

GROUP 2: Teddy & Ronalyn

KPIS RETENUS

NOM KPI	Définition	Tables associées
CA au mois	Permet d'avoir la somme totale des ventes par mois	orders
Produit le plus/moins vendu	Permet de savoir quel produit est actuellement en tendance, permet aussi de gérer les stocks	order_items, product
Nombre de nouveaux clients chaque mois	Permet de voir les temps forts/creux du site e-commerce	customers
Prix d'un panier moyen par mois	Permet de connaître le pouvoir d'achat des clients en fonction du mois	orders
Temps moyen entre la première inscription et la première commande	Permet de voir l'attractivité du site, s'il est vendeur ou non	customers, orders

KPI 1 : CA au mois

```
SELECT d.month, d.year, SUM(amount) as ca_par_mois FROM fact_sales fs
JOIN dates d ON fs.order_date_id = d.id_date
GROUP BY d.month, d.year
ORDER BY d.year ASC, d.month ASC;
```

KPI 2 : Produit le plus/moins vendu

```
WITH ventes AS (
  SELECT p.name,
         SUM(fs.quantity) AS nb_ventes
  FROM fact_sales fs
  JOIN products p ON p.id_product = fs.id_product
  GROUP BY p.id_product, p.name
)
SELECT name, nb_ventes
FROM ventes
WHERE nb_ventes = (SELECT MAX(nb_ventes) FROM ventes)
OR nb_ventes = (SELECT MIN(nb_ventes) FROM ventes);
```

KPI 3: Nombre de nouveaux clients chaque mois

```
SELECT
  d.annee,
  d.mois,
  count(c.id_customer) AS nouveaux_clients
FROM customers c
  JOIN dates d ON c.signup_date_id = d.id_date
GROUP BY d.annee, d.mois
ORDER BY d.annee, d.mois;
```

KPI 4: Prix d'un panier moyen par mois

```
SELECT
  d.year,
  d.month,
  ROUND(SUM(fs.amount)::numeric / COUNT(DISTINCT fs.id_order), 2) AS panier_moyen
FROM fact_sales fs
  JOIN dates d ON fs.order_date_id = d.id_date
GROUP BY d.year, d.month
ORDER BY d.year, d.month;
```

KPI 5: Temps moyen entre la première inscription et la première commande

```
WITH first_signup AS (
  SELECT customers.id_customer,
    min(customers.signup_date) AS signup_date
  FROM ecommerce.customers
  GROUP BY customers.id_customer
), first_order_with_delta AS (
  SELECT fs.id_customer,
    fs.signup_date,
    min(o.order_date) AS first_order_date,
    min(o.order_date) - fs.signup_date AS delta_days
  FROM first_signup fs
    JOIN ecommerce.orders o ON o.id_customer = fs.id_customer AND o.order_date
    >= fs.signup_date
  GROUP BY fs.id_customer, fs.signup_date
)
SELECT to_char(date_trunc('month'::text, first_order_with_delta.signup_date::timestamp
with time zone), 'YYYY-MM'::text) AS mois_inscription,
  round(avg(first_order_with_delta.delta_days), 2) AS delai_moyen_jours
FROM first_order_with_delta
```

```
WHERE first_order_with_delta.first_order_date IS NOT NULL
GROUP BY (date_trunc('month'::text, first_order_with_delta.signup_date::timestamp with
time zone))
ORDER BY (to_char(date_trunc('month'::text,
first_order_with_delta.signup_date::timestamp with time zone), 'YYYY-MM'::text));
```