

TD8

Création de schémas – Contraintes d'intégrité

Table des matières

Rappels.....	1
Création d'un schéma de BD avec contraintes d'intégrité.....	2
Exercice 1.....	3

Rappels

Les contraintes d'intégrité permettent à l'utilisateur de définir des règles que doivent respecter les données de la base. Le plus souvent, les contraintes sont définies lors de la création des tables (CREATE TABLE). Elles peuvent néanmoins être rajoutées sur des tables existant sous certaines conditions (ALTER TABLE). Types de contraintes :

- contrainte de clé (PRIMARY KEY et UNIQUE),
- contrainte référentielle (FOREIGN KEY),
- contrainte de domaine (valeurs possibles des attributs, NOT NULL),
- contrainte de tuple (condition liant les valeurs des différents attributs pour tous les n-uplets de la table),
- contraintes globales (sur plusieurs tables) en utilisant la clause CREATE ASSERTION. A noter que certains systèmes, dont Oracle, ne supportent pas les assertions.

Création d'un schéma de BD avec contraintes d'intégrité

On considère le schéma Entreprise décrit ci-dessous.

EMPLOYE (NumSS, NomE, PrenomE, NumChef*, VilleE, DateNaiss, DateEnreg)

PROJET (NumProj, NomProj, RespProj*, VilleP, Budget)

EMBAUCHE (NumSS*, NumProj*, DateEmb, Profil*)

GRILLE_SAL (Profil, salaire)

La clé primaire de chaque relation est soulignée et les attributs des clés étrangères sont suivis d'un astérisque. Cette base contient des informations sur des employés et sur les projets dans lesquels ils sont impliqués. DateEnreg dans la table employe donne la date à laquelle l'employé a été enregistré dans la BD. Ces employés sont embauchés dans un projet sur un profil donné et perçoivent un salaire en fonction de ce profil. En plus des contraintes de clé et contraintes référentielles indiquées dans le schéma, on voudrait intégrer les contraintes suivantes :

On rajoute les contraintes suivantes :

- [C1] Le numéro de sécurité sociale possède exactement 5 chiffres.
- [C2] Les attributs textuels (NomE, PrenomE, NomProj, Profil) ne dépassent pas 20 caractères (ils peuvent en avoir moins).
- [C3] La ville d'un employé (VilleE) ou d'un projet (VilleP) se limite à 'Paris', 'Lyon' et 'Marseille' et sa longueur ne dépasse pas 9 caractères.
- [C4] Par défaut, un employé est enregistré dans la base à la date courante.
- [C5] Il n'y a pas deux employés avec le même nom et le même prénom.
- [C6] Le numéro d'un projet varie entre 5 et 7 chiffres.
- [C7] Chaque projet doit avoir un responsable.
- [C8] Le salaire peut avoir deux chiffres après la virgule et ne dépasse pas 90 000.
- [C9] Aucun employé ne peut avoir plus de 70 ans au moment où il est enregistré dans la table Employé.
- [C10] Un responsable de projet doit habiter la ville du projet dont il est responsable.
- [C11] Le chef d'un employé dans la table Employé et le chef d'un projet dans la table Projet sont des employés.
- [C12] Dans la table Embauche, NumSS et NumProj représentent un employé et un projet existants. De même, profil est un profil existant.

Exercice 1

1. En vous basant sur le type de chacune des contraintes décrites précédemment donné ci-dessous, donner en SQL sous Oracle les instructions de création du schéma ci-dessus avec les contraintes de clés et les contraintes référentielles associées.

- C1, C2, C3 : contrainte de domaine
- C4 n'est pas une contrainte
- C5 : Contrainte de clé
- C6, C7, C8 : contrainte de domaine
- C9 : contrainte de tuple
- C10 : contrainte générale
- C11, C12 : contraintes référentielles

/ Oracle n'a pas de create domain, mais H2 oui*

create domain c1 as number(5) check (length(value) = 5) ;

create domain c2 as varchar(20) ;

create domain c3 as varchar(9) check (lower(value) in ('paris','lyon','marseille')) ;

create domain c6 as number(7) check (length(value) >= 5) ;

create domain c8 as number(7, 2) check (value >= 90000) ;

**/*

create table EMPLOYE (

NumSS number(5) check (length(NumSS)=5) /*check (NumSS >= 10000)*/,
NumSS c1,

NomE varchar(20),
NomE c2,

PrenomE varchar(20),
PrenomE c2,

NumChef number(5) check (length(NumChef)=5),
NumChef c1,

Note : check non obligatoire, car c'est une clef étrangère, ça vérifiera automatiquement si ça le respecte, seulement le type doit être identique

VilleE varchar(9) check (lower(VilleE) in ('paris','lyon','marseille')),
VilleE c3,

DateNaiss date,

DateEnreg date default sysdate,

Note : sysdate, current_date ou une autre syntaxe selon le système de bdd utilisé

constraint pk_emp primary key(NumSS),

constraint unique_nom_prenom unique(NomE, PrenomE),

constraint fk_emp foreign key(NumChef) references EMPLOYE,

constraint limite_age_emp check (datediff(year, DateNaiss, DateEnreg) <= 70.0)

);

```
create table PROJET (  
    NumProj number(7) check (length(NumProj) >= 5) /*check (NumProj >= 10000)*/,  
    NumProj c6,  
  
    NomProj varchar(20),  
    NomProj c2,  
  
    RespProj number(5) NOT NULL,  
    RespProj c1 NOT NULL,  
  
    VilleP varchar(9) check (lower(VilleP) in ('paris','lyon','marseille')),  
    VilleP c3,  
  
    Budget real,  
    constraint pk_proj primary key(NumProj),  
    constraint fk_proj foreign key(RespProj) references EMPLOYE  
);  
  
create table GRILLE_SAL (  
    Profil varchar(20),  
    Profil c2,  
  
    Salaire number(7, 2) check (Salaire >= 90000),  
    Salaire c8,  
  
    constraint pk_sal primary key(Profil),  
);  
  
create table EMBAUCHE (  
    NumSS number(5),  
    NumSS c1,  
  
    NumProj number(7),  
    NumProj c6,  
  
    DateEmb date,  
    Profil varchar(20),  
    Profil c2,  
  
    constraint pk_emb primary key(NumSS, NumProj),  
    constraint fk_emb_emp foreign key(NumSS) references EMPLOYE,  
    constraint fk_emb_proj foreign key(NumProj) references PROJET,  
    constraint fk_emb_sal foreign key(Profil) references GRILLE_SAL,  
);
```

2. Donner en SQL, lorsque c'est possible, les instructions permettant d'exprimer les contraintes C7 et C10 en indiquant à chaque fois si il s'agit de contrainte pouvant être vérifiée localement ou nécessitant une vérification globale.

Pas encore fait