📕

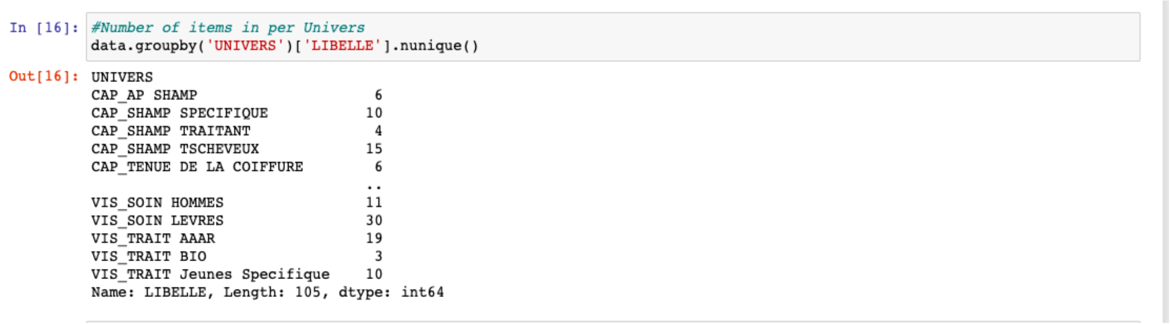
DAT: Exploration and Data visualisation

the main goal of this first part of the project is to explore the data base and add useful description to the dataset.

💡 **The data shape:**
The dataset has exactly 7 245 522 lines and 8 columns, the columns are: 1.TICKET_ID: ticket Id
2.MOIS_VENTE: month of sale
3.PRIX_NET: net price
4.FAMILLE: product family
5.UNIVERS: product universe
6.MAILLE: mesh of product
7.LIBELLE: product wording
8.CLI_ID: client id

💡 About the columns:
The dataset contains about 853 514 different client and 1 484 different products. These products bellow to 34 maille,105 universe and 9 family.
These information has been collected for a year (12 month)

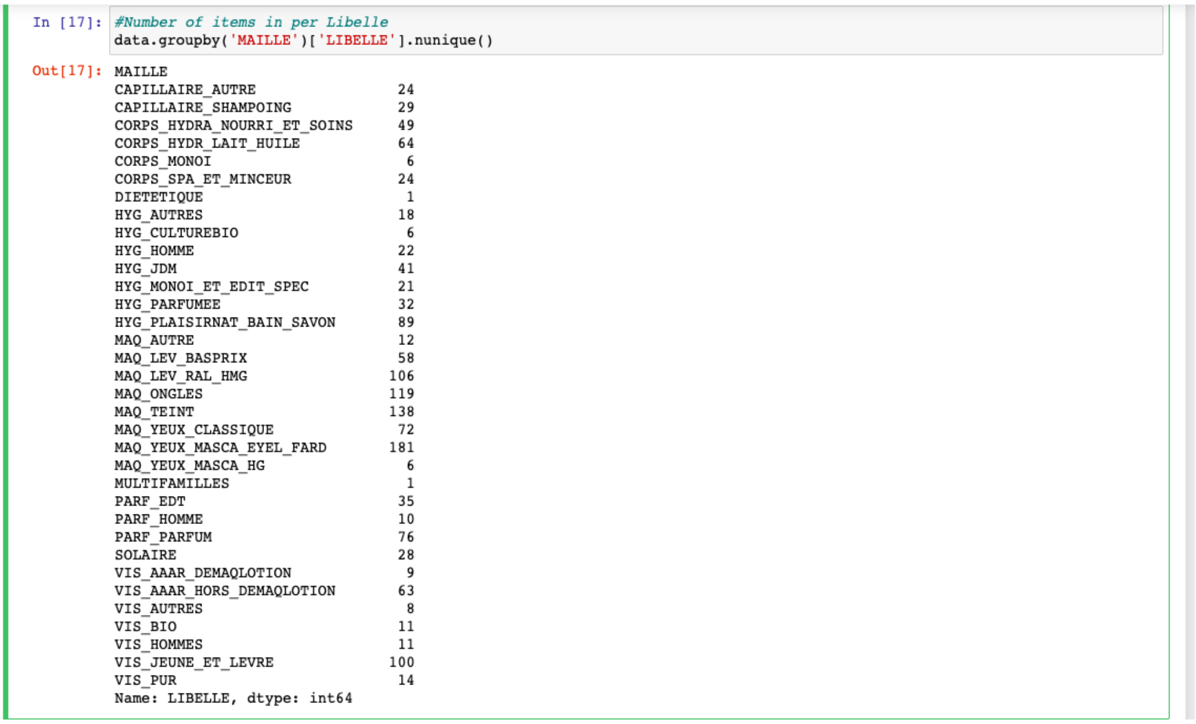
💡 Number of items per maille



```
In [16]: #Number of items in per Univers
         data.groupby('UNIVERS')['LIBELLE'].nunique()

Out[16]: UNIVERS
         CAP_AP SHAMP                  6
         CAP_SHAMP SPECIFIQUE         10
         CAP_SHAMP TRAITANT            4
         CAP_SHAMP TSCHEVEUX          15
         CAP_TENUE DE LA COIFFURE      6
                                      ..
         VIS_SOIN HOMMES              11
         VIS_SOIN LEVRES              30
         VIS_TRAIT AAAR               19
         VIS_TRAIT BIO                 3
         VIS_TRAIT Jeunes Specifique  10
         Name: LIBELLE, Length: 105, dtype: int64
```

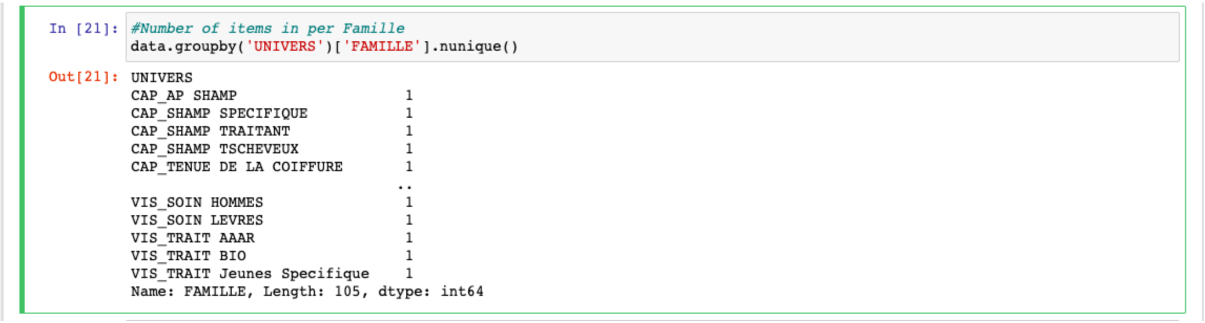
Number of items per univers



```
In [17]: #Number of items in per Libelle
         data.groupby('MAILLE')['LIBELLE'].nunique()

Out[17]: MAILLE
         CAPILLAIRE_AUTRE                24
         CAPILLAIRE_SHAMPOING            29
         CORPS_HYDRA_NOURRI_ET_SOINS     49
         CORPS_HYDR_LAIT_HUILE           64
         CORPS_MONOI                      6
         CORPS_SPA_ET_MINCEUR            24
         DIETETIQUE                       1
         HYG_AUTRES                      18
         HYG_CULTUREBIO                   6
         HYG_HOMME                       22
         HYG_JDM                         41
         HYG_MONOI_ET_EDIT_SPEC          21
         HYG_PARFUMEE                    32
         HYG_PLAISIRNAT_BAIN_SAVON       89
         MAQ_AUTRE                       12
         MAQ_LEV_BASPRIX                 58
         MAQ_LEV_RAL_HMG                106
         MAQ_ONGLES                     119
         MAQ_TEINT                      138
         MAQ_YEUX_CLASSIQUE              72
         MAQ_YEUX_MASCA_EYEL_FARD       181
         MAQ_YEUX_MASCA_HG                6
         MULTIFAMILLES                    1
         PARF_EDT                        35
         PARF_HOMME                      10
         PARF_PARFUM                     76
         SOLAIRE                         28
         VIS_AAAR_DEMAQLOTION             9
         VIS_AAAR_HORS_DEMAQLOTION       63
         VIS_AUTRES                       8
         VIS_BIO                         11
         VIS_HOMMES                      11
         VIS_JEUNE_ET_LEVRE             100
         VIS_PUR                         14
         Name: LIBELLE, dtype: int64
```

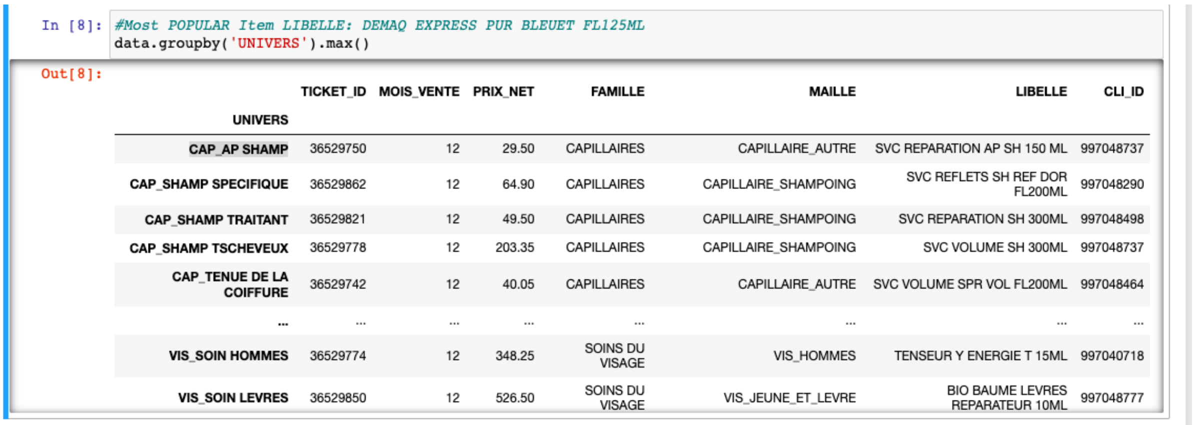
Number of items per famille



```
In [21]: #Number of items in per Famille
         data.groupby('UNIVERS')['FAMILLE'].nunique()

Out[21]: UNIVERS
         CAP_AP SHAMP                  1
         CAP_SHAMP SPECIFIQUE          1
         CAP_SHAMP TRAITANT            1
         CAP_SHAMP TSCHEVEUX           1
         CAP_TENUE DE LA COIFFURE      1
                                      ..
         VIS_SOIN HOMMES               1
         VIS_SOIN LEVRES               1
         VIS_TRAIT AAAR                1
         VIS_TRAIT BIO                 1
         VIS_TRAIT Jeunes Specifique   1
         Name: FAMILLE, Length: 105, dtype: int64
```

Most popular items in each category:
By grouping the items in category we can get the most solde items (the most popular) in the dataset, the table shows the result



```
In [8]: #Most POPULAR Item LIBELLE: DEMAQ EXPRESS PUR BLEUET FL125ML
        data.groupby('UNIVERS').max()
```

Out[8]:

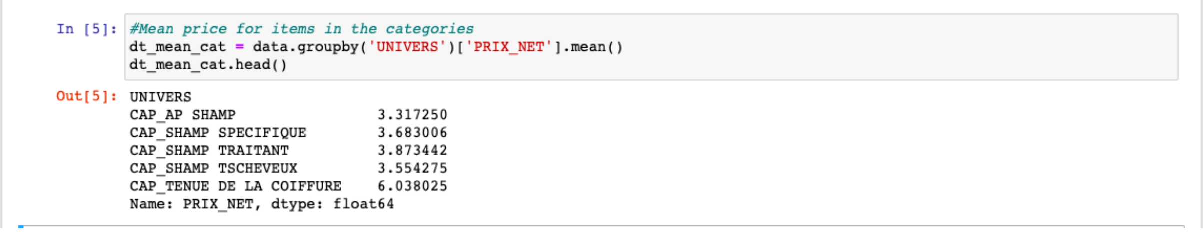
	TICKET_ID	MOIS_VENTE	PRIX_NET	FAMILLE	MAILLE	LIBELLE	CLI_ID
UNIVERS							
CAP_AP SHAMP	36529750	12	29.50	CAPILLAIRES	CAPILLAIRE_AUTRE	SVC REPARATION AP SH 150 ML	997048737
CAP_SHAMP SPECIFIQUE	36529862	12	64.90	CAPILLAIRES	CAPILLAIRE_SHAMPOING	SVC REFLETS SH REF DOR FL200ML	997048290
CAP_SHAMP TRAITANT	36529821	12	49.50	CAPILLAIRES	CAPILLAIRE_SHAMPOING	SVC REPARATION SH 300ML	997048498
CAP_SHAMP TSCHEVEUX	36529778	12	203.35	CAPILLAIRES	CAPILLAIRE_SHAMPOING	SVC VOLUME SH 300ML	997048737
CAP_TENUE DE LA COIFFURE	36529742	12	40.05	CAPILLAIRES	CAPILLAIRE_AUTRE	SVC VOLUME SPR VOL FL200ML	997048464
...	...	...	...	...	...	...	...
VIS_SOIN HOMMES	36529774	12	348.25	SOINS DU VISAGE	VIS_HOMMES	TENSEUR Y ENERGIE T 15ML	997040718
VIS_SOIN LEVRES	36529850	12	526.50	SOINS DU VISAGE	VIS_JEUNE_ET_LEVRE	BIO BAUME LEVRES REPARATEUR 10ML	997048777

The mean price spend about 5.97

The mean number of items per tickets:
To calculate the mean number of items per tickets, start by calculating the number of tickets we have, for that we estimate it with nunique() function **2 734 841 ticket**, then calculate the number of items solde for all this tickets using sum() function: 7 245 522
So, the mean number of items by ticket is about 3 items (=2.64)

The mean number of items per clients:
like we have already calculate the mean item per tickets we know we have 7 245 522 item solde and we have about 853 514 client, the mean is about 8 items (=8.48)

The mean price for items in the category:
Can be calculated with this formula data.groupby('UNIVERS')['PRIX_NET'].mean()



```
In [5]: #Mean price for items in the categories
        dt_mean_cat = data.groupby('UNIVERS')['PRIX_NET'].mean()
        dt_mean_cat.head()

Out[5]: UNIVERS
        CAP_AP SHAMP                3.317250
        CAP_SHAMP SPECIFIQUE        3.683006
        CAP_SHAMP TRAITANT          3.873442
        CAP_SHAMP TSCHEVEUX         3.554275
        CAP_TENUE DE LA COIFFURE    6.038025
        Name: PRIX_NET, dtype: float64
```

as in the screenshot, for each category we could have the mean price