

# BASE DE DONNEES - ACHATS

*Théo Lorio / Gabriel Bailleul / Louis Iwinski*



06/10/2021

# Base de données - Achats

*Théo Lorio / Gabriel Bailleul / Louis Iwinski*

## Sommaire

### Étape 1 page 2-5

- Diagramme E/R de la base de données
- Description sommaire de la base de données
- Tests des contraintes et affichages des erreurs

### Étape 2 page 6

- Diagramme E/R de la base de données sans contraintes d'intégrités

# Étape 1

## Diagramme E/R

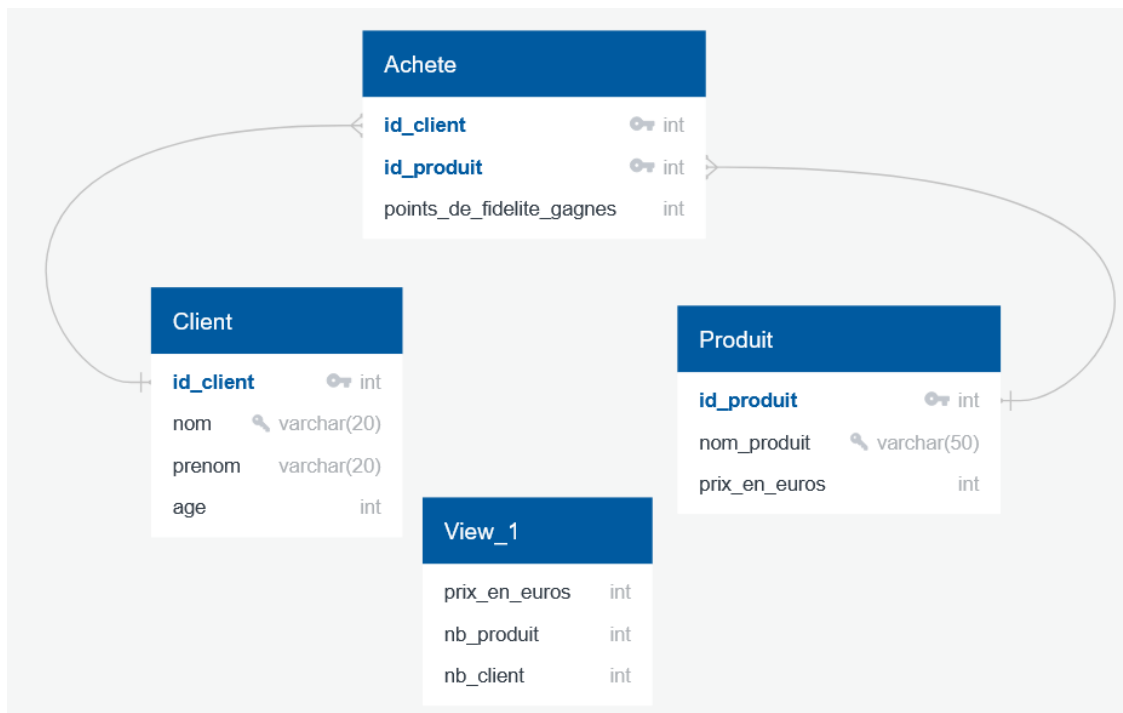


Diagramme E/R de notre base de données Achats réalisé sur QuickDBD

Légende :

 Unique
  Primary Key
  varchar(50)
  int Type des données

## Description sommaire de la base de données

Durant ce premier semestre, en groupe de 3 (Louis Iwinski, Théo Lorio et Gabriel Bailleul), on nous a demandé de réaliser une activité, dans le cadre du cours Base de données avancées, qui permet de créer une base de données. Nous avons décidé de la nommer « Achats » et elle comporte trois tables (« Client », « Produit » et « Achete ») et une vue (« view\_1 »).

```

achats=# \dt
          Liste des relations
Schema |   Nom   | Type  | Propriétaire
-----+-----+-----+-----
public | achete  | table | postgres
public | client  | table | postgres
public | produit | table | postgres
(3 lignes)

```

Tableau des trois tables de la base de données « Achats »

Cette base de données pourrait s'apparenter à une base qui pourrait être utilisée dans un magasin de sport et de vêtements. On peut le voir grâce aux tuples insérés dans les différentes tables de base de données.

Par exemple :

```

achats=# select * from produit;
 id_produit |  nom_produit  | prix_en_euros
-----+-----+-----
          1 | Ballon de basket |          15
          2 | T-shirt RC Lens |          60
          3 | Sac           |          20
          4 | Kayak         |         150
          5 | Chaussettes   |           8
          6 | Protéines     |           4
          7 | Sacoche       |           7
          8 | Bermudas     |          15
          9 | Eau           |           3
         10 | Chasuble     |          10
         11 | Chaussures   |          50
         12 | Drapeau      |          14
         13 | Protege genou |          40
         14 | Panier basket |          80
         15 | Velo         |         160
(15 lignes)

```

Tableau de tuples de la table Produit

Les tables de cette base de données sont créées avec plusieurs contraintes. Notamment, avec des « NOT NULL » ou encore des « UNIQUE ». Ce qui

permet de ne pas faire d'erreurs quand on insère de nouvelles valeurs dans les tables en question.

```
create table client(
  id_client int primary key,
  nom varchar(20) not null,
  prenom varchar(20) not null,
  age int,
  check(id_client>0),
  check(age>0),
  unique(nom),
  unique(id_client)
)
```

```
create table produit(
  id_produit int primary key,
  nom_produit varchar(50) not null,
  prix_en_euros int not null,
  check(prix_en_euros>=0),
  check(id_produit>0),
  unique(nom_produit)
)
```

```
create table achete(
  id_client int,
  id_produit int,
  points_de_fidelite_gagnes int,
  primary key(id_produit,id_client),
  foreign key(id_client) references client(id_client) on delete cascade on update cascade,
  foreign key(id_produit) references produit(id_produit) on delete cascade on update cascade,
  check(points_de_fidelite_gagnes>=0)
)
```

### Capture de la création des tables de la base de données

On a donc trois tables. La première se nomme Client et regroupe l'identifiant (id\_client), le nom (nom), le prénom (prenom) ainsi que l'âge (age) du client. La deuxième se nomme Produit et regroupe l'identifiant (id\_produit), le nom du produit (nom\_produit) et le prix en euros (prix\_en\_euros) du produit. Pour finir, on a la table Achète qui regroupe les deux clés primaires des deux autres tables (id\_client et id\_produit).

Chaque table comporte quinze tuples différentes.

## Tests des contraintes et affichages des erreurs

- Test numéro 4 Client– sur la contrainte NOT NULL sur nom de Client

```
achats=# insert into client values(16,null,'Marc',17);
ERREUR:  une valeur NULL viole la contrainte NOT NULL de la colonne « nom » dans la relation « client »
DÉTAIL : La ligne en échec contient (16, null, Marc, 17).
```

- Test numéro 6 Client – sur la contrainte de l'age>0 sur age de Client

```
achats=# insert into client values(16,'Marc','Oracle',0);
ERREUR: la nouvelle ligne de la relation « client » viole la contrainte de vérification « client_age_check »
DÉTAIL : La ligne en échec contient (16, Marc, Oracle, 0).
```

- Test numéro 3 Produit – Contrainte UNIQUE sur nom\_produit de Produit

```
achats=# insert into produit values(18,'Velo',3);
ERREUR: la valeur d'une clé dupliquée rompt la contrainte unique « produit_nom_produit_key »
DÉTAIL : La clé « (nom_produit)=(Velo) » existe déjà.
```

- Test numéro 8 Produit – Contrainte varchar(50) sur nom\_produit de  
Produit

[illegible]

- Test numéro 1 Achete – Contrainte UNIQUE sur (id\_client,id\_produit) de Achete

```
achats=# insert into achete values(1,6,15);
ERREUR:  la valeur d'une clé dupliquée rompt la contrainte unique « achete_pkey »
DÉTAIL : La clé « (id_produit, id_client)=(6, 1) » existe déjà.
```

## Étape 2

### Diagramme E/R sans contrainte d'intégrité



Diagramme E/R de notre base de données Achats sans contraintes d'intégrité réalisé sur QuickDBD