



REPUBLIQUE DU SENEGAL



ECOLE SUPERIEURE MULTINATIONALE DES TELECOMMUNICATIONS

Classe : INGC 2

Option : Ingénierie des Données et Intelligence Artificielle

Cours : Base de données NoSQL

**Projet 2 : Réalisation d'un tableau
de bord à l'aide MongoDB Charts**

Présenté par :

Lydie Larhidi NGARDOLEGOM

Professeur : Mr Preira

Année Académique : 2022-2023


PLAN

I. Environnement	3
II. Dataset	8
1. Choix	8
2. Description	8
III. Mise en œuvre des graphiques	9
1) Graphique 1 : <i>Nombre d'utilisateurs par Pays</i>	10
2) Graphique 2 : <i>Nombre d'abonnements par dispositif de visionnage</i>	11
3) Graphique 3 : <i>Nombre d'abonnements par type d'abonnement</i>	13
4) Graphique 4 : <i>Nombre d'abonnements par tranche d'âge</i>	14
5) Graphique 5 : <i>Nombre d'abonnements par outil de visionnage et par genre</i>	15
6) Graphique 6 : <i>Nombre d'abonnements par type d'abonnement et par tranche d'âge</i>	17
7) Graphique 7 : <i>Nombre d'abonnements par revenu et par date de dernier paiement</i>	18
8) Graphique 8 : <i>Nombre d'abonnements par pays et par tranche d'âge</i>	20
9) Graphique 9 : <i>Nombre d'abonnements par type d'abonnement et par pays</i>	21
10) Graphique 10 : <i>Nombre total d'abonnees</i>	22
IV. Présentation finale du Dashboard	24

I. Environnement


1. Procédure d'installation de MongoDB Charts


Pour procéder à l'installation de MongoDB Charts, il faudrait commencer par créer un compte sur le site [MongoDB Atlas Charts / MongoDB](#) et ensuite créer un cluster qui sera utilisé.



Create your account

Have an account? [Log in now](#)

 Google

 GitHub

Or with email and password

Email Address
We recommend using your work email

lydiengar@gmail.com ✓

First Name

Lydie ✓

Password

..... ✓

Your password must:


- Contain at least **8 characters**
- Contain unique characters, numbers, or symbols
- Not contain your email address

Company Name

Optional

☒ I accept the [Privacy Policy](#) and the [Terms of Service](#)

✓ Je ne suis pas un robot


reCAPTCHA
Confidentialité - Conditions

Sign up

Create a Shared Cluster




Serverless

Dedicated







Shared

For learning and exploring MongoDB in a sandbox environment. Basic configuration controls.
No credit card required to start. Upgrade to dedicated clusters for full functionality.
Explore with sample datasets. Limit of one free cluster per project.

Cloud Provider & Region

★ Recommended region ⓘ ⓘ Dedicated tier region ⓘ

NORTH AMERICA	EUROPE	AUSTRALIA
 Oregon (us-west-2) ★	 Paris (eu-west-3) ★	 Sydney (ap-southeast-2) ★
 N. Virginia (us-east-1) ★	 Stockholm (eu-north-1) ★	 Melbourne (ap-southeast-4) ★

FREE

Free forever! Your M0 cluster is ideal for experimenting in a limited sandbox. You can upgrade to a production cluster anytime.

Cancel

Create Cluster

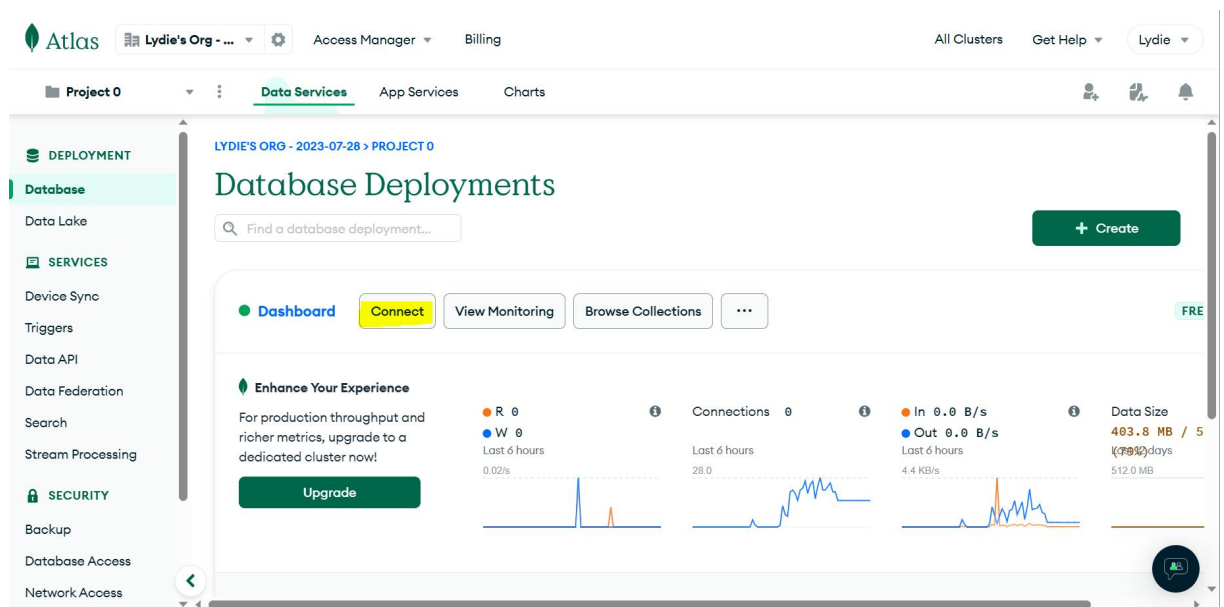
2. Procédure de déploiement de la base de données sur MongoDB Atlas

Une fois que l'environnement a bien été préparé, nous allons passer à la procédure de déploiement de notre base de données sur MongoDB Atlas.

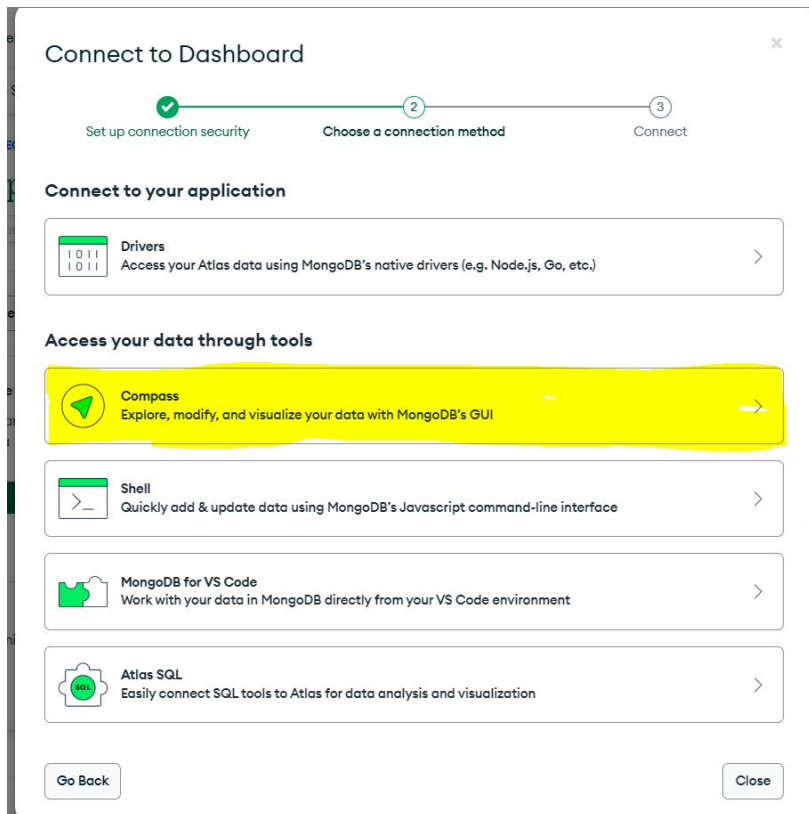
Cela se fait suivant les étapes suivantes :



Sur cette page, nous allons sélectionner **Connect** pour choisir le mode par lequel on charge nos données.

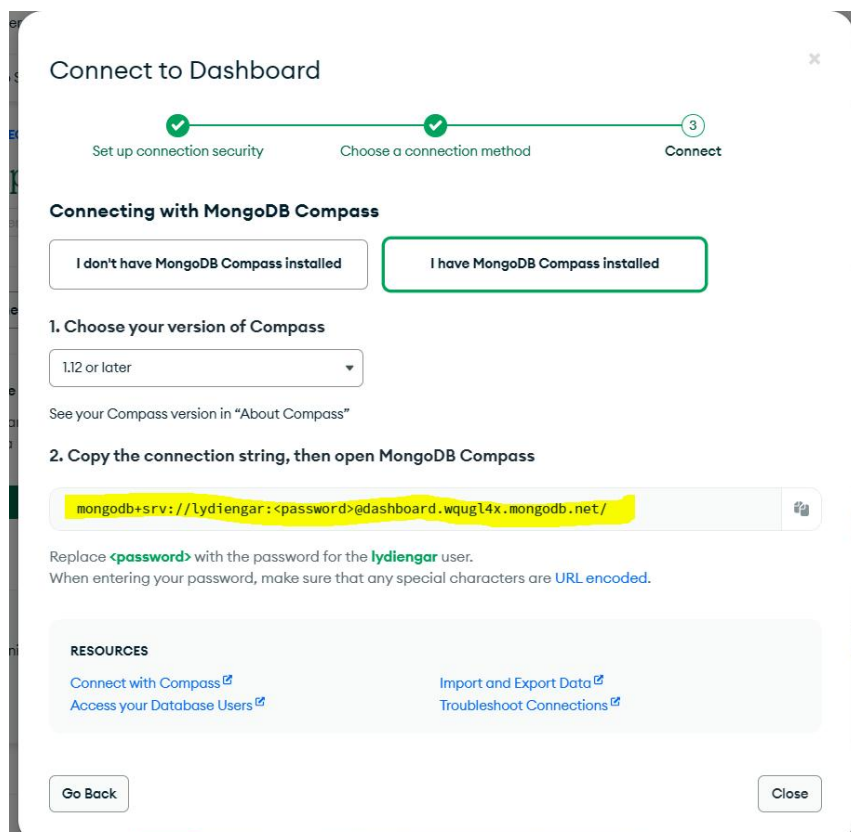


- Une fois cela fait, nous allons choisir **Compass**, qui est une interface graphique officielle fournie par MongoDB pour explorer, gérer et interagir avec des bases de données MongoDB.



Nous aurons cette interface.

Vu que nous avons déjà eu à utiliser MongoDB Compass, nous allons sélectionner cette case pour pouvoir charger notre dataset.



Nous allons copier ces chaînes de caractères qui comportent nos informations de connexion, ensuite lancer notre MongoDB Compass.

New Connection

Connect to a MongoDB deployment

URI ⓘ

Edit Connection String ☒

mongodb+srv://lydiengar:<ingc2023>@dashboard.wqugl4x.mongodb.net/

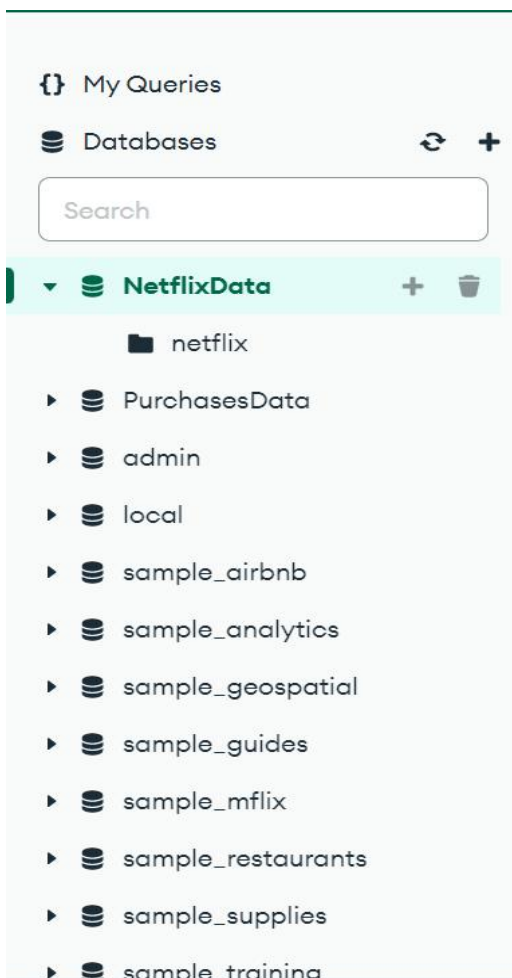
> Advanced Connection Options

Save

Save & Connect

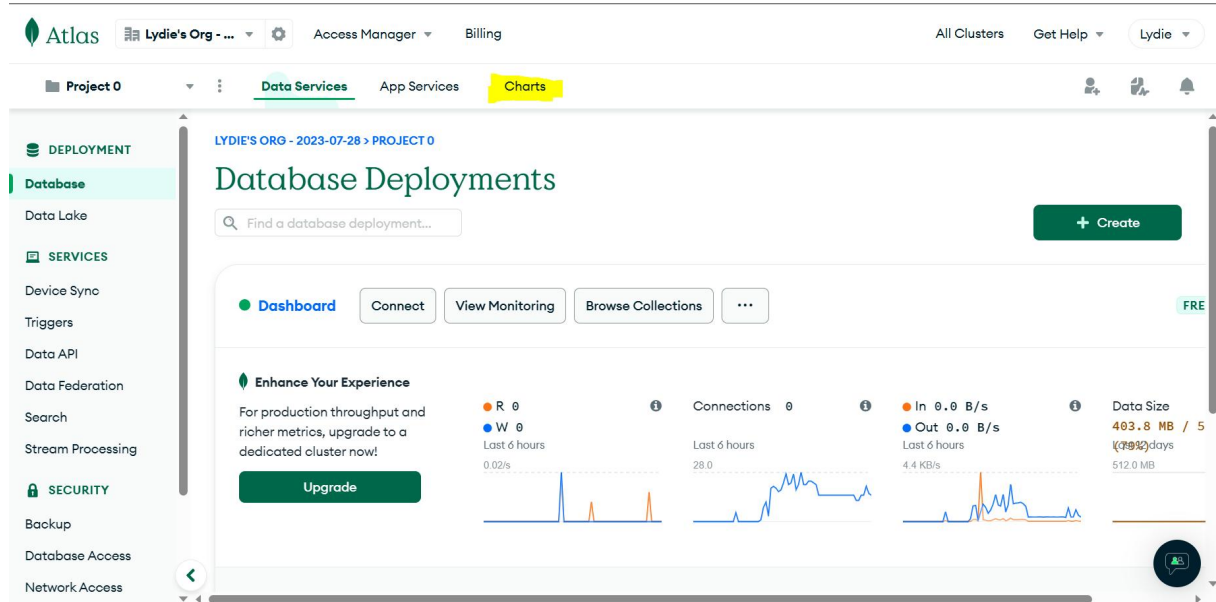
Connect

Nous nous connectons grâce au code de connexion avec nos identifiants. Dans ce cas, login : **lydiengar** et password : **ingc2023**.

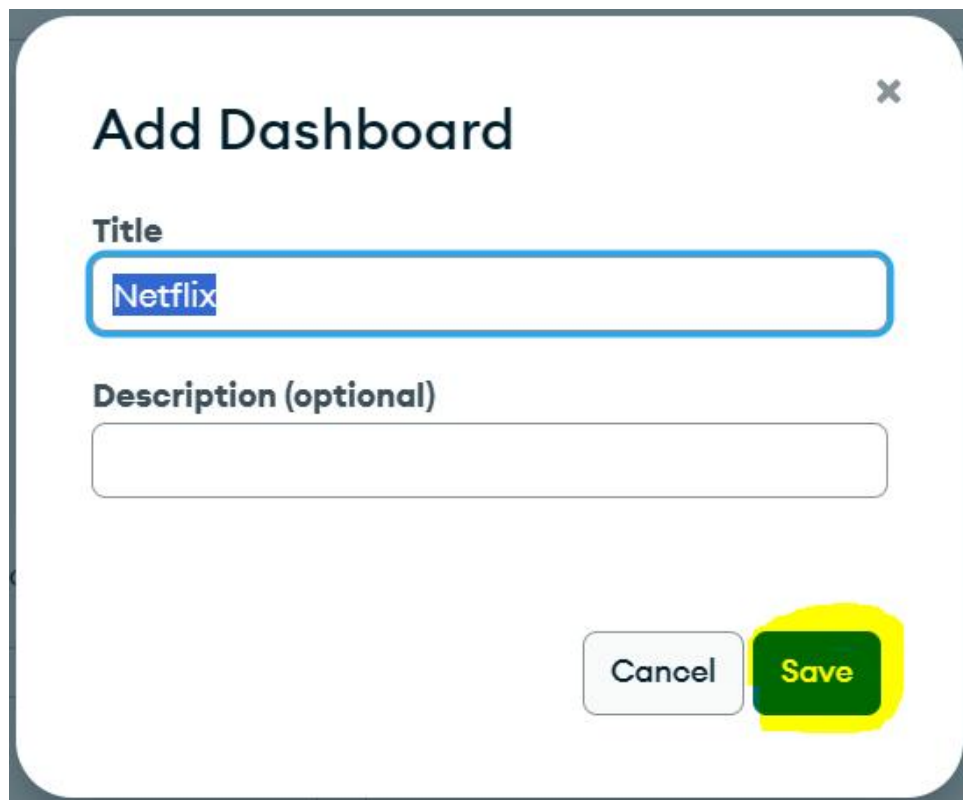


Nous pouvons voir toutes les bases de données que nous avons dans notre serveur. Celle que nous allons utiliser plutôt est NetflixData.

Nous allons désormais passer à la création des Charts (graphiques), qui vont constituer notre tableau de bord, en cliquant sur Charts.



Nous allons donc créer notre tableau de bord pour ce faire.



II. Dataset

1. Choix

Notre dataset a été téléchargé suivant le lien : [Netflix Userbase Dataset / Kaggle](#).

Ce dataset a été choisi à cause de sa facilité de compréhension, de son accessibilité. Il permet de faire des analyses grâce à sa large diversité de données, dans le but d'étudier le comportement des utilisateurs de Netflix dans les grandes régions du monde.

2. Description

Le jeu de données offre un échantillon de 2500 lignes de la base d'utilisateurs de Netflix, présentant divers aspects des abonnements des utilisateurs, des revenus et des détails de compte. Chaque ligne représente un utilisateur unique, identifié par son identifiant. Le jeu de données inclut des informations telles que le type d'abonnement de l'utilisateur, les revenus mensuels générés par leur abonnement, la date à laquelle ils ont rejoint Netflix, la date de leur dernier paiement et le pays dans lequel ils se trouvent.

Le dataset est présenté avec les 10 champs suivants :

- Age : L'âge de l'utilisateur.
- Country : Le pays d'où provient l'utilisateur.
- Device : Le type d'appareil que l'utilisateur utilise pour accéder à Netflix (par exemple, téléphone portable, tablette, ordinateur, téléviseur, etc.).
- Gender : Le genre de l'utilisateur (homme, femme).
- Join Date : La date à laquelle l'utilisateur s'est inscrit sur Netflix.
- Last Payment Date : La date de la dernière transaction de paiement associée à l'utilisateur, qui pourrait être liée à la facturation de l'abonnement Netflix.
- Monthly Revenue : Le montant mensuel de revenus générés à partir de cet utilisateur.
- Plan Duration : La durée du plan d'abonnement de l'utilisateur (par exemple ici, mensuel, annuel).
- Subscription Type : Le type d'abonnement de l'utilisateur (par exemple, Standard, Premium, Basic).
- User ID : Un identifiant unique attribué à chaque utilisateur dans le dataset.

Ces champs permettraient potentiellement d'analyser les tendances d'utilisation, les préférences d'abonnement, les habitudes de paiement et d'autres aspects du comportement des utilisateurs sur la plate-forme Netflix.

Ci-dessous, les 10 premières lignes de notre jeu de données.

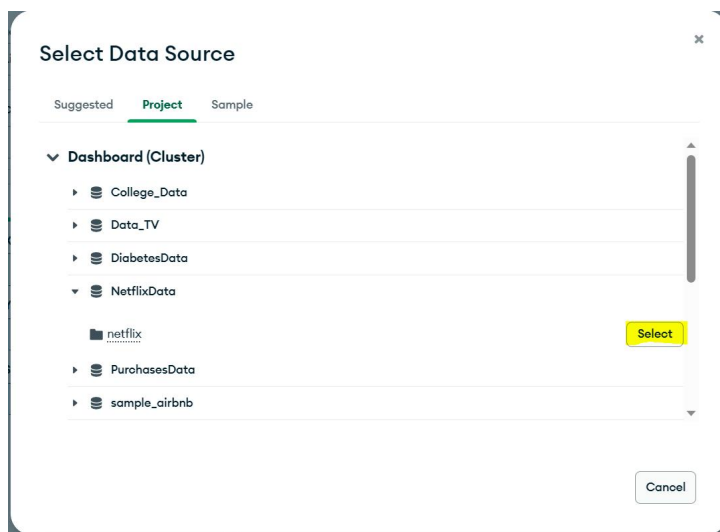
🔍 User ID	▲ Subscripti...	# Monthly R...	▲ Join Date	▲ Last Paym...
1	Basic	10	15-01-22	10-06-23
2	Premium	15	05-09-21	22-06-23
3	Standard	12	28-02-23	27-06-23
4	Standard	12	10-07-22	26-06-23
5	Basic	10	01-05-23	28-06-23
6	Premium	15	18-03-22	27-06-23
7	Standard	12	09-12-21	25-06-23
8	Basic	10	02-04-23	24-06-23
9	Standard	12	20-10-22	23-06-23
10	Premium	15	07-01-23	22-06-23

📍 Country	# Age	▲ Gender	▲ Device	▲ Plan Durat...
United States	28	Male	Smartphone	1 Month
Canada	35	Female	Tablet	1 Month
United Kingdom	42	Male	Smart TV	1 Month
Australia	51	Female	Laptop	1 Month
Germany	33	Male	Smartphone	1 Month
France	29	Female	Smart TV	1 Month
Brazil	46	Male	Tablet	1 Month
Mexico	39	Female	Laptop	1 Month
Spain	37	Male	Smartphone	1 Month
Italy	44	Female	Smart TV	1 Month

III. Mise en œuvre des graphiques

Nous allons a présent mettre en œuvre nos différents graphiques.

D'abord, nous allons créer un Chart en sélectionnant le dataset qu'on veut utiliser, ce qui est notre source de données. Nous choisissons comme source de données le jeu de données des utilisateurs Netflix :



1) Graphique 1 : *Nombre d'utilisateurs par Pays*

➤ Objectif :

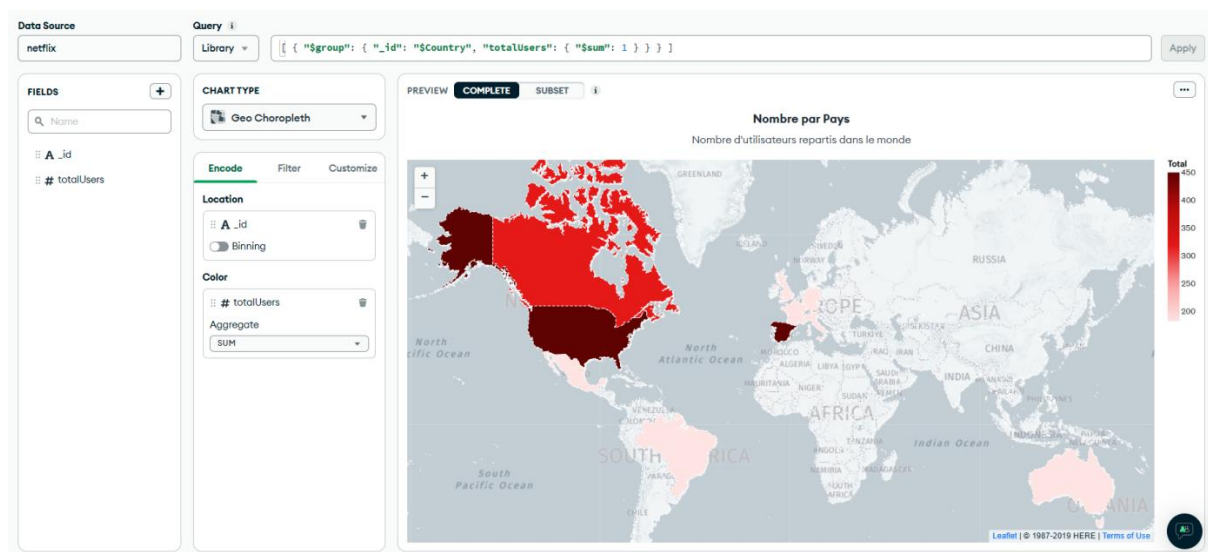
Cette requête regroupe les données par pays et utilise l'opérateur \$sum pour compter le nombre total d'utilisateurs pour chaque pays. Nous avons donc la répartition des utilisateurs abonnés par localité.

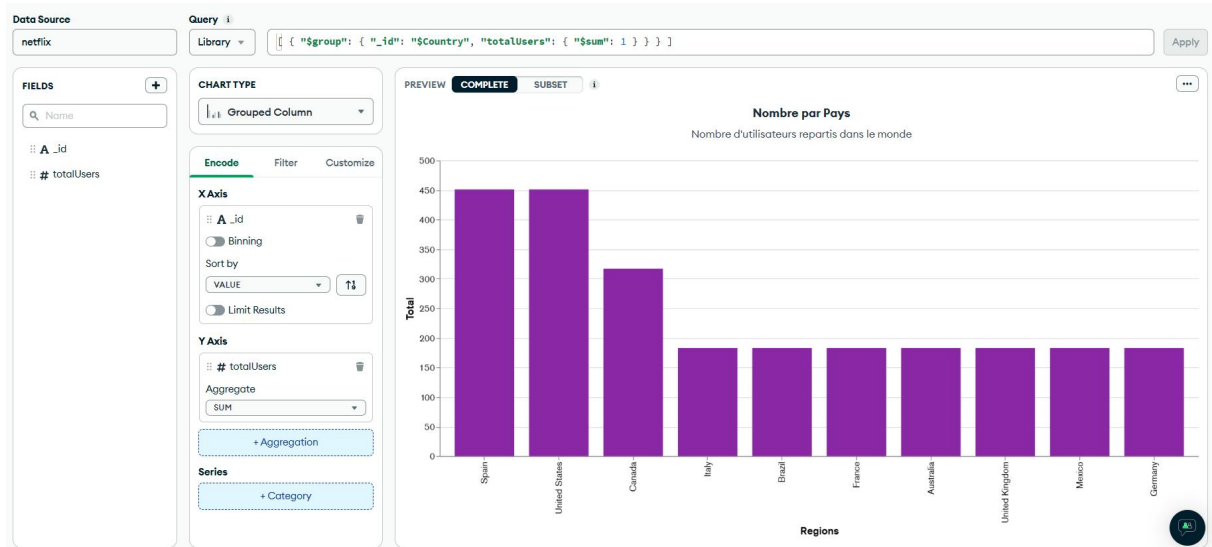
➤ Requête :

```
[
  {
    "$group": {
      "_id": "$Country",
      "totalUsers": { "$sum": 1 }
    }
  }
]
```

- "\$group": C'est une étape d'agrégation qui regroupe les documents en fonction des critères spécifiés.
- "_id": "\$Country": C'est l'expression qui définit ce selon quoi les documents seront regroupés. Dans ce cas, les documents sont regroupés en fonction de la valeur du champ "Country".
- "totalUsers": { "\$sum": 1 } : C'est une opération d'agrégation qui compte le nombre de documents (utilisateurs) dans chaque groupe. L'opérateur \$sum est utilisé avec la valeur 1 pour compter chaque document (car chaque document représente un utilisateur).

➤ Graphique :





2) Graphique 2 : *Nombre d'abonnements par dispositif de visionnage*

➤ Objectif :

Cette requête regroupe les données par outil de visionnage et utilise l'opérateur \$sum pour compter le nombre total d'utilisateurs pour chaque pays. Nous avons donc la répartition des utilisateurs abonnés par type d'appareil de visionnage.

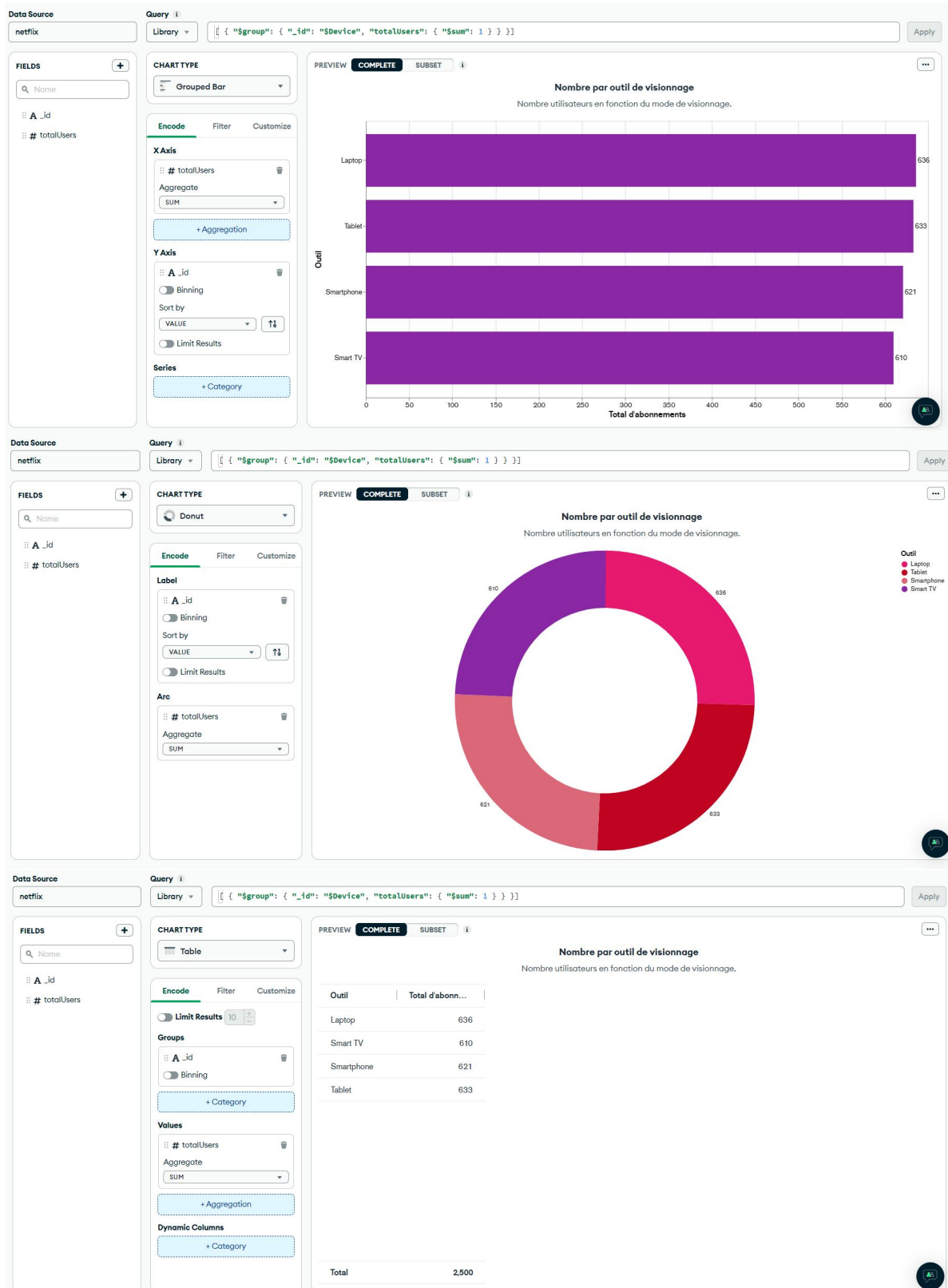
➤ Requête :

```
[
  {
    "$group": {
      "_id": "$Device",
      "totalUsers": { "$sum": 1 }
    }
  }
]
```

- \$group: Cette étape regroupe les documents de la collection en fonction de la valeur de l'attribut "Device" de chaque document. Dans cette étape, _id est utilisé pour spécifier l'attribut selon lequel les documents seront regroupés.

- "totalUsers": C'est un champ calculé qui utilise l'opérateur \$sum pour compter le nombre total de documents regroupés dans chaque groupe. Dans ce contexte, il est utilisé pour compter combien de fois chaque type de dispositif apparaît dans la collection.

➤ Graphique :



3) Graphique 3 : *Nombre d'abonnements par type d'abonnement*

➤ Objectif :

Ce graphe est utilisé pour avoir des informations sur la distribution des différents types d'abonnements parmi les utilisateurs.

➤ Requête :

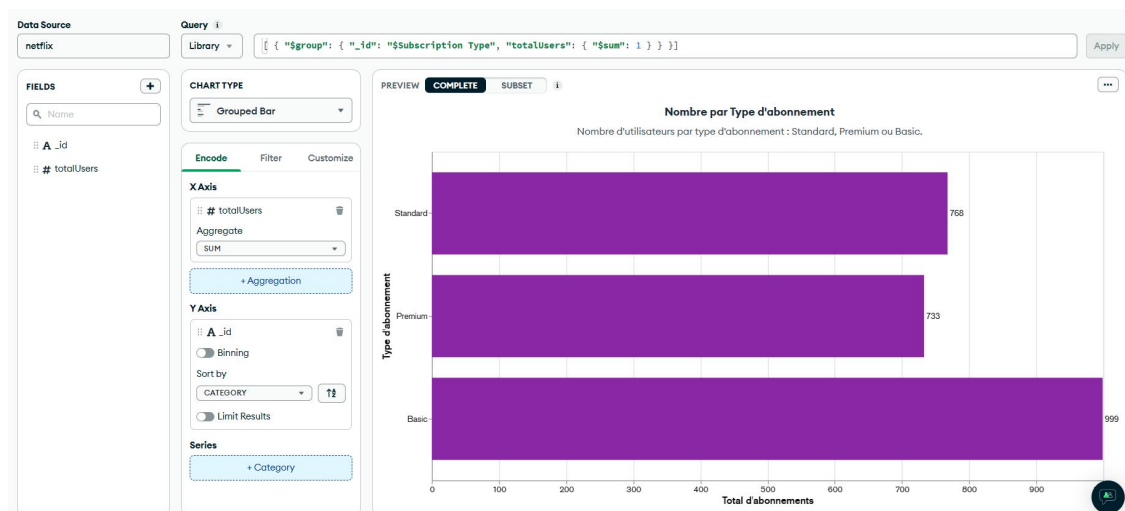
```
[
{
  "$group": {
    "_id": "$Subscription Type",
    "totalUsers": { "$sum": 1 }
  }
}
```

- \$group: Comme précédemment, cette étape regroupe les documents de la collection en fonction de la valeur de l'attribut "Subscription Type" de chaque document.

- _id: _id est défini sur "Subscription Type", ce qui signifie que les documents seront regroupés en fonction des différentes valeurs de l'attribut "Subscription Type".

- "totalUsers": C'est un champ calculé qui utilise l'opérateur \$sum pour compter le nombre d'utilisateurs ayant chaque type d'abonnement dans la collection.

➤ Graphique :



4) Graphique 4 : *Nombre d'abonnements par tranche d'âge*

➤ Objectif :

Ce graphe permet d'obtenir des statistiques sur la distribution des âges parmi les utilisateurs. Nous avons des tranches d'âge de 10 ans.

➤ Requête :

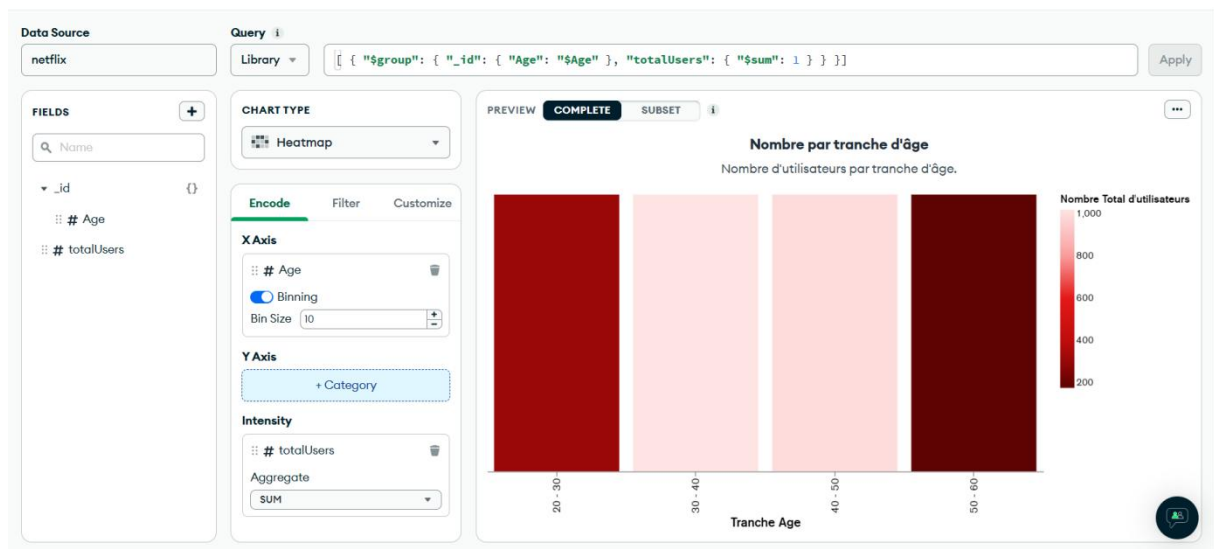
```
[
{
  "$group": {
    "_id": {
      "Age": "$Age"
    },
    "totalUsers": { "$sum": 1 }
  }
}
]
```

- \$group: Cette étape regroupe les documents de la collection qui seront regroupés en fonction des différentes valeurs de l'attribut "Age".

- _id: _id contient un seul champ "Age" qui est associé à la valeur de l'attribut "Age" de chaque document. Cela permet de regrouper les documents ayant la même valeur d'âge ensemble.

- "totalUsers": Ce champ permet de compter combien de fois chaque âge apparaît dans la collection.

➤ Graphique :



5) Graphique 5 : *Nombre d'abonnements par outil de visionnage et par genre*

➤ Objectif :

Ce graphe permet d'avoir la répartition des utilisateurs en fonction de l'outil utilisé pour le visionnage mais également avec une séparation de genre.

➤ Requête :

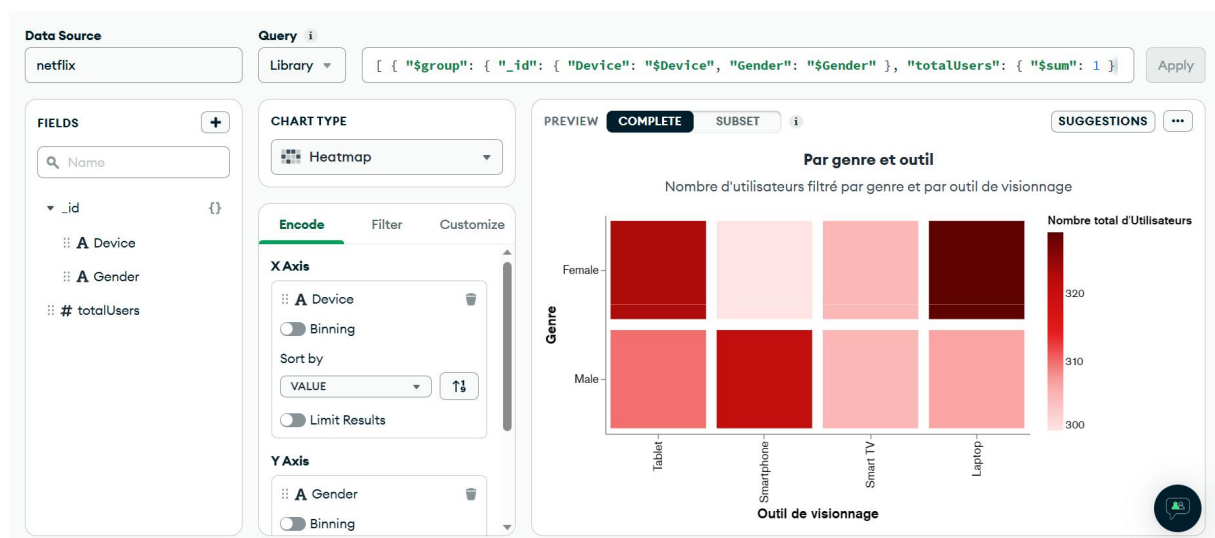
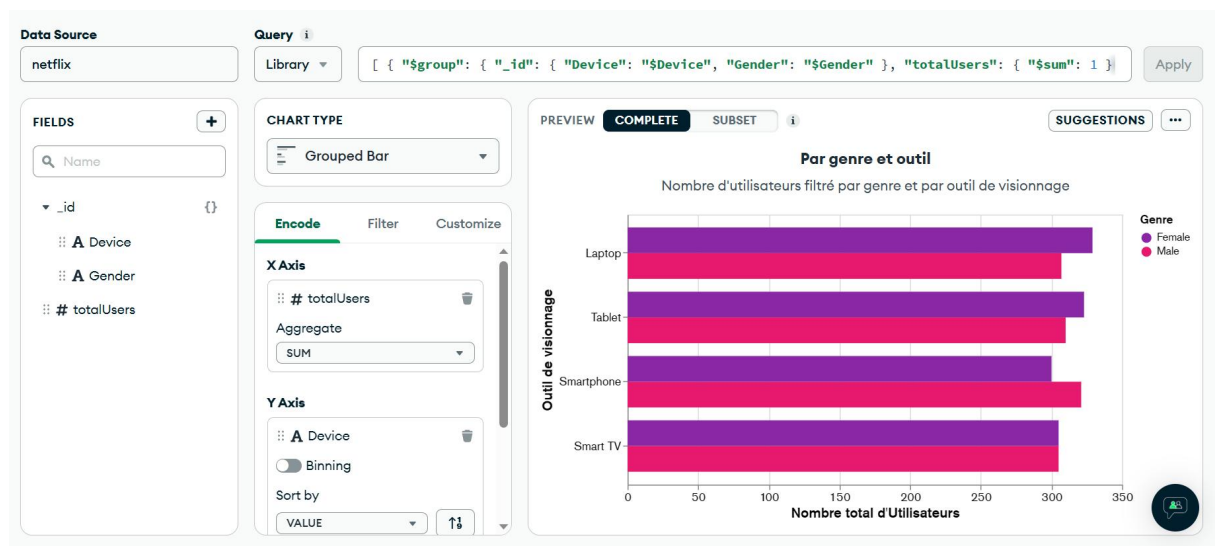
```
[
{
  "$group": {
    "_id": {
      "Device": "$Device",
      "Gender": "$Gender"
    },
    "totalUsers": { "$sum": 1 }
  }
}
```

- \$group: Cette étape regroupe les documents qui sont regroupés en fonction de deux attributs : "Device" et "Gender".

- _id: La clé _id contient deux champs : "Device" et "Gender". Ces champs correspondent aux valeurs des attributs respectifs de chaque document. Cela signifie que les documents seront regroupés en fonction des différentes combinaisons de outils et du genre.

- "totalUsers": Ce champ permet de compter combien de fois chaque combinaison de dispositif et de genre apparaît dans la collection.

➤ Graphique :



6) Graphique 6 : ***Nombre d'abonnements par type d'abonnement et par tranche d'âge***

➤ Objectif :

Ce graphe permet d'avoir la distribution des utilisateurs par type d'abonnement et par tranche d'âge de 10 ans.

➤ Requête :

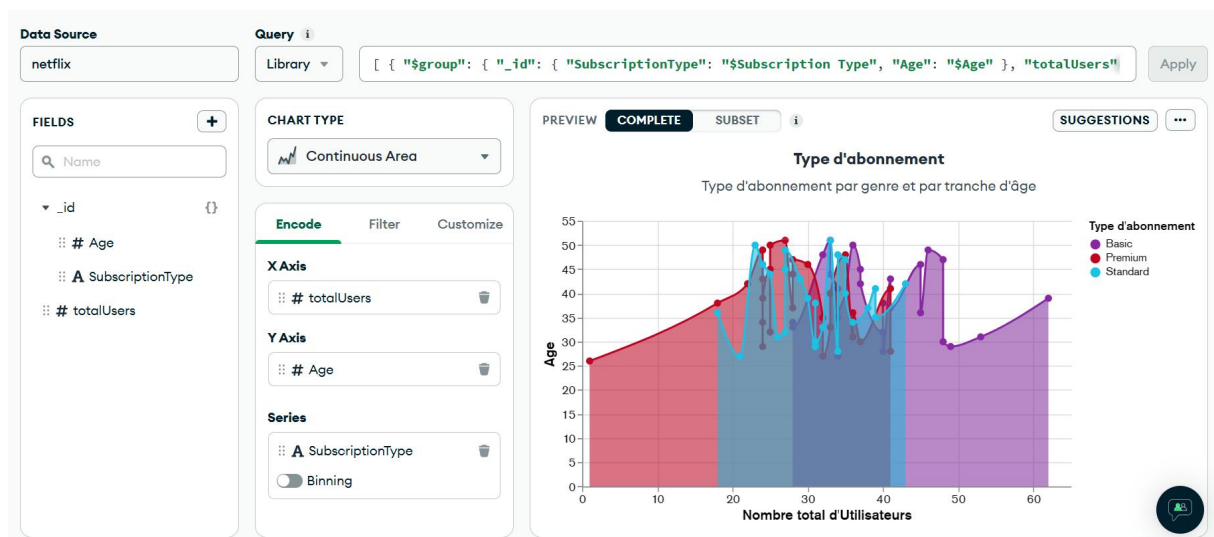
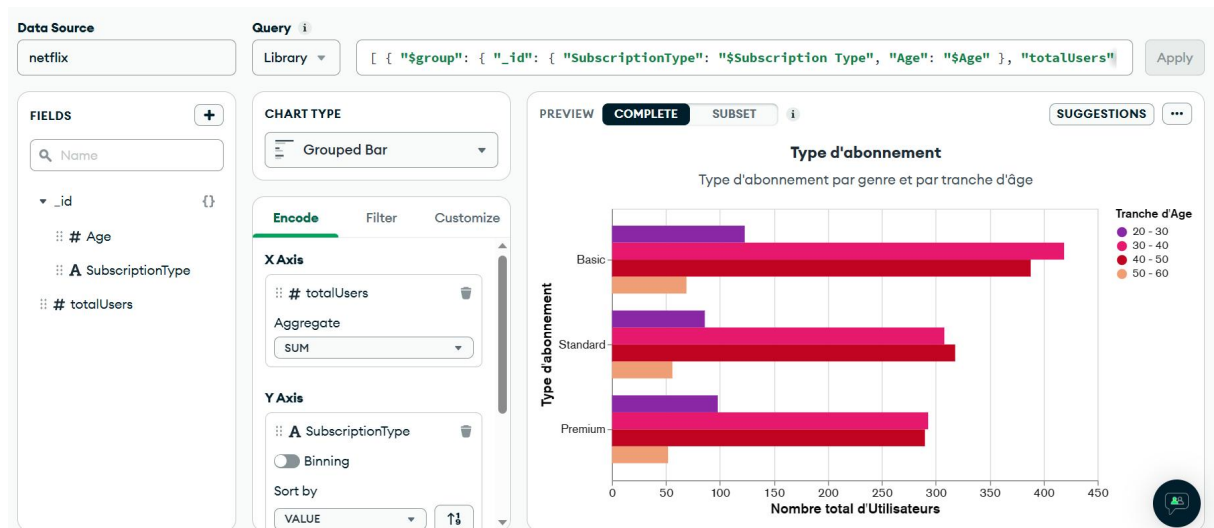
```
[
{
  "$group": {
    "_id": {
      "SubscriptionType": "$Subscription Type",
      "Age": "$Age"
    },
    "totalUsers": { "$sum": 1 }
  }
}
```

- \$group: Cette étape regroupe les documents qui sont regroupés en fonction de deux attributs : "Subscription Type" et "Age".

- _id: La clé _id contient deux champs : "SubscriptionType" et "Age". Ces champs correspondent aux valeurs des attributs respectifs de chaque document. Cela signifie que les documents seront regroupés en fonction des différentes combinaisons de types d'abonnement et d'âges.

- "totalUsers": Ce champ permet de compter combien de fois chaque combinaison de type d'abonnement et d'âge apparaît dans la collection.

➤ Graphique :



7) Graphique 7 : *Nombre d'abonnements par revenu et par date de dernier paiement*

➤ Objectif :

Il s'agit de compter le nombre total d'utilisateurs pour chaque combinaison de revenu mensuel et de date de dernier paiement, ce qui pourrait être utilisé pour analyser les revenus générés par différents groupes d'utilisateurs en fonction de leurs habitudes de paiement.

➤ Requête :

```
[
{
  "$group": {
    "_id": {
```

```

    "Revenue": "$Monthly Revenue",

    "LastPayment": "$Last Payment Date"

  },

  "totalUsers": { "$sum": 1 }

}

}

]

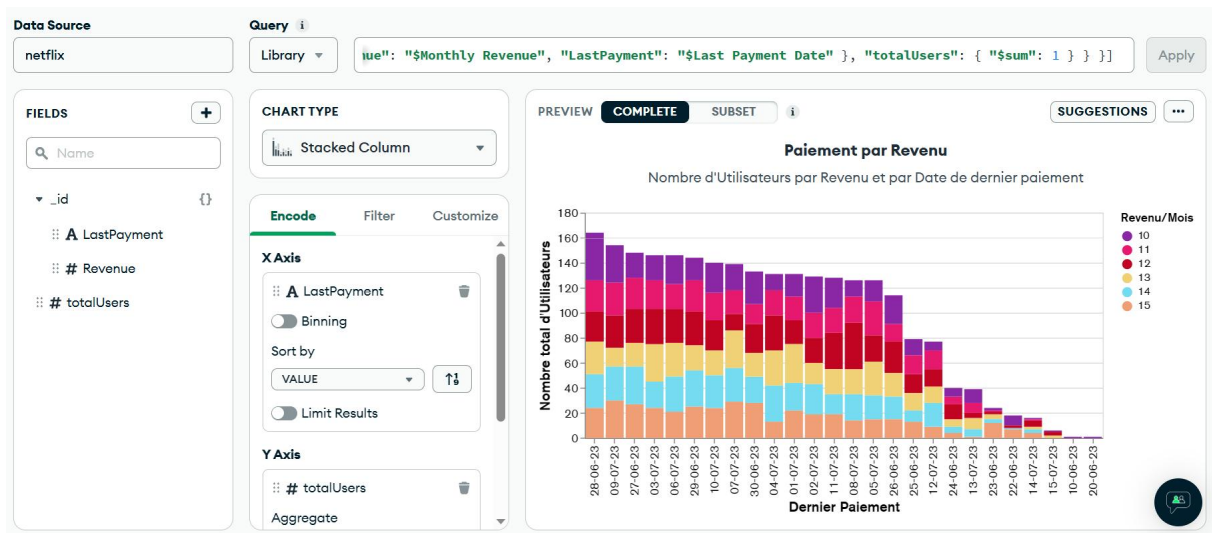
```

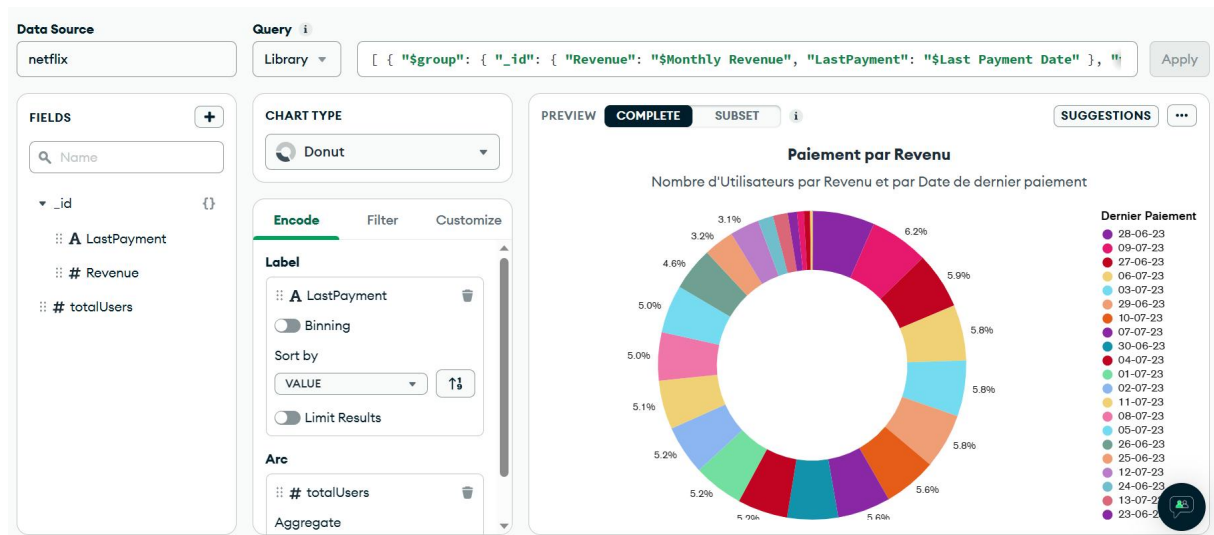
- \$group: Cette étape regroupe les documents de la collection en fonction de deux attributs : "Monthly Revenue" et "Last Payment Date".

- _id: La clé _id contient deux champs : "Revenue" et "LastPayment" qui correspondent aux valeurs des attributs respectifs de chaque document. Cela signifie que les documents seront regroupés en fonction des différentes combinaisons de revenus mensuels et de dates de dernier paiement.

- "totalUsers": Ce champ permet de compter combien de fois chaque combinaison de revenu mensuel et de date de dernier paiement apparaît dans la collection.

➤ Graphique :





8) Graphique 8 : *Nombre d'abonnements par pays et par tranche d'age*

➤ Objectif :

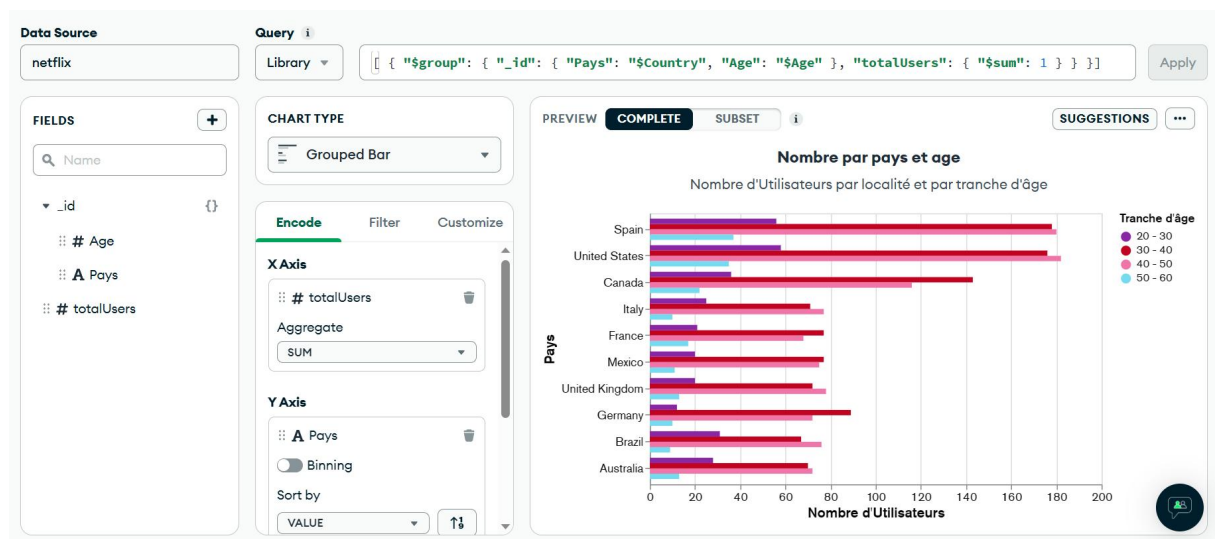
L'objectif est de compter le nombre total d'utilisateurs pour chaque combinaison de pays et d'âge, ce qui pourrait être utilisé pour analyser la distribution des utilisateurs par pays et par groupe d'âge.

➤ Requête :

```
[
{
  "$group": {
    "_id": {
      "Pays": "$Country",
      "Age": "$Age"
    },
    "totalUsers": { "$sum": 1 }
  }
}
```

- `$group`: Cette étape regroupe les documents de la collection en fonction de deux attributs : "Country" et "Age".
- `_id`: La clé `_id` contient deux champs : "Country" et "Age". Ces champs correspondent aux valeurs des attributs respectifs de chaque document. Cela signifie que les documents seront regroupés en fonction des différentes combinaisons de pays et d'âges.
- "totalUsers": Ce champ permet de compter combien de fois chaque combinaison de pays et d'âge apparaît dans la collection.

➤ Graphique :



9) Graphique 9 : **Nombre d'abonnements par type d'abonnement et par pays**

➤ Objectif :

Il consiste à compter le nombre total d'utilisateurs pour chaque combinaison de type d'abonnement et d'âge, ce qui pourrait être utilisé pour obtenir des informations sur la répartition des utilisateurs en fonction de ces deux attributs.

➤ Requête :

```
[
{
  "$group": {
    "_id": {
      "SubscriptionType": "$Subscription Type",
```

"Pays": "\$Country"

},

"totalUsers": { "\$sum": 1 }

}

}

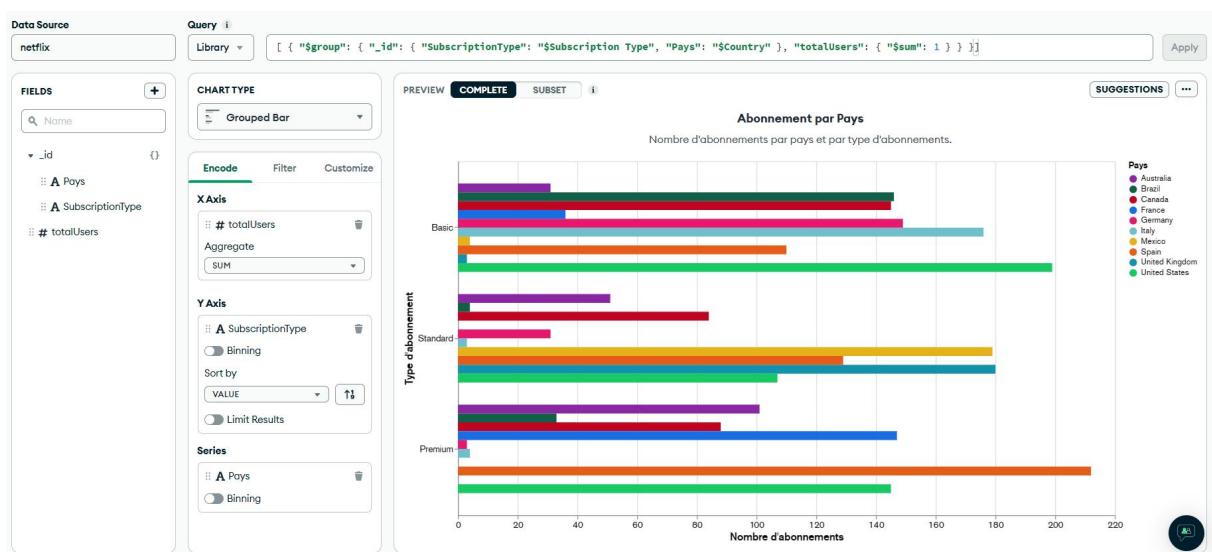
]

- \$group: Cette étape regroupe les documents de la collection. Les documents sont regroupés en fonction de deux attributs : "Subscription Type" et "Age".

- _id: La clé _id contient les deux champs : "SubscriptionType" et "Age". Ces champs correspondent aux valeurs des attributs respectifs de chaque document. Cela signifie que les documents seront regroupés en fonction des différentes combinaisons de types d'abonnement et d'âges.

- "totalUsers": Ce champ permet de compter combien de fois chaque combinaison de type d'abonnement et d'âge apparaît dans la collection.

➤ Graphique :



10) Graphique 10 : **Nombre total d'abonnees**

➤ Objectif :

Le graphe a pour objectif d'afficher le nombre total de documents contenus dans notre dataset de travail.

➤ Requête :

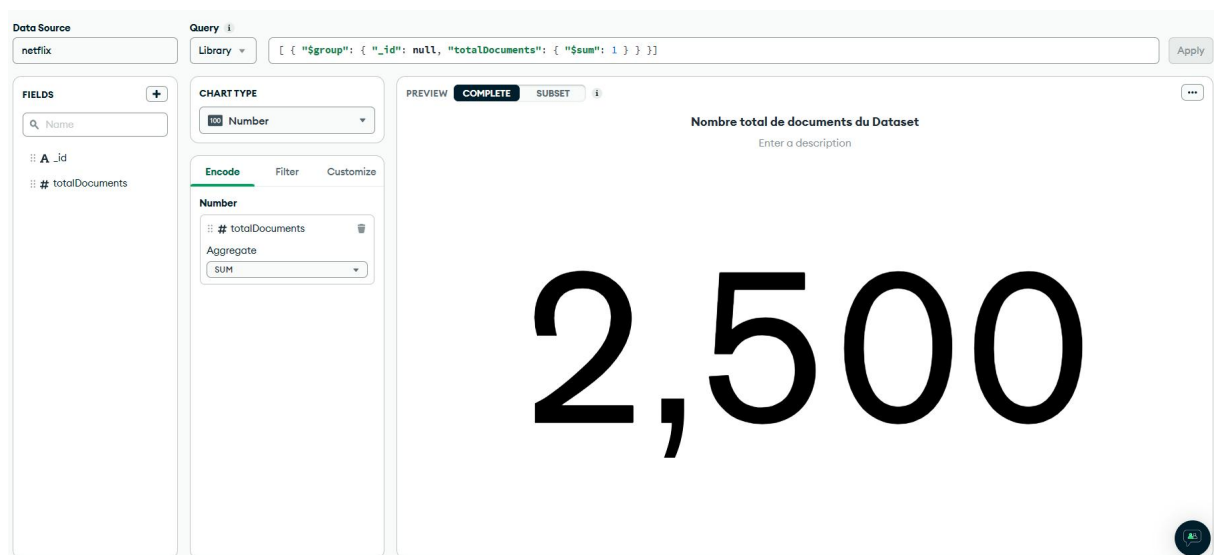
```
[
{
  "$group": {
    "_id": null,
    "totalDocuments": { "$sum": 1 }
  }
}
]
```

- \$group: Cette étape regroupe tous les documents de la collection en un seul groupe, car nous utilisons _id comme null.

- _id: Comme mentionné, il est défini sur null pour regrouper tous les documents ensemble.

- "totalDocuments": Ce champ utilise l'opérateur \$sum pour compter le nombre total de documents regroupés.

➤ Graphique :



IV. Présentation finale du Dashboard

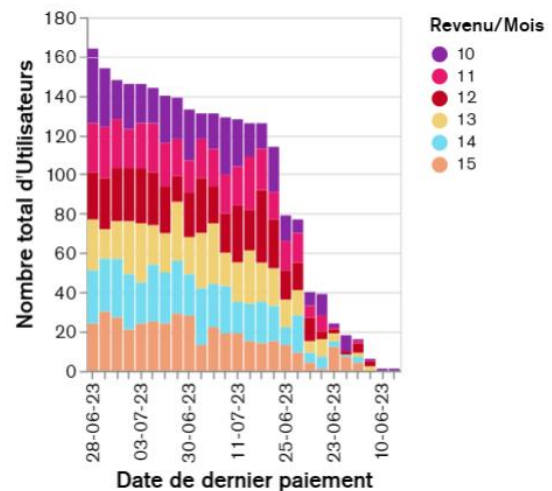
Ci-dessous est le tableau de bord réalisé, avec les différents graphiques. Notre tableau de bord comporte 10 graphiques qui permettraient d'analyser l'ensemble du jeu de données Netflix suivant les divers champs.

Nombre total de documents du Dataset

2,500

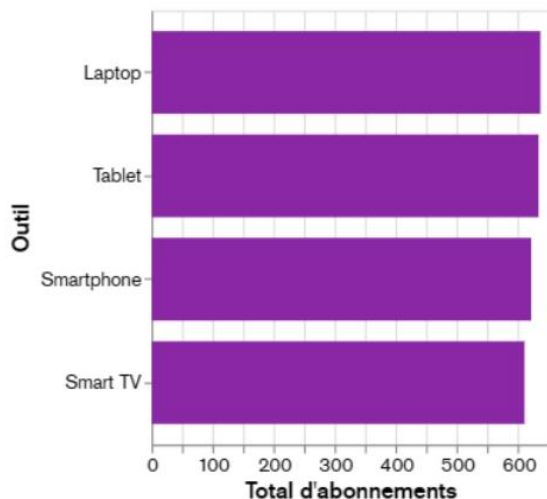
Paielement par Revenu

Nombre par Revenu et par Date de dernier paiement



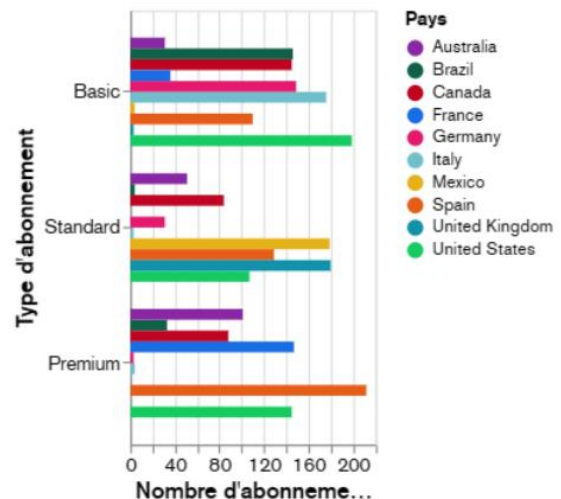
Nombre par outil de visionnage

Nombre utilisateurs en fonction du mode de visionna...



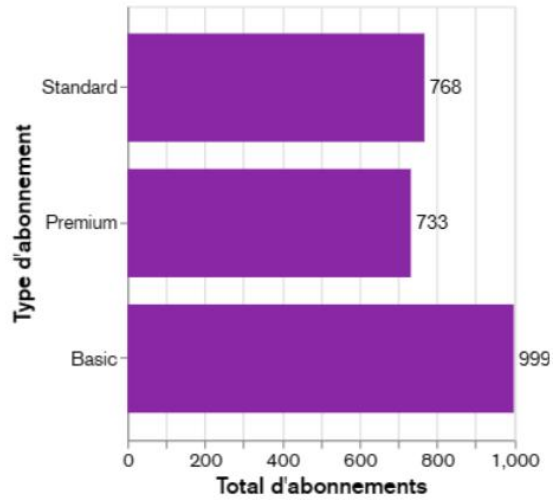
Abonnement par Pays

Nombre d'abonnements par pays et par type d'abonn...



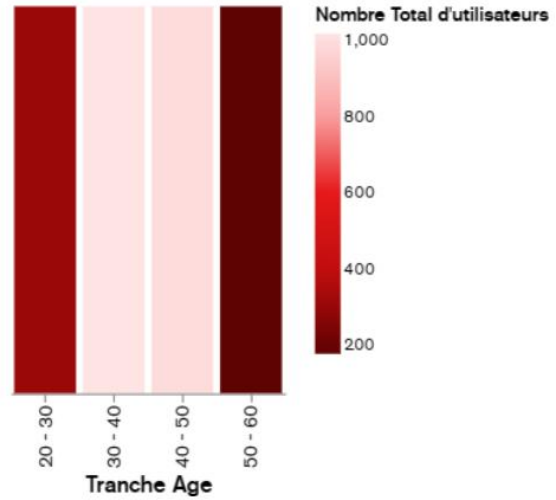
Nombre par Type d'abonnement

Nombre d'utilisateurs par type d'abonnement : Stand...



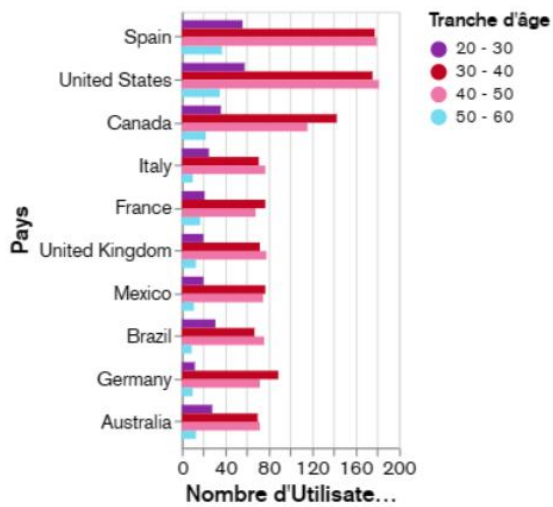
Nombre par tranche d'âge

Nombre d'utilisateurs par tranche d'âge.



Nombre par pays et age

Nombre d'Utilisateurs par localité et par tranche d'âge



Nombre par Pays

Nombre d'utilisateurs repartis dans le monde

