

Exploitation d'une base de données.

Tableau de bord.



Table des matières :

I – Documentation technique.....	3
a) Objectif du tableau de bord :	3
b) Cible/ Périodicité de la MAJ :	3
c) Graphique n°1 :	4
d) Graphique n°2 :	6
e) Graphique n°3 :	8
f) Graphique n°4 :	10
g) Graphique n°5 :	12
h) Graphique n°6 :	15

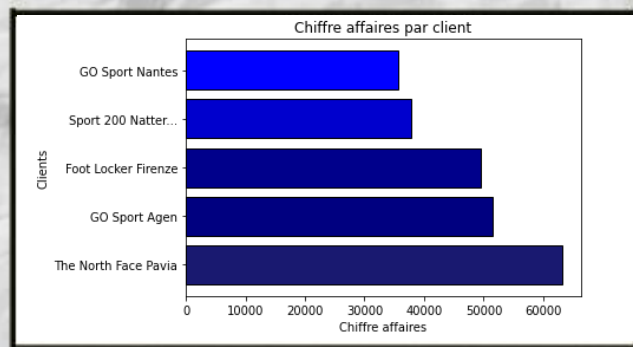
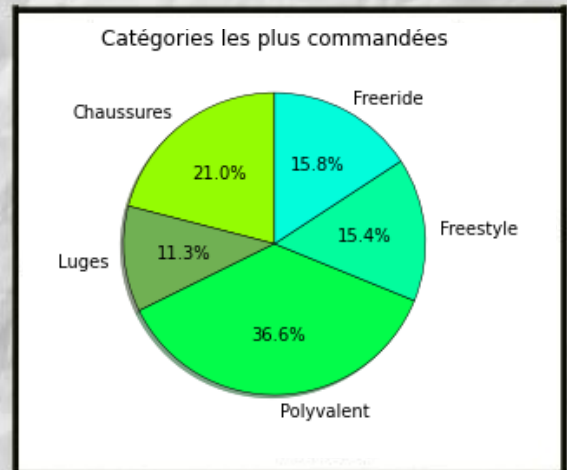
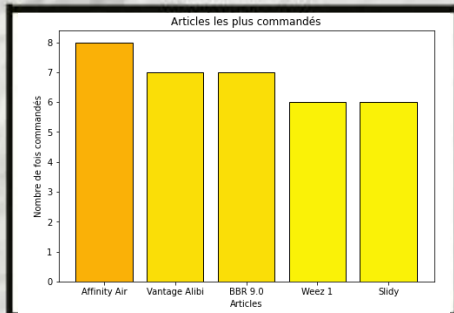
Chiffre d'affaire total : 377 804 €

Tableau de bord 1/2

BIRET-TOSCANO Esteban

Du 27/02 au 28/04

BALMES Bastien



Toutes les données utilisées sont issues
de requêtes SQL, expliquées dans la doc technique)

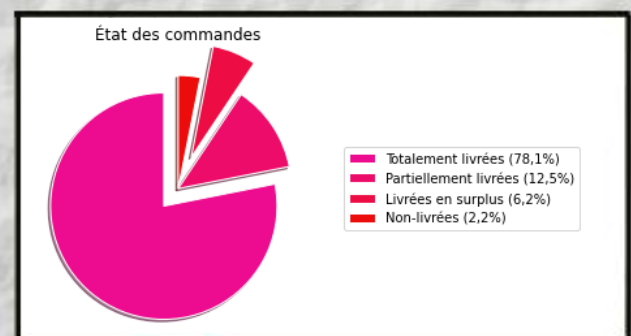
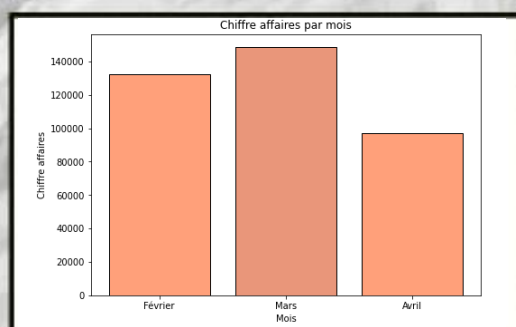
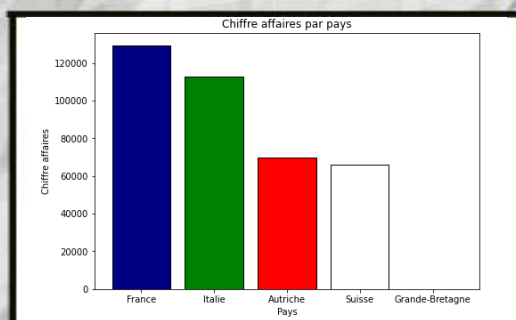
Chiffre d'affaire total : 377 804 €

Tableau de bord 2/2

BIRET-TOSCANO Esteban

Du 27/02 au 28/04

BALMES Bastien



Toutes les données utilisées sont issues
de requêtes SQL, expliquées dans la doc technique)

I – Documentation technique :

a) Objectif du tableau de bord :

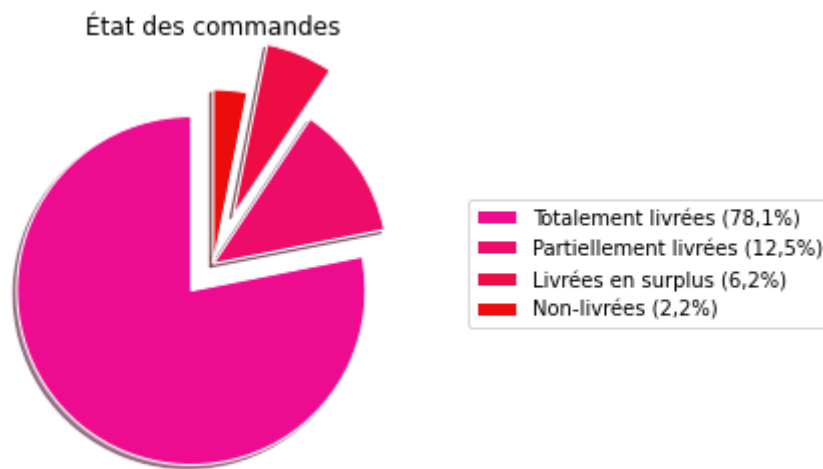
Il permet à l'entreprise de connaître sa situation économique, financière, humaine, commerciale etc pendant une période donnée. Cet outil est une aide aux futures prises de décisions de l'entreprise.

b) Cible/ Périodicité de la MAJ :

Le tableau de bord se base sur les chiffre du mois de février/mars et avril.

Il est destiné au chef d'entreprise, responsable commercial/opérationnel/service achat.

c) Graphique n°1 :



Graphique représentant l'état des commandes, en pourcentage.

Code Python :

```
import matplotlib.pyplot as plt

#données
labels = 'Totalement livrées (78,1%)', 'Partiellement livrées (12,5%)', 'Livrées en surplus (6,2%)', 'Non-livrées (2,2%)'
sizes = [25, 4, 2, 1]

myexplode = [0.2, 0, 0.3, 0]
colors = ['#ed0c8f', '#ed0c6a', '#ed0c44', '#ed0c0c']
patches, texts = plt.pie(sizes, colors=colors, shadow=True, startangle=90,
wedgeprops = {"edgecolor" : "white", 'linewidth': 1, 'antialiased': True}, explode = myexplode)

plt.legend(patches, labels, loc="center", bbox_to_anchor =(1, 0, 1, 1))
plt.title('État des commandes')

#affichage
plt.show()
```

Requêtes SQL :

```
SELECT numCommande
FROM DETAIL_COMMANDE
WHERE quantiteLivree = 0
GROUP BY numCommande ;
```

NUMCOMMANDE
3

Les commandes pas livrées du tout, qui correspondent à 1 commande.

```
SELECT numCommande
FROM DETAIL_COMMANDE
WHERE quantiteLivree < quantiteCommandee
AND quantiteLivree != 0 ;
```

NUMCOMMANDE
1
5
13
23

Les commandes livrées partiellement, qui correspondent à 4 commandes.

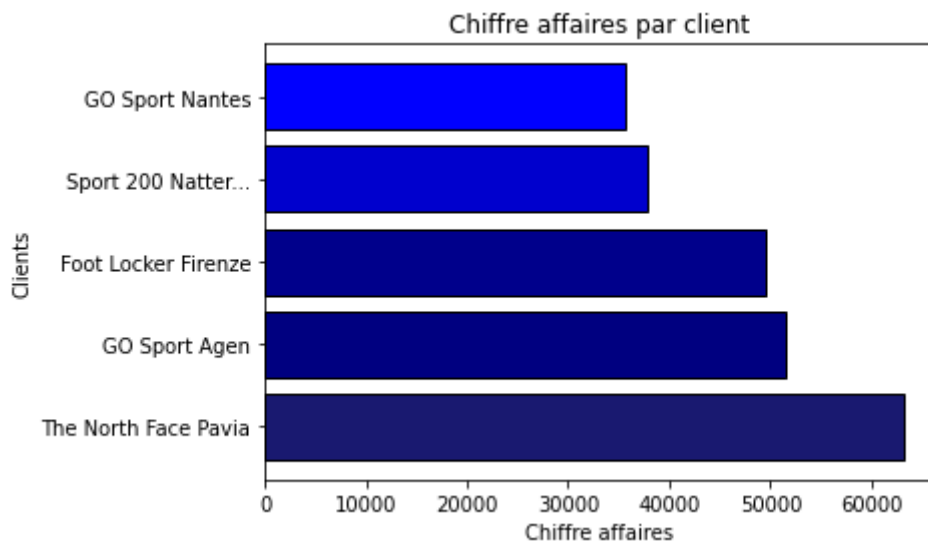
```
SELECT numCommande
FROM DETAIL_COMMANDE
WHERE quantiteLivree > quantiteCommandee
GROUP BY numCommande ;
```

NUMCOMMANDE
14
24

Les commandes livrées en surplus, qui correspondent à 2 commandes.

Les données dans 'sizes' correspondent donc aux résultats de ces requêtes (les commandes livrées correctement sont donc le reste, c'est-à-dire $32 - 1 - 4 - 2 = 25$). En effet, 32 commandes ont été passés au total.

d) Graphique n°2 :



Graphique représentant les 5 plus gros clients par chiffres d'affaires, par ordre décroissant

Code Python :

```
import matplotlib.pyplot as plt
import numpy as np

#données x et y
chiffre_affaires = [63188, 51525, 49540, 37775, 35575]
clients = ('The North Face Pavia', 'GO Sport Agen', 'Foot Locker Firenze', 'Sport 200 Natter...', 'GO Sport Nantes')

y_pos = np.arange(len(clients))
plt.barh(y_pos, chiffre_affaires, color = ['midnightblue', 'navy', 'darkblue', 'mediumblue', 'blue'], edgecolor = 'black')
plt.yticks(y_pos, clients)
plt.xlabel('Chiffre affaires')
plt.ylabel('Clients')

plt.title('Chiffre affaires par client')

#affichage
plt.show()
```

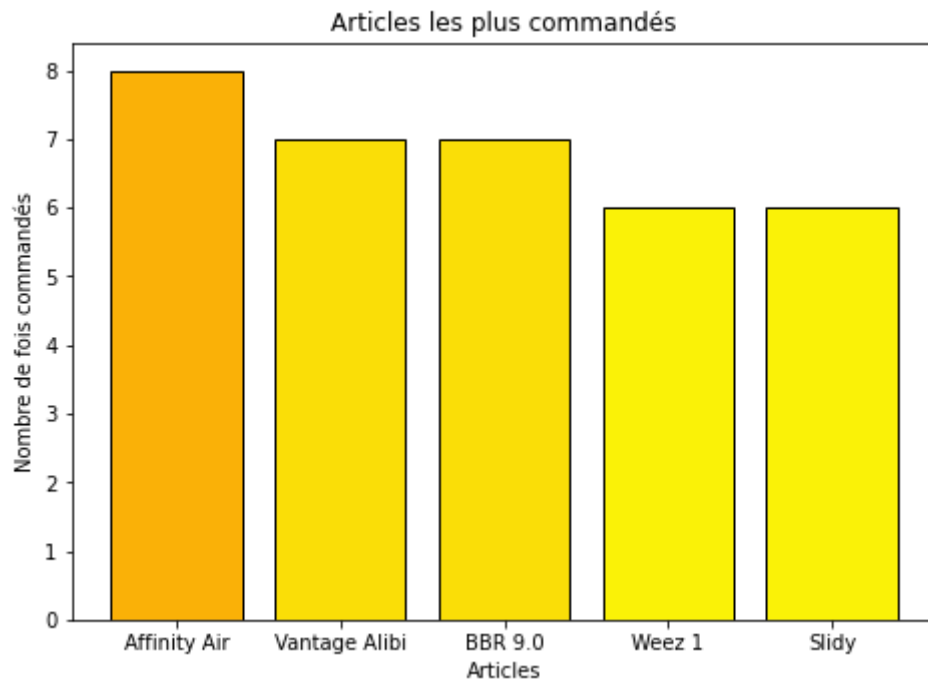
Requête SQL :

```
SELECT C.numClient, C.nomClient, SUM(ROUND (CO.montantTTC, 0)) AS CHIFFRE_AFFAIRE
FROM CLIENT C, COMMANDE CO
WHERE C.numClient = CO.numClient
GROUP BY C.numClient, C.nomClient
ORDER BY CHIFFRE_AFFAIRE DESC ;
```

NUMCLIENT	NOMCLIENT	CHIFFRE_AFFAIRE
17	The North Face Pavia	63188
1	GO Sport Agen	51525
16	Foot Locker Firenze	49540
12	SPORT 2000 Natter Talstation Bergbahnen	37775
3	GO Sport Nantes	35575
2	GO Sport Albi	34001
9	OCHSNER Sport Bull	33540
13	Olympiaplatz 2 EKZ Stadion Center	31925
11	OCHSNER Sport Stans	18981
4	GO Sport Aulnay	17712
5	GO Sport Paris 12	16694
8	OCHSNER Sport Neuchatel	14184
10	OCHSNER Sport Luzern	12600

Pour effectuer notre graphique, il nous suffit de prendre le chiffre d'affaires et les noms clients des 5 premières lignes.

e) Graphique n°3 :



Graphique représentant les articles les plus commandées.

Code Python :

```
import matplotlib.pyplot as plt
fig = plt.figure()
ax = fig.add_axes([0,0,1,1])

#données x et y
articles = ['Affinity Air', 'Vantage Alibi', 'BBR 9.0', 'Weez 1', 'Slidy']
nb_fois_commandes = [8,7,7,6,6]

ax.bar(articles,nb_fois_commandes, color = ['#fab107', '#fade07', '#fade07', '#faf207', '#faf207'], edgecolor = 'black')
plt.xlabel('Articles')
plt.ylabel('Nombre de fois commandés')
plt.title('Articles les plus commandés')

#affichage
plt.show()
```

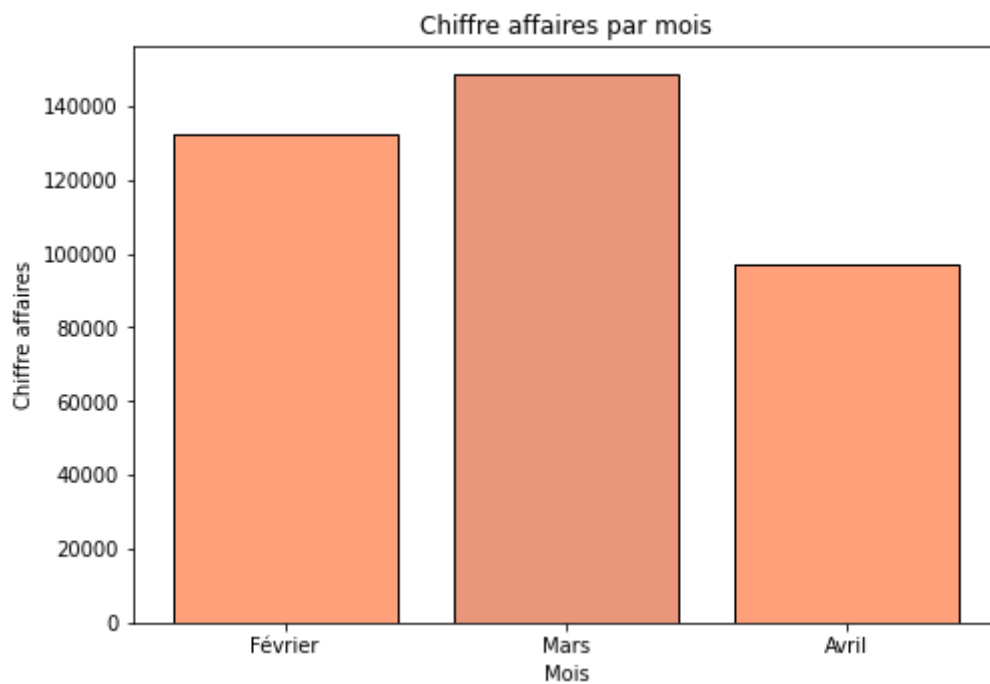

Requête SQL :

```
SELECT A.nomArticle, A.numArticle, COUNT (D.numCommande) AS NB_FOIS_COMMANDE
FROM ARTICLE A, DETAIL_COMMANDE D
WHERE A.numArticle = D.numArticle
GROUP BY (A.numArticle, A.nomArticle) ;
ORDER BY NB_FOIS_COMMANDE DESC ;
```

NOMARTICLE	NUMARTICLE	NB_FOIS_COMMANDE
1 AFFINITY AIR	25	8
2 VANTAGE ALIBI	27	7
3 BBR 9.0	26	7
4 WEEZ 1	5	6
5 SLIDY	1	6
6 Q-85	18	4
7 DERVY	32	4
8 Q-98	19	3
9 SICKSTICK	30	3
10 Q-115	17	3
11 SL SERIE	28	2
12 QUES MAX 110	11	2
13 EXPERIENCE 83 TPX	20	2
14 RESERVE	31	2
15 XLT	29	2
16 WEEZ 2	3	2
17 QUICKY	2	2
18 Q-LAB	16	2
19 LIVE FIT 70	10	2
20 LIVE FIT 60	9	2
21 GHOST 90	8	2
22 WAVE1	4	2
23 Q-90	15	1
24 VR27	22	1
25 X PRO 110	14	1
26 LIVE FIT 130	13	1
27 SMOKE	21	1
28 QUES MAX BC 120	12	1
29 VR07	24	1
30 X-LANDER 6.0	23	1
31 WAVE 2	6	1

Pour effectuer notre graphique, il nous suffit de prendre le nom des articles et leur nombre de fois commandés des 5 premières lignes.

f) Graphique n°4 :



Graphique représentant le chiffre d'affaires de l'entreprise par mois.

Code Python :

```
import matplotlib.pyplot as plt

fig = plt.figure()
ax = fig.add_axes([0,0,1,1])

#données x et y
mois = ['Février', 'Mars', 'Avril']
chiffre_affaires = [132082, 148837, 96884]

x_pos = np.arange(len(mois))
ax.bar(mois, chiffre_affaires)
plt.bar(x_pos, chiffre_affaires, color=['lightsalmon', 'darksalmon'], edgecolor='black')
plt.xlabel('Mois')
plt.ylabel('Chiffre affaires')
plt.title('Chiffre affaires par mois')

#affichage
plt.show()
```

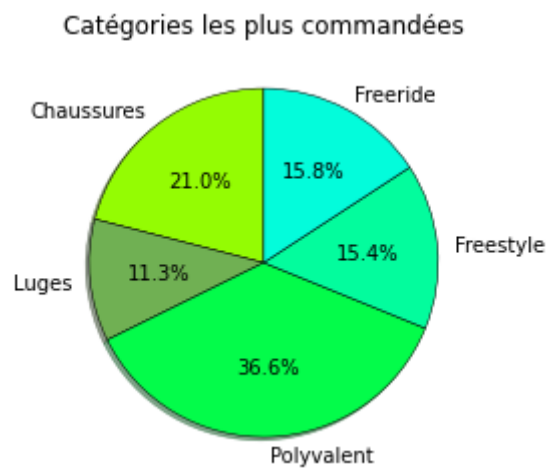
Requête SQL :

```
SELECT SUM (montantHT)AS montantTTC, TO_CHAR (dateCommande, 'MONTH') AS MOIS  
FROM COMMANDE  
GROUP BY dateCommande, TO_CHAR (dateCommande, 'MONTH') ;
```

	MONTANTTTC	MOIS
1	84584	AVRIL
2	132082,6	FÉVRIER
3	148837,5	MARS
4	12300	AVRIL

On a donc **132 082** pour le mois de février, **148 837** pour le mois de mars et **96 884** pour le mois d'avril. On utilise donc ces valeurs pour créer notre graphique.

g) Graphique n°5 :



Graphique représentant le pourcentage des catégories les plus commandées.

Code Python :

```
import matplotlib.pyplot as plt

#données
labels = 'Chaussures', 'Luges', 'Polyvalent', 'Freestyle', 'Freeride'
nb_commandes = [489, 264, 852, 358, 368]

colors = ['#94fc03', '#70b053', '#03fc4a', '#03fc9d', '#03fcdb']
fig1, ax1 = plt.subplots()
ax1.pie(nb_commandes, labels=labels, autopct='%1.1f%%', shadow=True, startangle=90,
        wedgeprops = {"edgecolor" : "black", 'linewidth': 0.5, 'antialiased': True},
        colors=colors)
ax1.axis('equal')
plt.title('Catégories les plus commandées')

#affichage
plt.show()
```

Requêtes SQL :

```
SELECT C.libellecategorie, SUM(D.quantitecommandee)
FROM CATEGORIE C, DETAIL_COMMANDE D
WHERE D.numarticle IN (1, 2, 3, 4, 5, 6)
AND C.libellecategorie = 'Luges'
GROUP BY C.libellecategorie ;
```

LIBELLECATEGORIE	SUM(D.QUANTITECOMMANDEE)
1 Luges	264

```
SELECT C.libellecategorie, SUM(D.quantitecommandee)
FROM CATEGORIE C, DETAIL_COMMANDE D
WHERE D.numarticle IN (8, 9, 10, 11, 12, 13, 14)
AND C.libellecategorie = 'Chaussures'
GROUP BY C.libellecategorie ;
```

LIBELLECATEGORIE	SUM(D.QUANTITECOMMANDEE)
1 Chaussures	489

```
SELECT C.libellecategorie, SUM(D.quantitecommandee)
FROM CATEGORIE C, DETAIL_COMMANDE D
WHERE D.numarticle IN (15, 16, 17, 18, 19)
AND C.libellecategorie = 'Freeride'
GROUP BY C.libellecategorie ;
```

LIBELLECATEGORIE	SUM(D.QUANTITECOMMANDEE)
1 Freeride	368

```
SELECT C.libellecategorie, SUM(D.quantitecommandee)
FROM CATEGORIE C, DETAIL_COMMANDE D
WHERE D.numarticle IN (20, 21, 22, 23, 24, 25, 26, 27)
AND C.libellecategorie = 'Polyvalent'
GROUP BY C.libellecategorie ;
```

LIBELLECATEGORIE	SUM(D.QUANTITECOMMANDEE)
1 Polyvalent	852

```

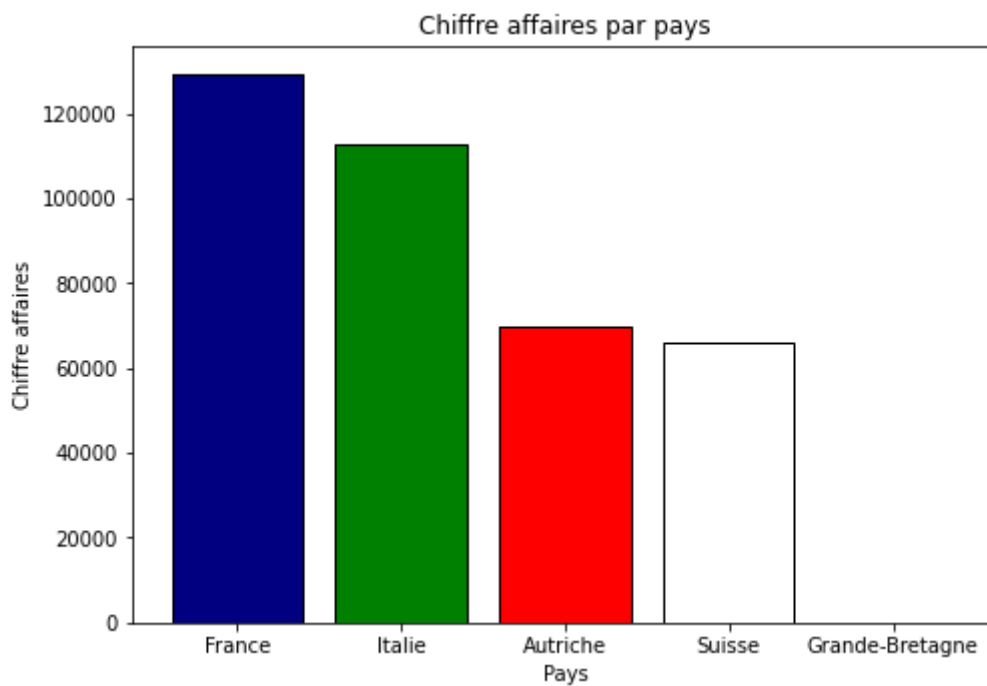
SELECT C.libellecategorie, SUM(D.quantitecommandee)
FROM CATEGORIE C, DETAIL_COMMANDE D
WHERE D.numarticle IN (28, 29, 30, 31, 32)
AND C.libellecategorie = 'Freestyle'
GROUP BY C.libellecategorie ;

```

	LIBELLECATEGORIE	SUM(D.QUANTITECOMMANDEE)
1	Freestyle	358

Les valeurs dans le **IN** ont été relevés en amont, pour avoir les numéros d'article de chaque catégorie.

h) Graphique n°6 :



Graphique représentant le chiffre d'affaires par pays.

Code Python :

```
import matplotlib.pyplot as plt

fig = plt.figure()
ax = fig.add_axes([0,0,1,1])

#données x et y
pays = ['France', 'Italie', 'Autriche', 'Suisse', 'Grande-Bretagne']
chiffre_affaires = [129489, 112627, 69600, 66087, 0]

x_pos = np.arange(len(pays))
ax.bar(pays, chiffre_affaires)
plt.bar(x_pos, chiffre_affaires, color=['navy', 'green', 'red', 'white'], edgecolor='black')
plt.xlabel('Pays')
plt.ylabel('Chiffre affaires')
plt.title('Chiffre affaires par pays')

#affichage
plt.show()
```

Requêtes SQL :

```
SELECT E.codeEtiquette, SUM(CO.montantHT) AS montant_Ventes
```

```
FROM COMMANDE CO, ETIQUETTE E, CLIENT C
```

```
WHERE E.codeTypeTva = 1
```

```
AND E.codeEtiquette = C.codeEtiquette
```

```
AND CO.numClient = C.numClient
```

```
GROUP BY E.codeEtiquette ;
```

	CODEETIQUETTE	MONTANT_VENTES
1	FR	129489,1

```
SELECT E.codeEtiquette, SUM(CO.montantHT) AS montant_Ventes
```

```
FROM COMMANDE CO, ETIQUETTE E, CLIENT C
```

```
WHERE E.codeTypeTva = 2
```

```
AND E.codeEtiquette = C.codeEtiquette
```

```
AND CO.numClient =
```

```
C.numClient
```

```
GROUP BY E.codeEtiquette ;
```

	CODEETIQUETTE	MONTANT_VENTES
1	CH	66087,5

```
SELECT E.codeEtiquette, SUM(CO.montantHT) AS montant_Ventes
```

```
FROM COMMANDE CO, ETIQUETTE E, CLIENT C
```

```
WHERE E.codeTypeTva = 3
```

```
AND E.codeEtiquette = C.codeEtiquette
```

```
AND CO.numClient = C.numClient
```

```
GROUP BY E.codeEtiquette ;
```

	CODEETIQUETTE	MONTANT_VENTES
1	AT	69600
2	IT	112627,5

La Grande-Bretagne n'apparaît pas car elle n'a pas passé de commande, elle a juste été un fournisseur pour l'entreprise.