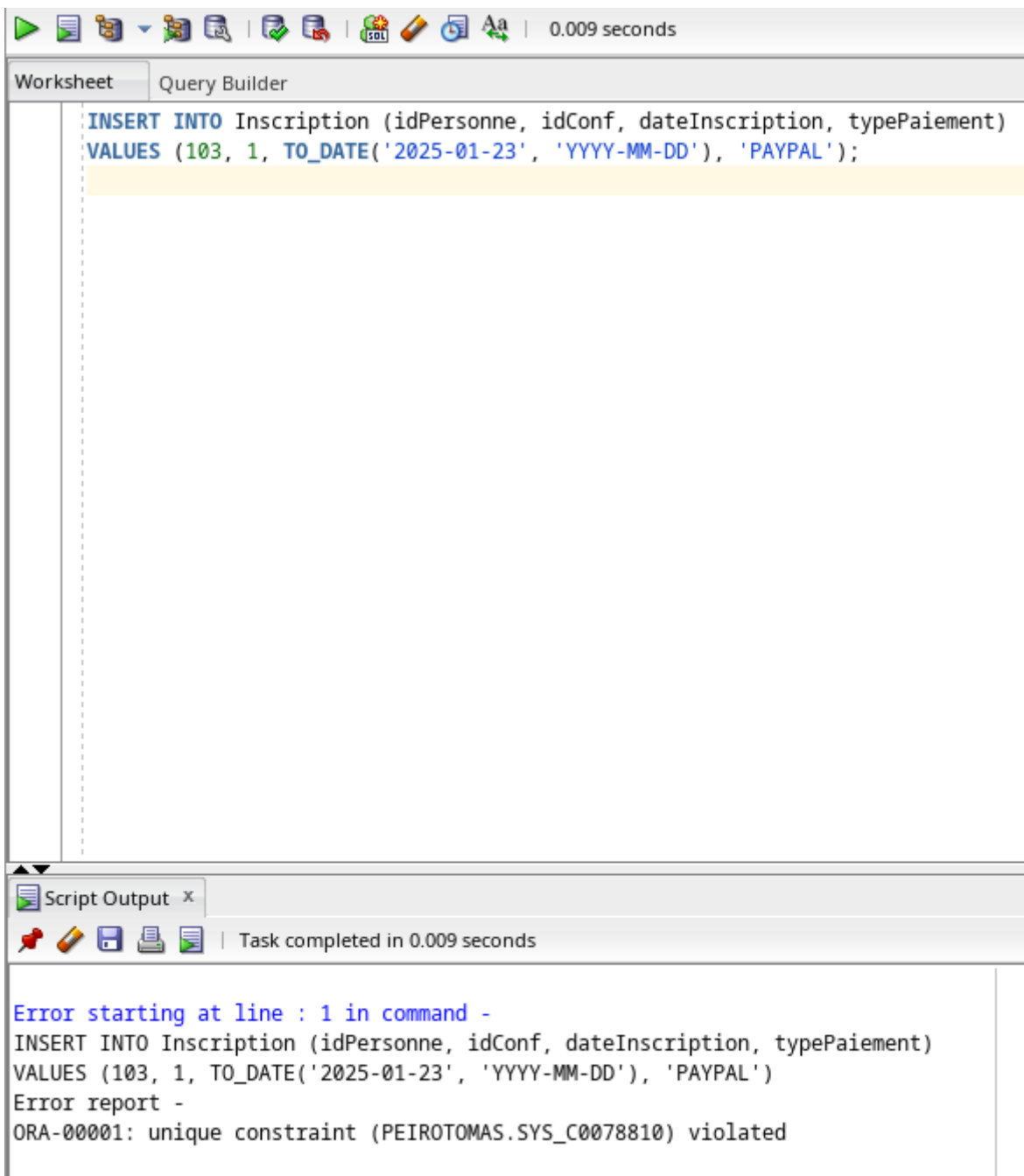


Cette insertion ne se réalise pas, car l'idPersonne=112 n'existe pas dans la table Participant (foreign key sur la table participant).

3) Autre exemple qui ne fonctionne pas :

```
INSERT INTO Inscription (idPersonne, idConf, dateInscription,  
typePaiement)VALUES (103, 1, TO_DATE('2025-01-23', 'YYYY-MM-DD'), 'PAYPAL');
```

ORA-00001: unique constraint violated



Cette erreur survient parce que la combinaison (idPersonne, idConf) existe déjà, ce qui viole la clé primaire.

Insertion pour SessionConf :

1) Les insertions qui fonctionnent :

```
INSERT INTO SessionConf (idConf, nomSession, numSalle, idPersonne, idConf2)VALUES (1, 'Presentation_AI', 1, 102, 1);
```

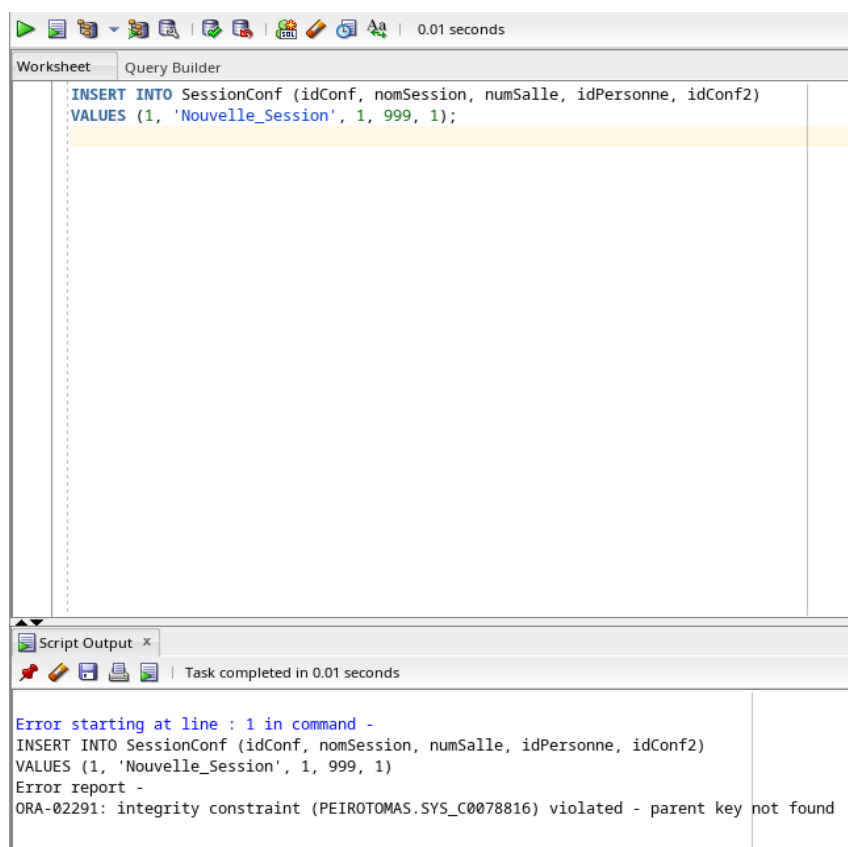
```
INSERT INTO SessionConf (idConf, nomSession, numSalle, idPersonne, idConf2)VALUES (2, 'Presentation_AI', 1, 104, 1);
```

commit;

Les deux insertions se réalisent avec succès. Les relations avec Inscription et Conférence sont respectées.

2) Insertion qui ne fonctionne pas :

```
INSERT INTO SessionConf (idConf, nomSession, numSalle, idPersonne, idConf2)VALUES (1, 'Nouvelle_Session', 1, 999, 1);
```



Cette insertion ne se réalise pas car : La valeur idPersonne = 999 n'existe pas dans Inscription, ce qui représente une violation de contrainte de la clé étrangère.

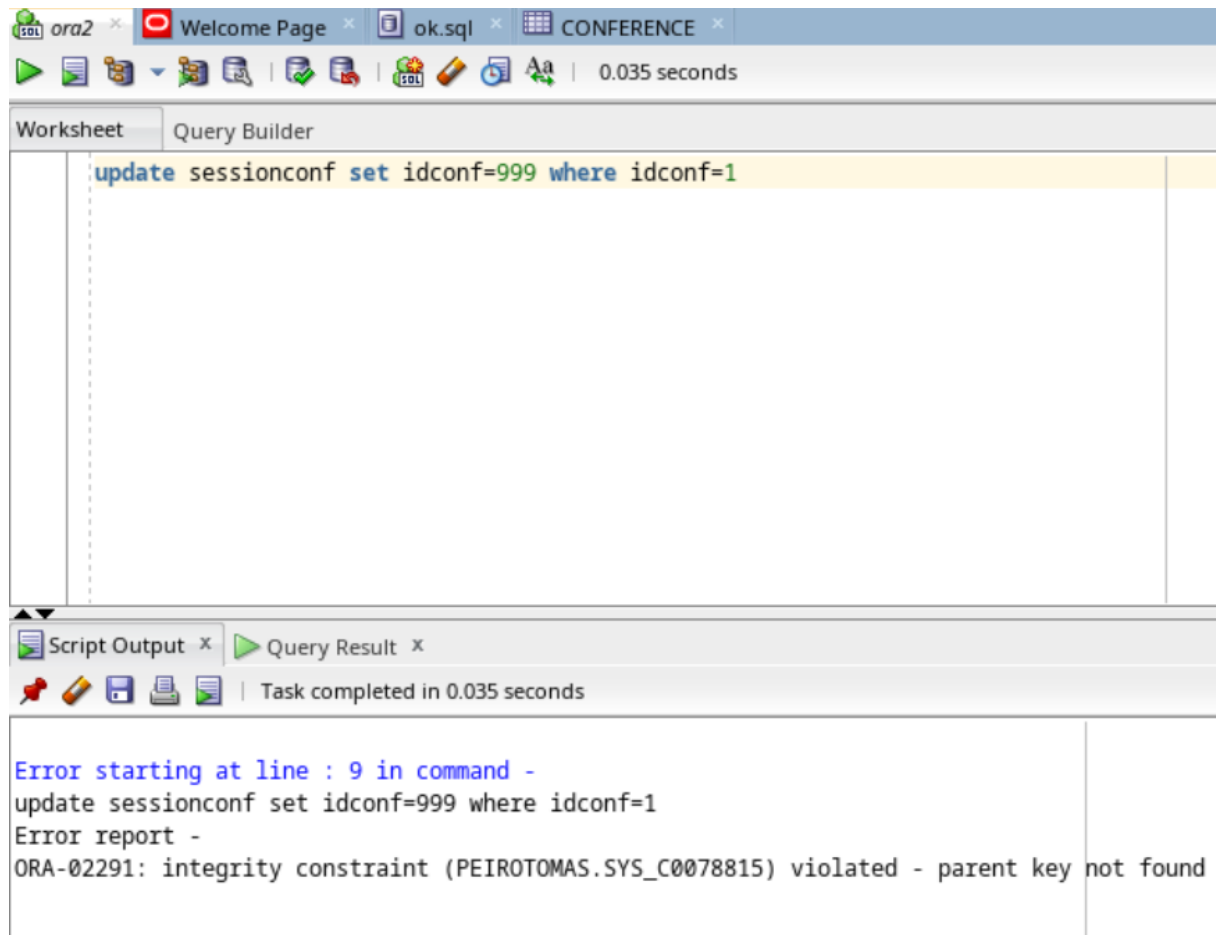
Mise à jour pour SessionConf :

1) Mise à jour invalide :

UPDATE SessionConf SET idConf2 = 999 WHERE nomSession = 'Presentation_AI';

Cette mise à jour ne fonctionne pas car, elle crée une relation vers une clé étrangère inexistante.

idConf=999 n'existe pas dans la table Conference.



Insertion pour ArticleAccept :

1) Insertions qui se réalisent correctement :

INSERT INTO ArticleAccept (idArticle, categorie, nbPages, texteRevisé, idConf, nomSession)VALUES (201, 'CLIMATIQUE', 10, 'ARTICLE VALIDE ', 1, 'Presentation_AI');

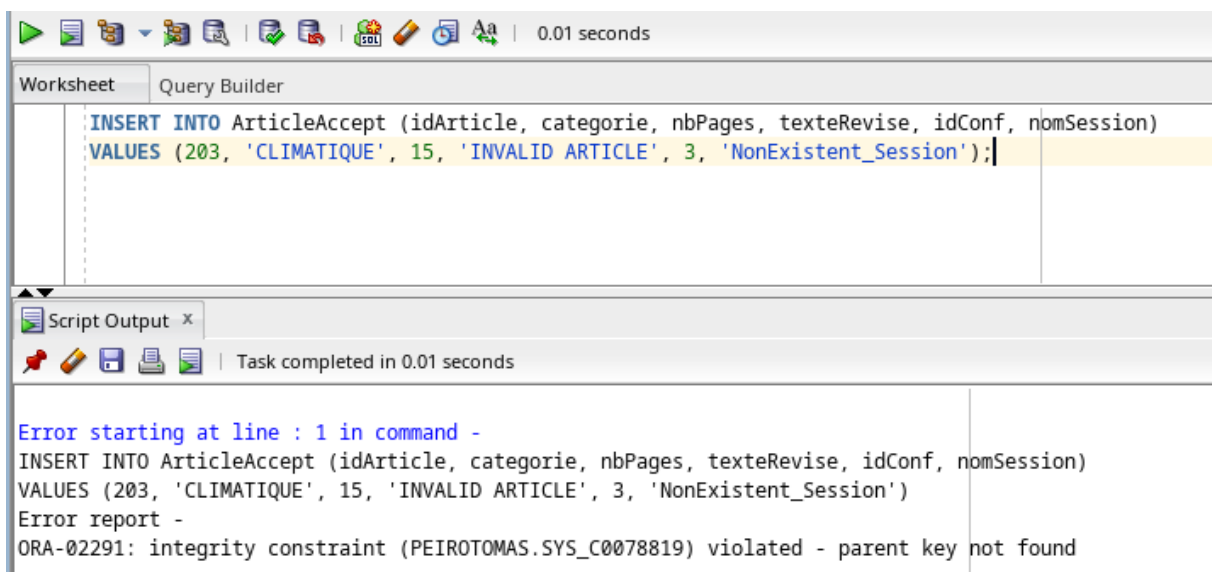
```
INSERT INTO ArticleAccept (idArticle, categorie, nbPages, texteRevisé, idConf, nomSession)VALUES (202, 'INFORMATIQUE', 120, 'ARTICLE AI VALIDE', 1, 'Presentation_AI');
```

```
commit;
```

Les deux insertions aboutissent sans erreurs. Les relations avec SessionConf sont respectées.

2) Insertion qui génère une erreur :

```
INSERT INTO ArticleAccept (idArticle, categorie, nbPages, texteRevisé, idConf, nomSession) VALUES (203, 'CLIMATIQUE', 15, 'INVALID ARTICLE', 3, 'NonExistent_Session');
```



Cette insertion ne se réalise pas car : Les valeurs idConf = 3 et nomSession = 'NonExistent_Session' n'existent pas dans SessionConf.

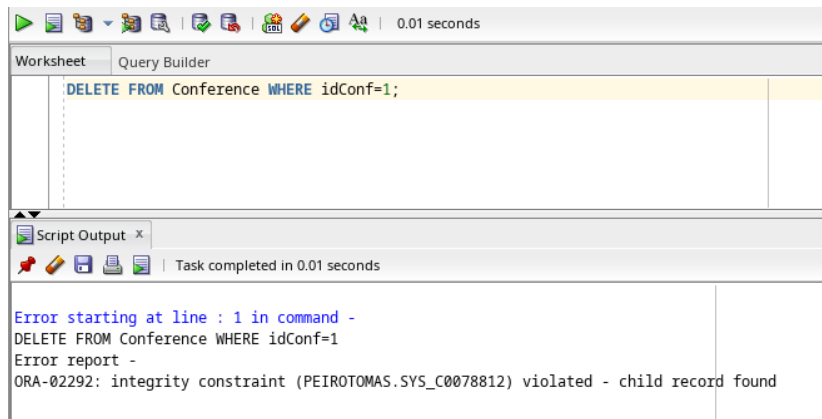
Les suppressions :

1) Pour la table Conference :

Nous allons essayer de supprimer une ligne de la table conference, avec un id qui est utilisé dans les tables SessionConf, Inscription et ArticleAccept.

```
DELETE FROM Conference WHERE idconf=1 ;
```

ORA-02292: integrity constraint violated - child record found



La suppression de la conférence `IdConf = 1` échoue, car des enregistrements dans les tables dépendantes (par exemple `Inscription`) font référence à cette conférence. La contrainte référentielle empêche la suppression pour préserver l'intégrité des données.

2) Pour la table `ArticleAccept` :

`DELETE FROM ArticleAccept WHERE idArticle = 201 ;`

`commit ;`

Cette suppression est réussie car aucun enregistrement dans les autres tables ne dépend de `idArticle = 201`. Aucune table ne dépend de la primary key de `ArticleAccept`, ainsi toutes les suppressions sont permises.

Conclusion :

Ce rapport nous démontre que les contraintes définies dans la base de données jouent un rôle crucial pour assurer l'intégrité et la cohérence des données. Les tests ont mis en évidence les comportements attendus pour les opérations d'insertion, de mise à jour et de suppression, en validant notamment :

- Les contraintes de clé primaire, qui empêchent les doublons et assurent l'unicité des enregistrements.
- Les contraintes de clé étrangère, qui garantissent la cohérence des relations entre les tables et empêchent des modifications incohérentes.
- Les contraintes sur les types d'attributs, qui force le respect des types de données.