

# TP — Informatique Décisionnelle & OLAP avec BigQuery

Jeu de données public : Iowa Liquor Sales

## Objectifs

- interroger un **entrepôt de données cloud** (BigQuery)
- analyser des données avec le **SQL OLAP**
- utiliser :
  - GROUP BY
  - GROUPING SETS
  - ROLLUP
  - CUBE
  - GROUPING ( )
- raisonner en termes de **cube décisionnel**

## Jeu de données utilisé

### Description

Le jeu de données **Iowa Liquor Sales** contient l'ensemble des ventes d'alcool dans l'État de l'Iowa.

Chaque ligne représente une **vente** avec :

- une date
- une zone géographique
- un produit
- des montants financiers

# Table publique BigQuery

```
`bigquery-public-data.iowa_liquor_sales.sales`
```

## Aperçu des attributs utiles

Colonne	Rôle OLAP
date	Dimension Temps
county	Dimension Géographie
city	Dimension Géographie
category_name	Dimension Produit
vendor_name	Dimension Fournisseur
bottles_sold	Mesure
sale_dollars	Mesure (chiffre d'affaires)

## Cube OLAP logique

### Table de faits

FACT\_SALES

### Dimensions

- **Temps** : date → mois → année
- **Géographie** : city → county
- **Produit** : category\_name
- **Fournisseur** : vendor\_name

### Mesures

- SUM(sale\_dollars) → chiffre d'affaires
- SUM(bottles\_sold) → volume

# PARTIE 1 — Prise en main BigQuery

1. Ouvrir la **console Google Cloud**
2. Aller dans **BigQuery**
3. Lancer la requête suivante :

```
SELECT *  
FROM `bigquery-public-data.iowa_liquor_sales.sales`  
LIMIT 10;
```

## PARTIE 2 — Agrégations simples ( GROUP BY )

### Total des ventes par catégorie de produit

```
SELECT  
    category_name,  
    SUM(sale_dollars) AS total_sales  
FROM `bigquery-public-data.iowa_liquor_sales.sales`  
GROUP BY category_name  
ORDER BY total_sales DESC;
```

## PARTIE 3 — Analyse temporelle

### Chiffre d'affaires par année

```
SELECT  
    EXTRACT(YEAR FROM date) AS year,  
    SUM(sale_dollars) AS total_sales  
FROM `bigquery-public-data.iowa_liquor_sales.sales`  
GROUP BY year  
ORDER BY year;
```

## PARTIE 4 — ROLLUP (hiérarchie temporelle)

### Ventes par année + total global

```
SELECT
  EXTRACT(YEAR FROM date) AS year,
  SUM(sale_dollars) AS total_sales,
  GROUPING(EXTRACT(YEAR FROM date)) AS is_total
FROM `bigquery-public-data.iowa_liquor_sales.sales`
GROUP BY ROLLUP(year)
ORDER BY year;
```

is\_total = 1 → total global

## PARTIE 5 — GROUPING SETS

### Ventes :

- par année
- par county
- total global

```
SELECT
  EXTRACT(YEAR FROM date) AS year,
  county,
  SUM(sale_dollars) AS total_sales
FROM `bigquery-public-data.iowa_liquor_sales.sales`
GROUP BY GROUPING SETS (
  (year),
  (county),
  ()
);
```

## PARTIE 6 — CUBE (analyse croisée)

### Ventes par :

- county
- catégorie
- toutes les combinaisons

```
SELECT
  county,
  category_name,
  SUM(sale_dollars) AS total_sales
FROM `bigquery-public-data.iowa_liquor_sales.sales`
GROUP BY CUBE(county, category_name);
```

Attention à l'explosion combinatoire (jeu volumineux)

## PARTIE 7 — Fonction GROUPING()

### Identifier les agrégats

```
SELECT
  county,
  SUM(sale_dollars) AS total_sales,
  GROUPING(county) AS agg
FROM `bigquery-public-data.iowa_liquor_sales.sales`
GROUP BY ROLLUP(county);
```

agg	Signification
0	valeur réelle
1	total agrégé

# PARTIE 8 — Exercices

## Exercice 1

Afficher le chiffre d'affaires par **ville**.

## Exercice 2

Afficher le chiffre d'affaires par **county** pour l'année 2022.

## Exercice 3

Afficher les ventes pour :

- catégories *Whiskey* et *Vodka*
- counties *Polk* et *Linn*

## Exercice 4

Afficher :

- a) les ventes par county
- b) le total toutes counties confondues

Utiliser `ROLLUP`

## Exercice 5

Classer les counties par chiffre d'affaires croissant.

## Exercice 6

Filtrer uniquement les counties dont le chiffre d'affaires dépasse **10 millions**.

## Exercice 7

Calculer **la moyenne des ventes par catégorie**.