Nahomy Durand & Mariam Traoré
20201414 - 20114254

Devoir#2
IFT2935

RAPPORT DE PROJET

Travail présenté au
Prof. Michel Boyer

Base de données
Université de Montréal
5 mars 2024

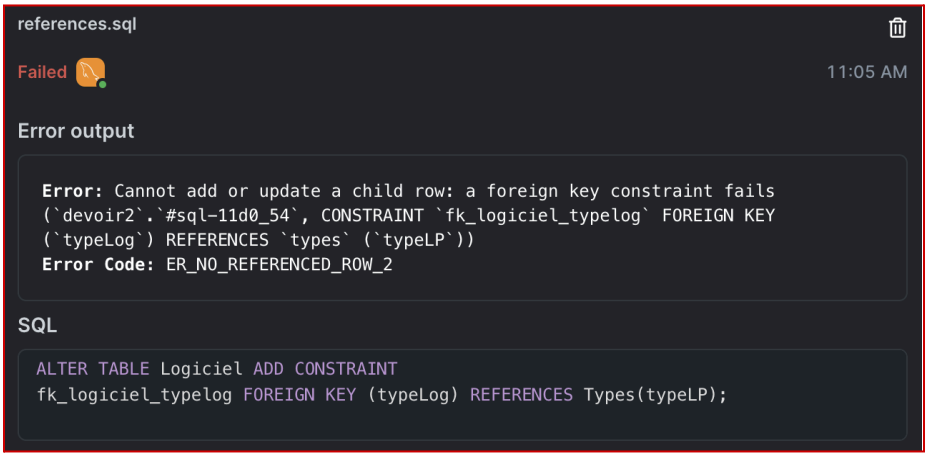
INTRODUCTION

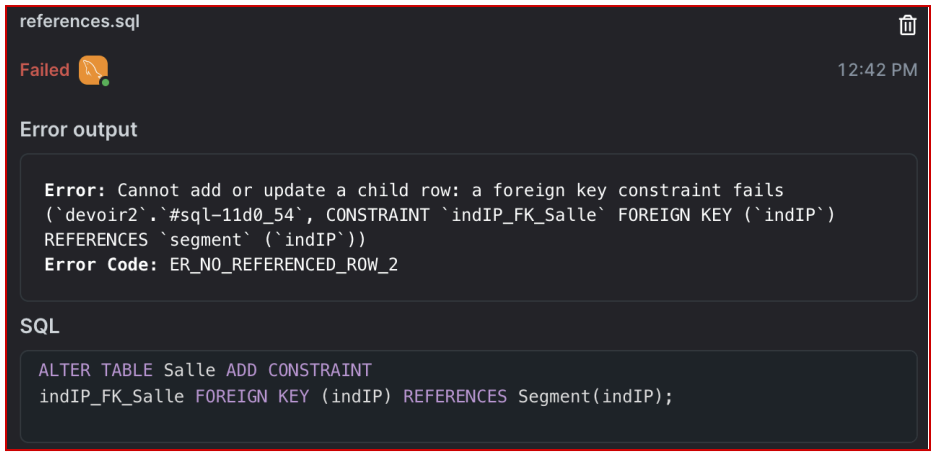
Dans le cadre du deuxième travail pratique du cours *Base de donnée - IFT2935*, nous avons créer un projet SQL répondant aux besoins spécifiques d'une entreprise souhaitant gérer efficacement son parc informatique. Ce rapport résumera les problèmes encontrés lors de l'implémentation de ce projet.

RAPPORT

Tâche #5 : **Suite à l'éxécution du fichier *references.sql*, expliquer l'erreur, et afficher les lignes qui ont posé le problème.**

Voici les erreurs que nous avons reçu lorsque nous avons exécuter *references.sql* :





Cette erreur fait référence à certaines instances qui existent déjà dans les tableaux concernées et qui signifie que l'intégrité référentielle n'est pas respecté.

Dans le cas du tableau SALLE, nous pouvons voir que les tuples 7 et 8 ont le indIP “130.120.83”.

	nSalle	nomSalle	nbPoste	indIP
1	s01	Salle 1	3	130.120.80
2	s02	Salle 2	2	130.120.80
3	s03	Salle 3	2	130.120.80
4	s11	Salle 11	2	130.120.81
5	s12	Salle 12	1	130.120.81
6	s21	Salle 21	2	130.120.82
7	s22	Salle 22	0	130.120.83
8	s23	Salle 23	0	130.120.83

Cependant, puisque #indIP de SALLE fait référence au indIP de SEGMENT, nous remarquons qu’il n’existe pas de tuple segment avec l’indIP “130.120.83”.

	indIP	nomSegment	etage
1	130.120.80	Brin RDC	null
2	130.120.81	Brin 1er etage	null
3	130.120.82	Brin 2eme etage	null

Cela veut dire que le tuple 7 et 8 du tableau SALLE essaye de référencer une clé qui n’existe pas dans le tableau SEGMENT. Nous avons donc enlever ces deux tuples pour procéder à la suite du projet.

De plus, pour le tableau LOGICIEL, voici les données que l’on possède :

	nLog	nomLog	dateAch	version	typeLog	prix
1	log1	Oracle 6	1995-05-13	6.2	UNIX	3000
2	log2	Oracle 8	1999-09-15	8i	UNIX	5600
3	log3	SQL Server	1998-04-12	7	PCNT	3000
4	log4	Front Page	1997-06-03	5	PCWS	500
5	log5	WinDev	1997-05-12	5	PCWS	750
6	log6	SQL*Net	null	2.0	UNIX	500
7	log7	I. I. S.	2002-04-12	2	PCNT	900
8	log8	DreamWeaver	2003-09-21	2.0	BeOS	1400

L’erreur provient du tuple #8 car son typeLog = “BeOS”. Cependant, le tableau TYPES ne possède pas de typeLP “BeOS”. Cela crée une erreur puisque typeLog ne peut pas se référencer à une clé du tableau TYPES qui n’existe pas. Le tuple “log8” a donc été enlevé.

	typeLP	nomType
1	NC	Network Computer
2	PCNT	PC Windows NT
3	PCWS	PC Windows
4	TX	Terminal X-Window
5	UNIX	Systeme Unix

Tâche #7 : Si on a ajouté les contraintes référentielles dans le fichier *ParcInfo.sql*, c’est quoi l’ordre de la création et de la suppression des relations. Expliquer.

Si nous ajoutons les contraintes référentielles directement dans notre fichier *ParcInfo.sql* lorsque nous faisons la création de nos tableaux, il faudra respecter l’ordre de création selon les tableaux possédant des clés étrangères et ceux qui n’en possèdent pas.
Il faudra alors créer, en ordre :

1. Le tableau TYPES puisqu’elle ne contient pas de clé étrangère.
2. Le tableau SEGMENT pour la même raison.
3. Les tableaux SALLE et LOGICIEL qui référencent TYPES et SEGMENT.
4. Le tableau POSTE qui dépend de SALLE.
5. Le tableau INSTALLER qui référence POSTE et LOGICIEL.

S’il faudrait supprimer ces relations, il faudrait le faire en ordre inverse ; commencer par les tableaux qui possèdent les clés étrangères pointant vers d’autres tableaux et finir par ceux qui n’en possède aucune. Cela évite des erreurs de contraites référentielles dans le cas où quelqu’un essayerai de supprimer un tableau étant toujours référencée par une clé étrangère d'un autre tableau.