TD 6 : Contraintes

Définition de données Contraintes

2024-10-25

Avec solutions

- **L3 MIASHS/Ingémath**
- **Université Paris Cité**
- Année 2024-2025
- Course Homepage

- Moodle

Contraintes SQL

Contraintes CHECK.

Cette contrainte permet de vérifier que les colonnes d'une ligne donnée vérifient une certaine condition. Par exemple, on pourrait vouloir vérifier que la valeur de la colonne `percentage` de la table `countrylanguage` est inférieure à 100 ou que la valeur de la colonne `return_date` de la table `rental` est soit `NULL` ou supérieure à `rental_date`. Ces contraintes se définissent lors de la création de la table. Par exemple :

```
CREATE TABLE solde(
    produit TEXT,
    reduc_pourcentage INT,
    CHECK (reduc_pourcentage<=100 AND reduc_pourcentage%10=0)
);
```

On peut ajouter une contrainte CHECK à une table qui existe déjà. Par exemple :

```
ALTER TABLE solde ADD
    CHECK (reduc_pourcentage >=0);
```

ou bien en nommant la contrainte :

```
ALTER TABLE solde
    ADD CONSTRAINT reduc_positive
    CHECK (reduc_pourcentage >=0);
```

Contraintes `DEFAULT / NOT NULL / UNIQUE`

Ces contraintes s'appliquent à une colonne en particulier. La contrainte `DEFAULT` spécifie une valeur par défaut pour remplir une colonne lorsque sa valeur n'est pas spécifiée, `NOT NULL` que la valeur de la colonne ne peut pas être `NULL` et `UNIQUE` que deux lignes différentes de la table ne peuvent pas avoir la même valeur sur les colonnes indiquées. Par exemple :

```
CREATE TABLE membre(
    nom VARCHAR(50) NOT NULL,
    prenom VARCHAR(50) NOT NULL,
    date_inscription DATE DEFAULT NOW(),
```

```
    UNIQUE(nom, prenom)
);
```

Contraintes PRIMARY/FOREIGN KEY.

Formellement, la contrainte `PRIMARY KEY` est équivalente à `UNIQUE` et `NOT NULL`. C'est un moyen d'identifier de façon unique chaque ligne. Chaque table est censée avoir une clé primaire. La contrainte `FOREIGN KEY` spécifie que la valeur d'une (ou de plusieurs) colonne contient des valeurs présentes dans une autre table. Par exemple, la colonne `capital` de `country` contient une valeur qui doit apparaître dans la colonne `id` de `city`. Par exemple :

```
CREATE TABLE membre(
    nom VARCHAR(50) NOT NULL,
    prenom VARCHAR(50) NOT NULL,
    id INT,
PRIMARY KEY (id)
);

CREATE TABLE amis(
    id1 INT,
    id2 INT,
  FOREIGN KEY (id1) REFERENCES membre (id)
);
```

On peut aussi ajouter une seconde clef étrangère comme ceci :

```
ALTER TABLE amis ADD
    FOREIGN KEY (id2) REFERENCES membre;
```

Contraintes EXCLUDE.

Les contraintes `NOT NULL, DEFAULT, KEY, CHECK` portent uniquement sur une ligne tandis que la contrainte `UNIQUE` permet uniquement de tester des égalités entre lignes. On veut parfois vérifier une condition plus riche sur deux lignes. Par exemple, si notre table contient des réservations d'une salle entre `start_date` et `end_date`, on ne veut pas que deux réservations se chevauchent. On écrira :

```
CREATE TABLE reservation(
    start_date DATE,
    end_date DATE,

    EXCLUDE USING gist (
        daterange(start_date, end_date) WITH &&)
);
```

Exercices

On va travailler sur une copie du schéma `world`. Il va donc falloir copier dans vos schémas les tables `world.country`, `world.city` et `world.countrylanguage`.

```
CREATE TABLE chaprot.td_country (like world.country) ;
```

Va copier les champs avec leur type et les contraintes `NOT NULL`. Mais aucune autre contrainte. Par contre, la commande :

```
CREATE TABLE chaprot.td_country (
    like world.country
    including all
) ;
```

va copier toutes les contraintes SAUF celles qui font référence à d'autres tables (clé étrangère, ...).

Question

- Copier les trois tables de `world` dans votre schéma (`chaprot`).
- Remplir les tables avec les valeurs qui sont dans les tables originelles.

Solution

```
INSERT INTO chaprot.td_country
    SELECT * FROM world.country ;

INSERT INTO chaprot.td_countrylanguage
    SELECT * FROM world.countrylanguage;

INSERT INTO chaprot.td_city
    SELECT * FROM world.city;
```

Question

Recréer les contraintes qui n'ont pas été copiées pour obtenir des tables qui soient réellement identiques.
Vérifier que les résultats de
`\d world.country`
et
`\d chaprot.td_country`
sont bien égaux.

Solution

```
ALTER TABLE chaprot.td_country
    ADD FOREIGN KEY (capital)
        REFERENCES chaprot.td_city (id);

ALTER TABLE chaprot.td_countrylanguage
    ADD FOREIGN KEY (countrycode)
        REFERENCES chaprot.td_country (countrycode);
```

Question

Ajouter la contrainte : La colonne `countrycode` de `chaprot.td_city` est une référence à la clé primaire de `chaprot.td_country`.

Solution

```
alter table chaprot.td_city
    add foreign key (countrycode)
        references chaprot.td_country(countrycode);
```

Question

Ajouter la contrainte : La colonne `percentage` de `chaprot.td_countrylanguage` contient un réel compris entre 0 et 100.

Solution

```
alter table chaprot.td_countrylanguage
    add check  (cast(0.0 as real)  < percentage and
                percentage <= cast (100 as real)
) ;
```

Ne marche pas car c'est le cas pour certaines lignes. il faut les supprimer !

Question

Ajouter la contrainte : Dans `chaprot.td_countrylanguage`, il n'y a pas deux lignes qui parlent du même pays et de la même langue.

Solution

```
alter table chaprot.td_countrylanguage add unique (countrycode,language)  ;
```

mais c'est inutile car (`countrycode,language`) est déjà une `PRIMARY KEY`.

Question

Ajouter la contrainte : Dans `chaprot.td_country`, pour chaque pays, il n'existe pas un autre pays de la même `region` qui n'est pas dans le même `continent`, soit la *dépendance fonctionnelle* : `region → continent`

Solution

```
create or replace function chaprot.check_reg (reg text, conti text)
returns bigint
language sql as
$$
select count (*)
from chaprot.td_country a
where a.region = reg and not a.continent = conti;
$$

Alter table chaprot.td_country add constraint RegionContinent check (chaprot.check_reg (region
```

OU

```
ALTER TABLE chaprot.td_country
    ADD CONSTRAINT df_region_continent
        EXCLUDE USING gist (region WITH =, continent WITH <>) ;
```

La seconde formulation de la contrainte souligne qu'il s'agit d'une contrainte de table et pas d'une contrainte de ligne.