

# Evaluation

## **COMPETENCE 2 : CONCEVOIR UNE BASE DE DONNEES**

*Réalisation des 4 étapes de conceptions*

# SOMMAIRE

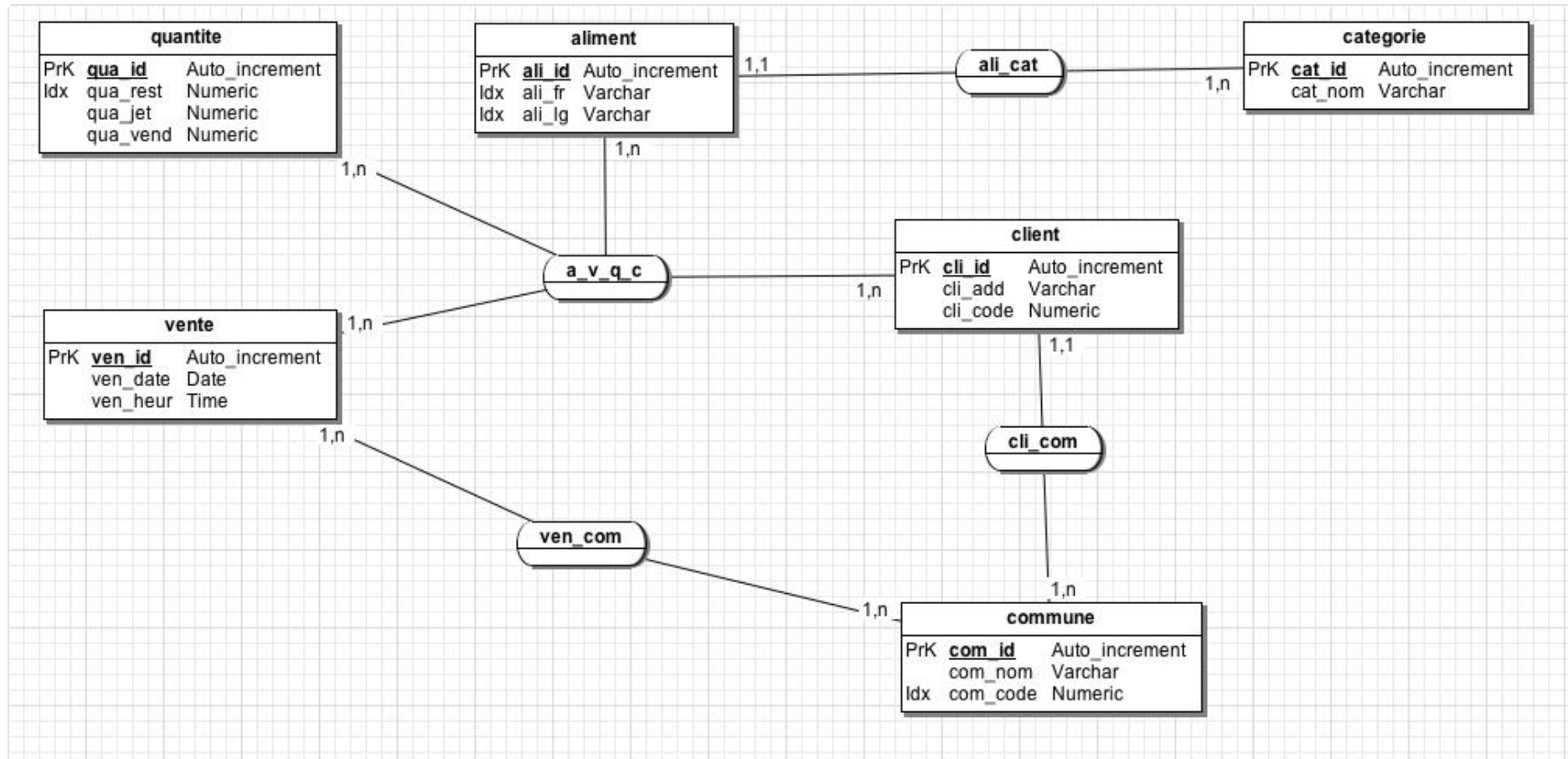
- I. Le dictionnaire de données
- II. Le MCD
- III. Le MLD
- IV. Le MPD

# I. Le dictionnaire de données

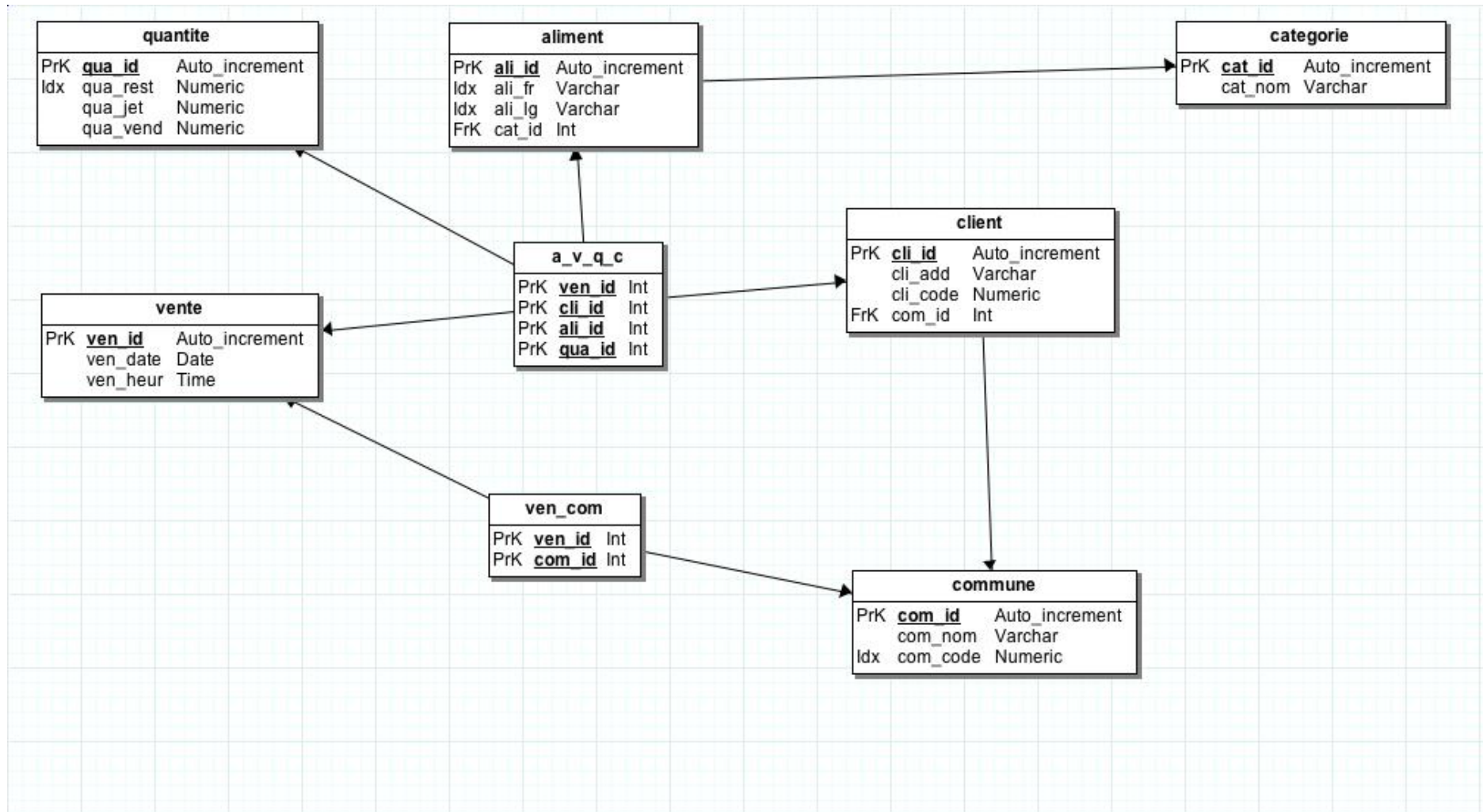
Dictionnaire de données					
Description de la donnée	Type	Longueur	Exemple	Nom de la colonne	Contraintes
identifiant de la table quantité	numérique	25	2	qua_id	obligatoire / auto-incrémentation / clé primaire
la quantité restante donc le stock	numérique	25	12	qua_rest	obligatoire / index / alert nb = 5
la quantité jetée de fruit ou de légume	numérique	25	5	qua_jet	obligatoire
la quantité vendue de fruit ou de légume	numérique	25	4	qua_vend	obligatoire
identifiant de la table aliment	numérique	25	6	ali_id	obligatoire / auto-incrémentation / clé primaire
nom du fruit	alphanumérique	30	fraise	ali_fr	obligatoire / index
nom du légume	alphanumérique	30	poireau	ali_lg	obligatoire / index
identifiant de la table catégorie	numérique	25	3	cat_id	obligatoire / auto-incrémentation / clé primaire
nom de la catégorie de l'aliment	alphanumérique	25	fruit	cat_nom	obligatoire
identifiant de la table client	numérique	25	1	cli_id	obligatoire / auto-incrémentation / clé primaire
adresse du client	alphanumérique	100	156 rue Avenue du Maréchal Poireau	cli_add	
code postal du client	numérique	5	9100	cli_code	

identifiant de la table vente	numérique	25	1	ven_id	obligatoire / auto-incrémentation / clé primaire
date de la vente (JJ/MM/AA)	date	8	08/03/18	ven_date	obligatoire
heure de la vente (HH:MM:SS)	time	8	11:16:04	ven_heur	obligatoire
identifiant de la table commune	numérique	25	1	com_id	obligatoire / auto-incrémentation / clé primaire
nom de la commune	alphanumérique	30	pamiers	com_nom	obligatoire
code postal de la commune	numérique	5	9100	com_code	obligatoire / index

## II. Le MCD



### III. Le MLD



## IV. Le MPD

Le SGBD choisi est PostgreSQL.

```
-----  
--      Script Postgre  
-----
```

```
-----  
-- Table: quantite  
-----
```

```
CREATE TABLE public.quantite(  
    qua_id SERIAL NOT NULL ,  
    qua_rest NUMERIC (25,0) NOT NULL ,  
    qua_jet NUMERIC (25,0) NOT NULL ,  
    qua_vend NUMERIC (25,0) NOT NULL ,  
    CONSTRAINT prk_constraint_quantite PRIMARY KEY (qua_id)  
)WITHOUT OIDS;
```

-- Table: aliment

```
-----  
CREATE TABLE public.aliment(  
    ali_id SERIAL NOT NULL ,  
    ali_fr VARCHAR (30) NOT NULL ,  
    ali_lg VARCHAR (30) NOT NULL ,  
    cat_id INT  NOT NULL ,  
    CONSTRAINT prk_constraint_aliment PRIMARY KEY (ali_id)  
)WITHOUT OIDS;
```

-- Table: categorie

```
-----  
CREATE TABLE public.categorie(  
    cat_id SERIAL NOT NULL ,  
    cat_nom VARCHAR (25) NOT NULL ,  
    CONSTRAINT prk_constraint_categorie PRIMARY KEY (cat_id)  
)WITHOUT OIDS;
```

-- Table: client

```
-----  
CREATE TABLE public.client(  
    cli_id SERIAL NOT NULL ,  
    cli_add VARCHAR (100) ,
```



```
    cli_code NUMERIC (5,0) ,
    com_id INT NOT NULL ,
    CONSTRAINT prk_constraint_client PRIMARY KEY (cli_id)
)WITHOUT OIDS;
```

```
-----
-- Table: vente
-----
```

```
CREATE TABLE public.vente(
    ven_id SERIAL NOT NULL ,
    ven_date DATE NOT NULL ,
    ven_heur TIMETZ NOT NULL ,
    CONSTRAINT prk_constraint_vente PRIMARY KEY (ven_id)
)WITHOUT OIDS;
```

```
-----
-- Table: commune
-----
```

```
CREATE TABLE public.commune(
    com_id SERIAL NOT NULL ,
    com_nom VARCHAR (30) NOT NULL ,
    com_code NUMERIC (5,0) NOT NULL ,
    CONSTRAINT prk_constraint_commune PRIMARY KEY (com_id)
)WITHOUT OIDS;
```

-----  
-- Table: a\_v\_q\_c  
-----

```
CREATE TABLE public.a_v_q_c(  
    ven_id INT NOT NULL ,  
    cli_id INT NOT NULL ,  
    ali_id INT NOT NULL ,  
    qua_id INT NOT NULL ,  
    CONSTRAINT prk_constraint_a_v_q_c PRIMARY KEY (ven_id,cli_id,ali_id,qua_id)  
)WITHOUT OIDS;
```

-----  
-- Table: ven\_com  
-----

```
CREATE TABLE public.ven_com(  
    ven_id INT NOT NULL ,  
    com_id INT NOT NULL ,  
    CONSTRAINT prk_constraint_ven_com PRIMARY KEY (ven_id,com_id)  
)WITHOUT OIDS;
```

```
CREATE INDEX quantite_qua_rest_idx ON quantite (qua_rest);  
CREATE INDEX aliment_ali_fr_idx ON aliment (ali_fr);  
CREATE INDEX aliment_ali_lg_idx ON aliment (ali_lg);  
CREATE INDEX commune_com_code_idx ON commune (com_code);
```

```

ALTER TABLE public.aliment ADD CONSTRAINT FK_aliment_cat_id FOREIGN KEY (cat_id) REFERENCES
public.categorie(cat_id);
ALTER TABLE public.client ADD CONSTRAINT FK_client_com_id FOREIGN KEY (com_id) REFERENCES
public.commune(com_id);
ALTER TABLE public.a_v_q_c ADD CONSTRAINT FK_a_v_q_c_ven_id FOREIGN KEY (ven_id) REFERENCES
public.vente(ven_id);
ALTER TABLE public.a_v_q_c ADD CONSTRAINT FK_a_v_q_c_cli_id FOREIGN KEY (cli_id) REFERENCES
public.client(cli_id);
ALTER TABLE public.a_v_q_c ADD CONSTRAINT FK_a_v_q_c_ali_id FOREIGN KEY (ali_id) REFERENCES
public.aliment(ali_id);
ALTER TABLE public.a_v_q_c ADD CONSTRAINT FK_a_v_q_c_qua_id FOREIGN KEY (qua_id) REFERENCES
public.quantite(qua_id);
ALTER TABLE public.ven_com ADD CONSTRAINT FK_ven_com_ven_id FOREIGN KEY (ven_id) REFERENCES
public.vente(ven_id);
ALTER TABLE public.ven_com ADD CONSTRAINT FK_ven_com_com_id FOREIGN KEY (com_id) REFERENCES
public.commune(com_id);

```

Les colonnes indexés sont les suivantes : qua\_rest, ali\_fr, ali\_lg, com\_code.

J'ai indexé ces colonnes pour répondre rapidement aux requêtes suivantes :

1. visualiser, sur un page 'géomarketing', les 5 communes les plus consommatrices de ses produits, en temps réel.
2. envoyer une alerte sur la page d'accueil de l'interface dès qu'il lui reste 5kg ou 5 unités d'un produit.