

---

## Travail pratique #2

### Conception physique et contraintes

---

## 1 Introduction

Au cours de la session, les travaux auront comme sujet le développement d'une application de gestion de données, à l'image des projets qui se font en entreprise. Chacun des 4 travaux du cours portera sur un ou plusieurs concepts vus en classe, et sera effectué en équipe de deux (2) étudiant(e)s maximum.

Dans ce second TP, votre travail consistera, dans un premier temps, à identifier les règles d'affaires (contraintes) associées au système présenté dans le document de mise en situation fourni avec le travail pratique #1. Ensuite, vous devrez créer un script SQL permettant d'implémenter vos tables ainsi que les contraintes sur la base de données Oracle. De plus, vous vous familiariserez avec les procédures stockées et l'environnement SQLPlus d'Oracle.

## 2 Méthodologie

### Installation d'Oracle Express

Pour réaliser ce TP, vous devrez installer Oracle Express (10g ou 11g) afin de créer votre environnement de développement. Ce SGBD se retrouve sur le site d'Oracle et nécessitera la création d'un compte (gratuit) pour en faire le téléchargement. De plus, c'est une version gratuite d'Oracle.

### Test de connexion sous SQLPlus

Vous pouvez vous connecter sur votre SGBD en utilisant l'outil « Run SQL Command Line » qui ouvrira SQLPlus. Vous devrez vous connecter (commande connect, username system et mot de passe uqac1234) pour ensuite pouvoir réaliser des opérations SQL (par exemple, select sysdate from dual).

## Création des tables de votre modèle dans l'environnement Oracle

En utilisant un éditeur de texte tel que Notepad++, VSCode ou autre, construisez votre fichier contenant les instructions SQL qui permettront de créer vos tables, vos contraintes (TRIGGERS) et vos procédures stockées.

- Il est plus facile de tester la création et le chargement des tables avant de tester les règles d'intégrité
- Donnez à ce fichier l'extension « .sql »
- N'oubliez pas de débiter votre fichier par la suppression des tables (DROP TABLE). Ainsi, lorsque le fichier sera relancé, il supprimera les tables existantes avant de les recréer.

Voici un exemple de fichier permettant la création de tables :

**cretab.sql**

```
drop table employe;
drop table departement;
create table departement
  (name_dept      char(10)      primary key,
   no_dept        number(2)      unique,
   tel_dept       char(8)       default '545-5011',
   budget         number(7)     default 20000);
create table employe
  (nomat          number(5)      primary key,
   nom_emp        char(15)      not null,
   ville          char(25)      default 'Saguenay',
   commission     number(10),
   salaire        number(5)     default 0,
   nom_dept       char(10)      references departement(name_dept));
create index emp_nom_idx on employe(nom_emp);
```

Lorsque votre fichier est créé, vous pourrez l'exécuter sur SQLDeveloper. Vous pouvez aussi l'exécuter sur SQLPlus, soit en spécifiant le chemin avec la commande @{path}{file} (par exemple @c:\mon\_fichier) ou encore en spécifiant uniquement le nom du fichier si celui-ci se retrouve directement dans le répertoire de l'application SQLPlus.

```
SQL> @cretab
Table dropped.
Table dropped.
Table created.
Table created.
Index created
SQL>
```

*Note : par défaut SQL assume une extension .sql aux fichiers de commandes*

### 3 Travail à réaliser

#### Tâche 1 : Création des tables

Écrire un script SQL nommé « cretab.sql » faisant la création de vos tables (CREATE TABLE).

Consigne 1 : Assurez-vous de bien identifier les contraintes de valeur (NOT NULL) dans vos tables, ainsi que les clés candidates (UNIQUE).

Consigne 2 : Bien que n'êtes pas obligé de mettre des commentaires dans votre script, choisissez des noms simples et descriptifs pour vos tables, colonnes et contraintes, et évitez le plus possible les acronymes obscurs (ex : PK TB FR). Évitez également les accents dans les noms.

#### Tâche 2 : Contraintes et procédures

Pour cette tâche, vous devrez identifier et implémenter les règles d'affaires (contraintes) associées au système. Pour l'implémentation des contraintes identifiées, vous choisirez une des trois approches suivantes :

1. Contrainte statique dans la table (CHECK);
2. Contrainte de domaine à l'aide d'une clé étrangère (FOREIGN KEY) ;
3. Contrainte dynamique complexe (TRIGGER).

Ensuite, vous devrez identifier les opérations qui mettent à jour l'état de la BD et encapsuler ces opérations à l'aide de procédures stockées (CREATE PROCEDURE). Par exemple, une procédure p.AjoutEtudiant pour la création d'un étudiant.

Consigne 1 : Lorsque possible, employez un CHECK au lieu d'un TRIGGER.

Consigne 2 : Pour les contraintes complexes sur les chaînes de caractères, utilisez la fonction d'Oracle REGEXP\_LIKE

Consigne 3 : Assurez-vous de bien tester vos contraintes.

Consigne 4 : Évitez de mettre une clé artificielle comme paramètre d'une procédure d'encapsulation. Par exemple, une procédure p\_AjoutEnseignant devrait seulement prendre en paramètre les informations relatives à l'enseignant, et la clé artificielle (ex : ID\_ENS) devrait être générée à l'intérieur de la procédure avec une SEQUENCE d'Oracle.

### Tâche 3 : Insertion des données

La troisième tâche consiste à créer un fichier SQL nommé « data.sql » qui permettra d'insérer des données dans votre BD selon les consignes suivantes :

Consigne 1 : Créez deux (2) départements

Consigne 2 : Ajoutez cinq (5) cours par département

Consigne 3 : Créez quatre (4) enseignants, les associer à un département et à un cours

### Tâche 4 : Gestion des droits d'accès

Vous devez avoir 2 utilisateurs (database user) distincts dans votre base de données. Le premier utilisateur de type « administrateur/DBA » sera celui avec lequel les tâches 1, 2 et 3 seront réalisées; cet utilisateur aura tous les accès et les tables/vues se retrouveront dans le schéma de ce dernier. Le second utilisateur, « utilisateur standard », devra pouvoir accéder aux données des différentes tables, mais en **lecture seulement**; il ne pourra ni insérer ni modifier les données présentes.

Vous devrez remettre dans votre rapport des captures d'écran des cas suivants :

- Avec l'utilisateur standard, afficher les données de la table Cours (SELECT \* FROM COURS).
- Avec l'utilisateur standard, tentez d'insérer un nouveau cours.
- Avec l'utilisateur standard, tentez d'insérer un enseignant.

### Tâche 5 : Rédaction du rapport

Enfin, la dernière tâche du travail sera de rédiger un rapport décrivant votre travail et justifiant vos décisions de conception.

Note : utilisez le gabarit prévu à cet effet. De plus, vous trouverez dans ce dernier une question théorique à répondre.

## 4 Consignes de remise

Au plus tard le 17 novembre 2025, vous devrez déposer les fichiers demandés sur le Moodle du cours.

### À REMETTRE :

- Le rapport rédigé en suivant les instructions du gabarit fournit avec ce fichier
- Votre fichier SQL permettant la création de vos tables/contraintes/procédures

Note : Si vous avez plusieurs fichiers, ceux-ci doivent être dans un répertoire compressé ayant comme nom 8TRD157 – TP2 [Nom des membres de l'équipe]

Note 2 : Si vous êtes en équipe, une seule remise sur Moodle est suffisante.

## 5 Barème de correction

L'évaluation du travail sera faite sur un total de 100 points, distribués comme suit :

COMPOSANTE	POINTS
Contraintes	35
Procédures	35
Gestion des accès	10
Création des tables	10
Question théorique	10
<b>TOTAL</b>	<b>100</b>