

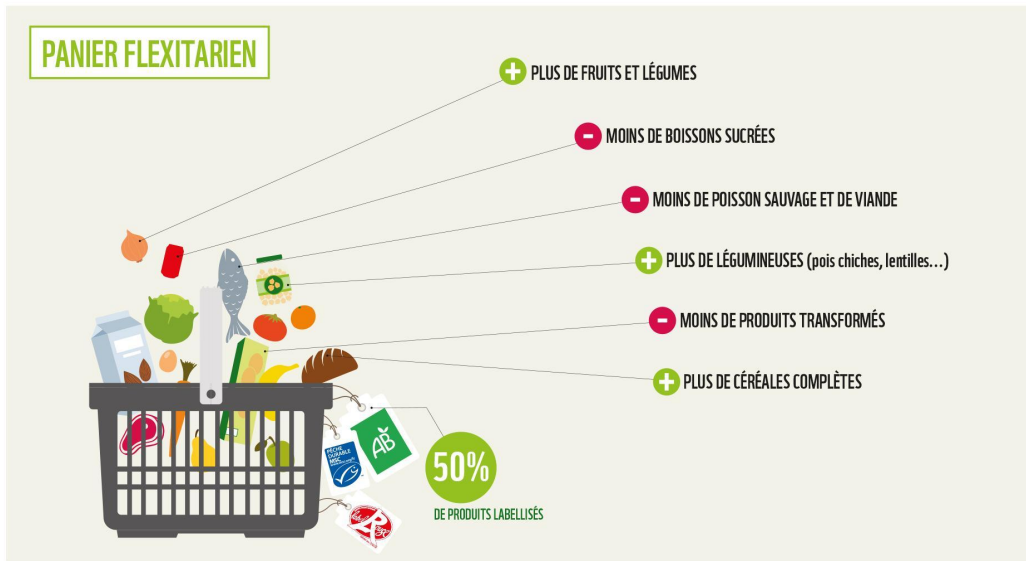
FoodFlix

A solid blue horizontal line positioned directly beneath the word "FoodFlix".

À quoi ça sert??

FoodFlix est une application qui va permettre aux utilisateurs d'avoir des informations concernant le produits alimentaire.

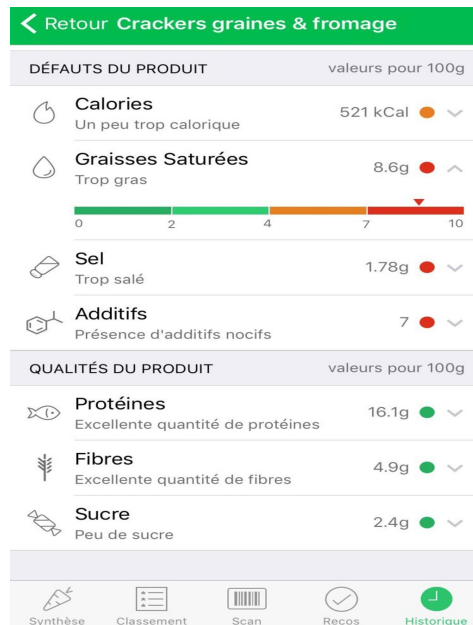
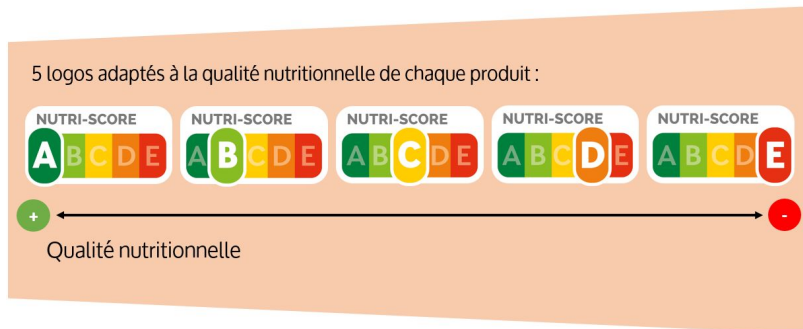
Grâce à une base de données récupérer des produits, nous allons choisir les information que nous allons transmettre au utilisateur afin qu'il puissent choisir ce qu'il souhaite!



À quoi ça ressemble?

Il y a différente façon de montrer ses informations aux utilisateurs tout dépend des informations que vous voulez leur transmettre.

Ex :



La base de donnée

On récupère la base de données sur kaggle, afin de pouvoir avoir d'avoir des information sur les produits alimentaires, qui pourront nous permettent d'effectuer des nutri score grâce aux informations récupérées dans cette base de donnée .

[Open Food Facts](#)



▼

Data Explorer
963.46 MB
 [en.openfoodfacts.org.produ...](#)

< **en.openfoodfacts.org.products.tsv** (963.46 MB)  

Detail Compact Column

10 of 163 columns ▼

À quoi ressemble cette base de données

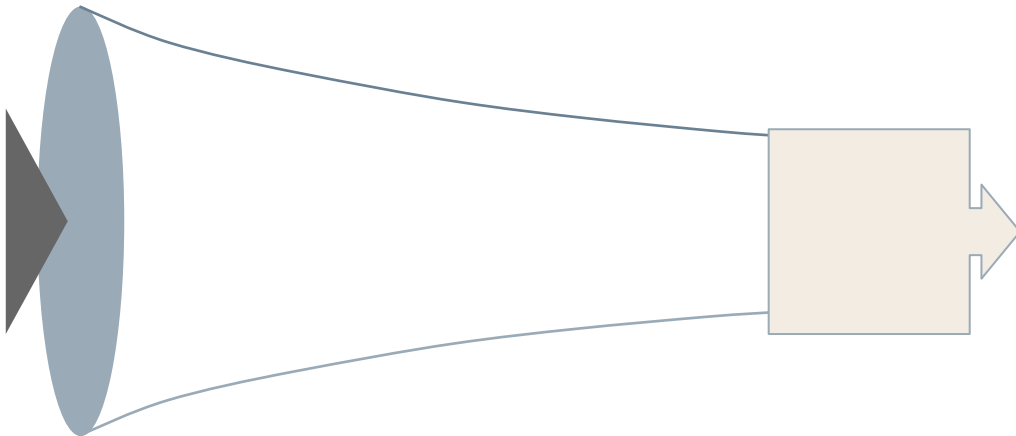
| | code | url | creator | created_t | created_datetime | last_modified_t | last_modified_datetime | product_name | generic_name | quantity | ... |
|-----|-------|---|----------------------------|------------|----------------------|-----------------|------------------------|--------------------------------|--------------|----------|-----|
| 0 | 3087 | http://world-en.openfoodfacts.org/product/0000... | openfoodfacts-contributors | 1474103866 | 2016-09-17T09:17:46Z | 1474103893 | 2016-09-17T09:18:13Z | Farine de blé noir | NaN | 1kg | ... |
| 1 | 4530 | http://world-en.openfoodfacts.org/product/0000... | usda-ndb-import | 1489069957 | 2017-03-09T14:32:37Z | 1489069957 | 2017-03-09T14:32:37Z | Banana Chips Sweetened (Whole) | NaN | NaN | ... |
| 2 | 4559 | http://world-en.openfoodfacts.org/product/0000... | usda-ndb-import | 1489069957 | 2017-03-09T14:32:37Z | 1489069957 | 2017-03-09T14:32:37Z | Peanuts | NaN | NaN | ... |
| 3 | 16087 | http://world-en.openfoodfacts.org/product/0000... | usda-ndb-import | 1489055731 | 2017-03-09T10:35:31Z | 1489055731 | 2017-03-09T10:35:31Z | Organic Salted Nut Mix | NaN | NaN | ... |
| 4 | 16094 | http://world-en.openfoodfacts.org/product/0000... | usda-ndb-import | 1489055653 | 2017-03-09T10:34:13Z | 1489055653 | 2017-03-09T10:34:13Z | Organic Polenta | NaN | NaN | ... |
| ... | ... | ... | ... | ... | ... | ... | ... | ... | ... | ... | ... |

Pour le moments la base de données n'est pas utilisable 163 colonnes et 356027 lignes

Stratégie de nettoyage de la base de données



| Caractéristiques | Valeur |
|--------------------|-----------|
| Nombre de lignes | 1 096 564 |
| Nombre de colonnes | 178 |



1ère étape

- suppression des colonnes null
- Filtrage sur 15 colonnes
- Suppression des doublons

2ème étape

- Sélection des aliments français
- Suppression des valeurs aberrantes et des valeurs manquantes

3ème étape

- Suppression des lignes

1ere étape : Nettoyage de la base de données

Pour commencer mon nettoyage, je supprime toutes les colonnes qui ne sont pas renseignées à 100%.

Après avoir récupéré les colonnes qui ne sont pas de tout renseignées. Je l'ai supprimé et je passe donc à 136 colonnes .

J'en profite aussi pour supprimer les métadonnées nous passons donc à 129 colonnes. Bien évidemment pas suffisant, mais cela permet déjà de faire un premier tri.

```
data.shape
```

```
(356027, 129)
```

1ere étape : Nettoyage de la base de données

Je décide donc de garder 15 colonnes afin d'avoir des informations nécessaire pour le nutri score. Nous passons donc à 356027 lignes.

```
data.head()
```

| | countries | categories | product_name | energy_100g | sugars_100g | fiber_100g | proteins_100g | sodium_100g | saturated-fat_100g | nutrition-score-fr_100g | nutrition_grade_fr | fat_100g | additives_n | additives | additives_tags |
|---|-----------|------------|--------------------------------|-------------|-------------|------------|---------------|-------------|--------------------|-------------------------|--------------------|----------|-------------|---|----------------|
| 0 | en:FR | NaN | Farine de blé noir | NaN | NaN | NaN | NaN | NaN | NaN | NaN | NaN | NaN | NaN | NaN | NaN |
| 1 | US | NaN | Banana Chips Sweetened (Whole) | 2243.0 | 14.29 | 3.6 | 3.57 | 0.000 | 28.57 | 14.0 | d | 28.57 | 0.0 | [bananas -> en:bananas] [vegetable-oil -... | NaN |
| 2 | US | NaN | Peanuts | 1941.0 | 17.86 | 7.1 | 17.86 | 0.250 | 0.00 | 0.0 | b | 17.86 | 0.0 | [peanuts -> en:peanuts] [wheat-flour -> ... | NaN |
| 3 | US | NaN | Organic Salted Nut Mix | 2540.0 | 3.57 | 7.1 | 17.86 | 0.482 | 5.36 | 12.0 | d | 57.14 | 0.0 | [organic-hazelnuts -> en:organic-hazelnuts] [organic-polenta -> en:organic-polenta] [... | NaN |
| 4 | US | NaN | Organic Polenta | 1552.0 | NaN | 5.7 | 8.57 | NaN | NaN | NaN | NaN | 1.43 | 0.0 | [organic-polenta -> en:organic-polenta] [... | NaN |

1ere étape : Nettoyage de la base de données

Je vérifie les doublons sur la colonnes product_name je remarque qu'il y a 106781 doublons. Je supprimes donc à 249246 sur 356027 initialement.

J'en profite aussi pour supprimer les lignes ou product name n'est pas renseignée

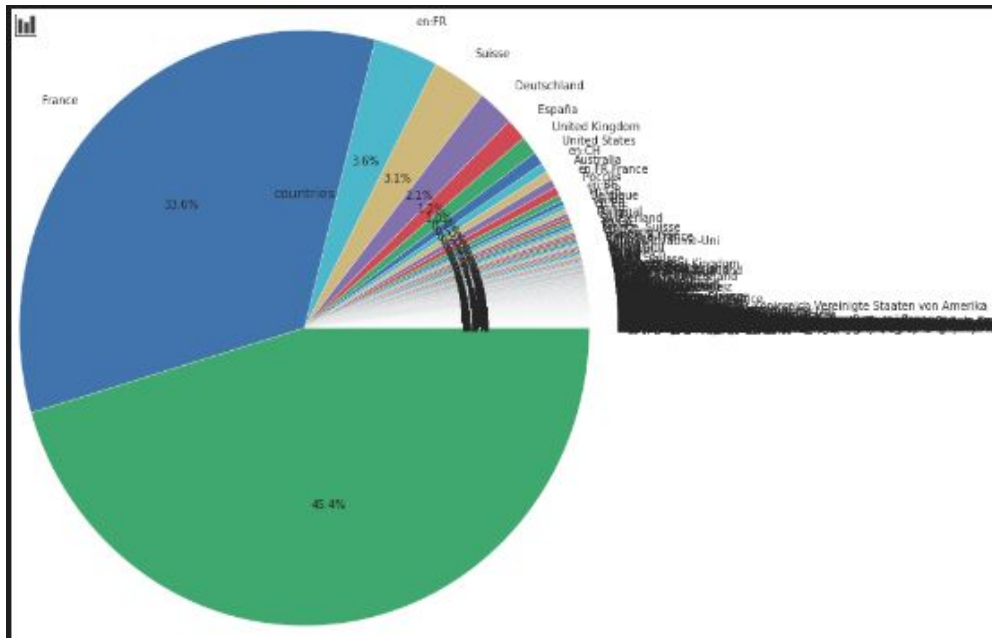
```
print(data.duplicated(['product_name']).value_counts())  
False      249246  
True       106781  
dtype: int64
```

Etape numéro 1 est terminée.

2ème étape : Nettoyage de la base de données

Voici la données dans la colonnes countries.

Nous allons trier tout cela et récupérer uniquement le pays france, avant cela nous allons regrouper tout les fr dans France.

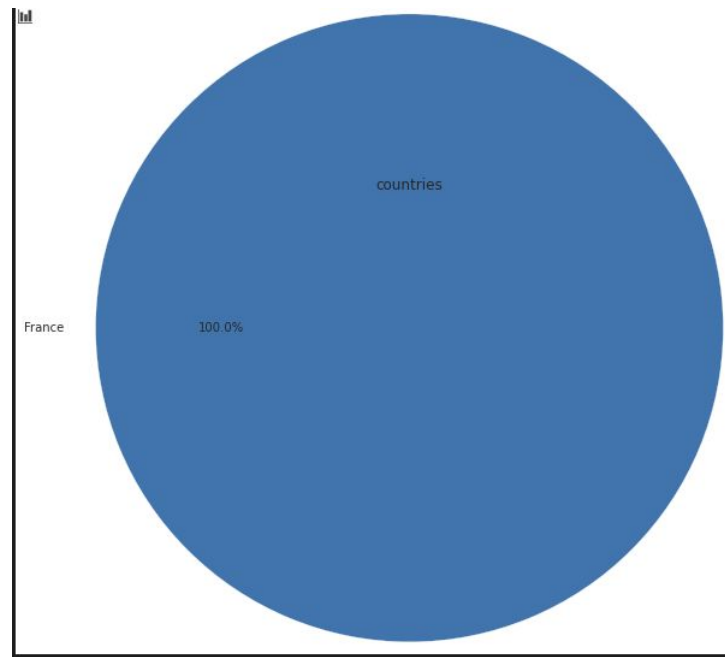


2ème étape : Nettoyage de la base de données

Voici le résultat après avoir trier countries et garder uniquement les valeurs de France.

Il nous reste “plus” que 98144 lignes

```
data.shape  
  
(98144, 15)
```



2ème étape : Nettoyage de la base de données

Pour trouver calculer le nutri score il nous faut au moins une de ses valeurs qui sont regroupés dans les colonnes `'fat_100g'`, `'energy_100g'`, `'sugars_100g'`, `'fiber_100g'`, `'proteins_100g'`, `'sodium_100g'`, `'saturated-fat_100g'`

Si aucunes d'entre elles n'est renseigné je supprime la ligne.

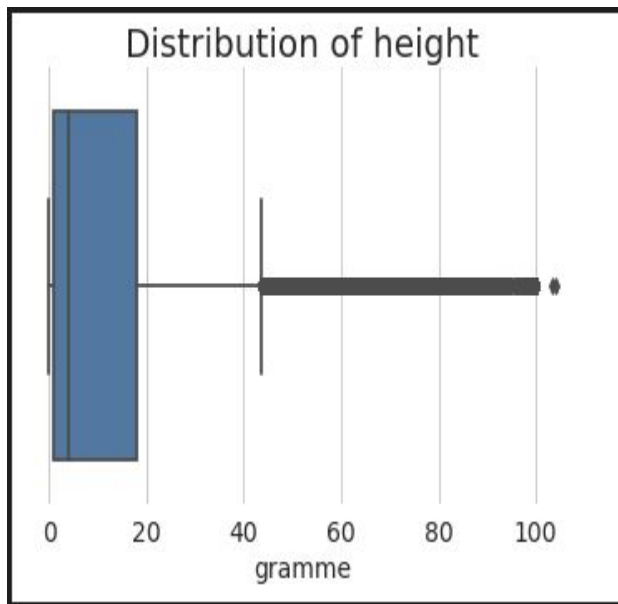
Après cela il nous reste 77526 lignes

2ème étape : Nettoyage de la base de données

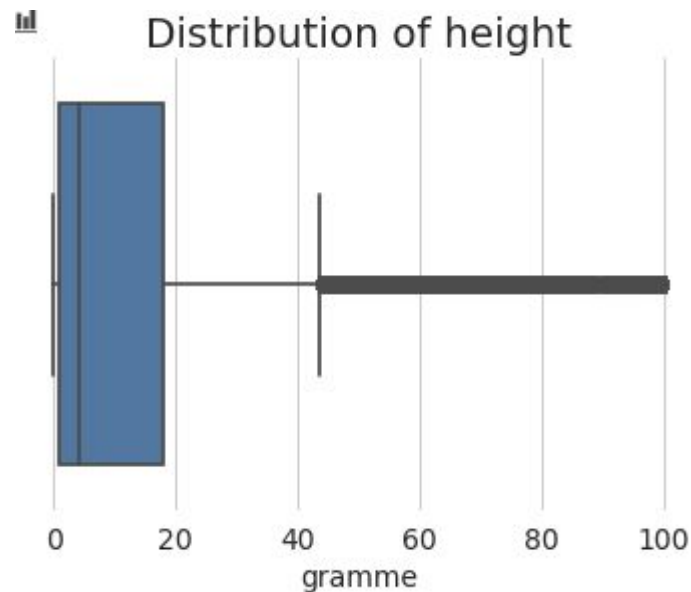
Maintenant nous allons supprimer les valeurs aberrante des colonnes.

Exemple :

Avant :

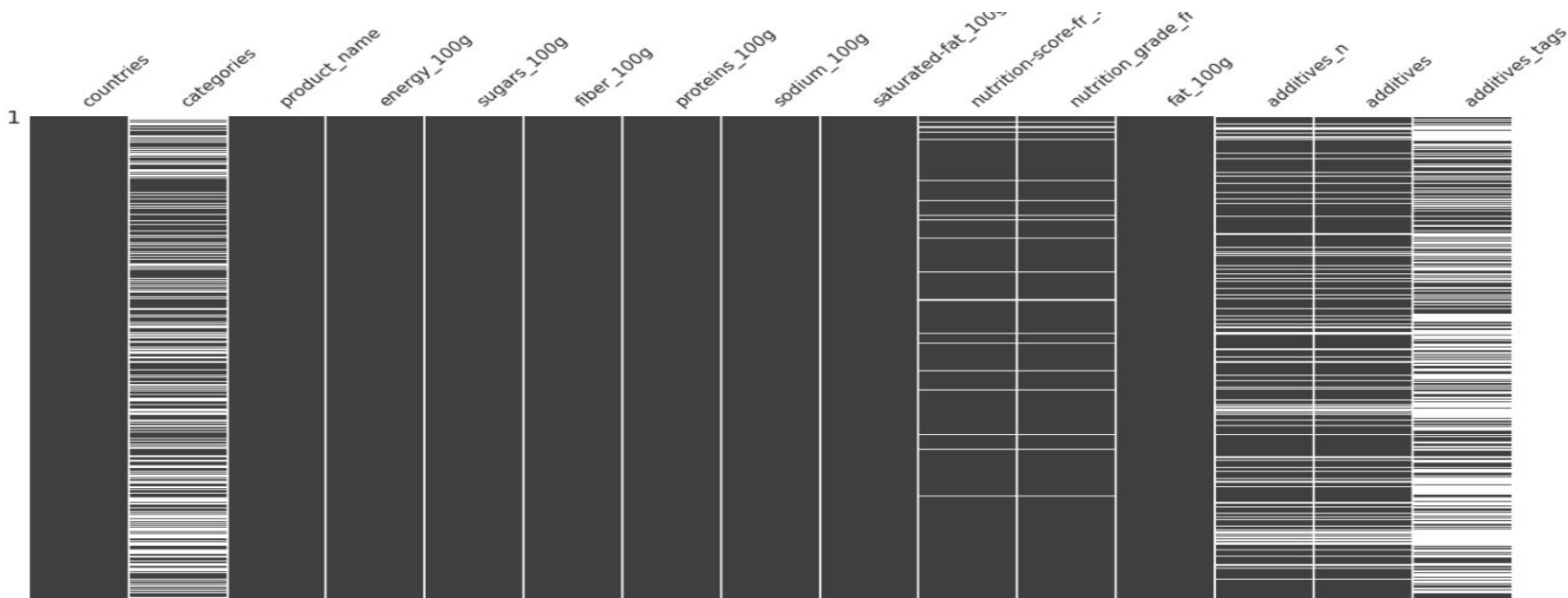


Après :



2ème étape : Nettoyage de la base de données

Concernant les données manquantes sur les colonnes qui permettent le calcul du nutri score, je décide de remplacer les NaN par 0.



3ème étape : Nettoyage de la base de données

Concernant les nutri score non renseignés manque de temps, je décide de supprimer les lignes concerné nous passons donc maintenant 74344 ligne et 15 colonnes

Il n'y a donc plus de nutri score non renseigné.



Bilan

Sur une base de donnée qui contenait 1096564 lignes et 178 colonnes.

Grâce au data cleaning et à la stratégie mis en place nous passons à 74344 lignes et 15 colonnes.

Et toute les lignes sont renseigné par le nom du produit et le nutri score mais contient aussi au moins une indication sur 100g.

