

BUT Informatique - Semestre 2 (2021/2022)

S2.04 - Exploitation d'une base de données

Étape 3 : Interrogation dynamique de la base



Solution informatique

L'application est divisée en deux fichiers, l'un étant les menus et leur fonctionnement, l'autre les fonctions créant les graphiques

Main :

```
import graph as g

def execute(req,date):
    if(req==1):
        g.reqAI(date)
    elif(req==2):
        g.reqAP(date)
    elif(req==3):
        g.reqAE(date)
    else:
        g.reqAL(date)

#----- MAIN -----#
actif=True
print("\n .... Bienvenue .... \n")
while(actif):
    print("Veuillez choisir l'une des possible incohérence : ")
    print("1 | Les accidents avec un sol incoherent par rapport au temps")
    print("2 | Les accidents sous la pluie ou non")
    print("3 | Les accidents dû à un éblouissement par phare en plein
jour")
    print("4 | Les accidents ayant lieu toujours au même endroit")
    choixReq=int(input("Entrez 1, 2, 3, ou 4 : "))
    while(choixReq>5 or choixReq<1):
        print("Erreur")
        choixReq=int(input("Entrez 1, 2, 3, ou 4 : "))
    choixDate=int(input("Selectionnez une année entre 1984 et 1998 ou
toutes(0) : "))
    while((choixDate>1998 or choixDate<1984) and choixDate!=0):
        print("Erreur")
        choixDate=int(input("Selectionnez une année entre 1984 et 1998 ou
toutes(0) : "))
    execute(choixReq,choixDate)
    choixQuit=int(input("Accédez a l'accueil(1) ou quittez(0) : "))
    while(choixQuit>1 or choixQuit<0):
        print("Erreur")
        choixQuit=int(input("\n Accéder a l'accueil(1) ou quitter(0) : "))
    if(choixQuit==0):
        actif=False
```

Fonction requête 1 (Intempérie/état du sol) :

```
def reqAI(date):
    # Préparation de la connexion à la BD avec le DSN
    conn = pyodbc.connect('DSN=Lakartxela')

    if date==0 :
        sql="SELECT MAccident.accident_id,MEtatSurface.libelle_etat_surface
AS 'Etat surface' ,MIntemperie.libelle AS 'Intemperie' \
        FROM `MAccident`,MEtatSurface,MIntemperie WHERE
MAccident.etat_surface_id = MEtatSurface.code_etat_surface AND
MAccident.intemp_id = MIntemperie.code "
    else :
        sql="SELECT MAccident.accident_id,MEtatSurface.libelle_etat_surface
AS 'Etat surface' ,MIntemperie.libelle AS 'Intemperie' \
        FROM `MAccident`,MEtatSurface,MIntemperie WHERE
MAccident.etat_surface_id = MEtatSurface.code_etat_surface AND
MAccident.intemp_id = MIntemperie.code \
        AND `date_id` IN (SELECT date_id FROM MDate WHERE
YEAR(MDate.DateFormatStandard) =" +str(date)+" )"

    dataframeEtatSurfaceIntemperie = pd.read_sql(sql,conn)

    graphiqueEtatSurfaceIntemperie =
pd.crosstab(dataframeEtatSurfaceIntemperie["Etat surface"],
dataframeEtatSurfaceIntemperie["Intemperie"], normalize="index")
    graphiqueEtatSurfaceIntemperie.plot(kind="bar",stacked=True)

    plt.title("Répartition de la météo pour chaque état de la surface")
    plt.legend(bbox_to_anchor=(1.40, 1.0), loc="upper right")
    plt.show()
```

Fonction requête 2 (Pluie) :

```
def reqAP(date):
    dicIntemperie = {"Beau temps":"Beau temps", #regrouper brouillard
tempete et grele
                    "Pluie forte":"Mauvais temps",
                    "Pluie legere":"Mauvais temps",
                    "Neige":"Mauvais temps",
                    "Grele":"Mauvais temps",
                    "Brouillard":"Mauvais temps",
                    "Vent fort tempete":"Mauvais temps",
                    "Inconnu":"Temps inconnu"}

    conn = pyodbc.connect('DSN=Lakartxela')

    if date==0 :
        sql="SELECT MAccident.accident_id AS 'Accident',MIntemperie.libelle
AS 'Intemperie' \
```

```

        FROM `MAccident` JOIN MIntemperie ON MAccident.intemp_id =
MIntemperie.code"
    else :
        sql="SELECT MAccident.accident_id AS 'Accident',MIntemperie.libelle
AS 'Intemperie' \
        FROM `MAccident` JOIN MIntemperie ON MAccident.intemp_id =
MIntemperie.code WHERE MAccident.date_id IN \
        (SELECT date_id FROM MDate WHERE YEAR(MDate.DateFormatStandard)
="+str(date)+" )"

    dataframeAccidentIntemperie = pd.read_sql(sql,conn)
    dataframeAccidentIntemperie["Nombre d'accidents"] = 1
    dataframeAccidentIntemperie['Meteo'] =
dataframeAccidentIntemperie['Intemperie'].map(dicIntemperie)

    graphiqueAccidentIntemperie = dataframeAccidentIntemperie[["Meteo",
"Nombre d'accidents"]].groupby("Meteo").sum()
    graphiqueAccidentIntemperie =
graphiqueAccidentIntemperie.sort_values("Nombre d'accidents", ascending =
False)
    graphiqueAccidentIntemperie.plot(kind="pie", y="Nombre d'accidents")

    plt.text(-1.19,-1.2,"Il n\ ' y a aucun accident lorsqu\'il fait beau")
    plt.title("Répartition des accidents en fonction de la météo")
    plt.ylabel("")
    plt.legend("",frameon=False)
    plt.show()

```

Fonction requête 3 (Eclairage) :

```

def reqAE(date):

    conn = pyodbc.connect('DSN=Lakartxela')

    if date==0 :
        sql="SELECT MAccident.accident_id AS 'Accident',MLuminosite.libelle
AS 'Luminosite' FROM `MAccident` \
        JOIN MLuminosite ON MAccident.lum_id = MLuminosite.code WHERE
cause_id = 86"
    else :
        sql="SELECT MAccident.accident_id AS 'Accident',MLuminosite.libelle
AS 'Luminosite' FROM `MAccident` \
        JOIN MLuminosite ON MAccident.lum_id = MLuminosite.code WHERE
cause_id = 86 AND MAccident.date_id IN \
        (SELECT date_id FROM MDate WHERE YEAR(MDate.DateFormatStandard)
="+str(date)+" )"

    dataframeCauseLuminosite = pd.read_sql(sql,conn)
    dataframeCauseLuminosite["Nombre d'accidents par éblouissement par les
phares"] = 1

```

```

    graphiqueCauseLuminosite = dataframeCauseLuminosite[["Luminosite",
"Nombre d'accidents par éblouissement par les
phares"]].groupby("Luminosite").sum()
    graphiqueCauseLuminosite = graphiqueCauseLuminosite.sort_values("Nombre
d'accidents par éblouissement par les phares", ascending = False)
    graphiqueCauseLuminosite.plot(kind="bar")

    plt.title("Répartition des accidents par éblouissement\npar les phares
en fonction de la luminosité")
    plt.ylabel("")
    plt.legend("",frameon=False)
    plt.show()

```

Fonction requête 4 (Lieux) :

```

def reqAL(date):

    conn = pyodbc.connect('DSN=Lakartxela')

    if date==0 :
        sql="SELECT MLieu.lieu_id AS 'Lieu',COUNT(MAccident.lieu_id) AS
Nombre FROM MAccident \
            JOIN MLieu ON MLieu.lieu_id = MAccident.lieu_id GROUP BY
MAccident.lieu_id ORDER BY `Nombre` DESC LIMIT 5;"
    else :
        sql="SELECT MLieu.lieu_id AS 'Lieu',COUNT(MAccident.lieu_id) AS
Nombre FROM MAccident \
            JOIN MLieu ON MLieu.lieu_id = MAccident.lieu_id WHERE
MAccident.date_id IN \
            (SELECT date_id FROM MDate WHERE YEAR(MDate.DateFormatStandard)
= 1984 ) GROUP BY MAccident.lieu_id ORDER BY `Nombre` DESC LIMIT 5;"

    dataframeAccidentLieu = pd.read_sql(sql,conn)
    graphiqueAccidentLieu = dataframeAccidentLieu.set_index('Lieu')
    #graphiqueCauseLuminosite =
graphiqueCauseLuminosite.sort_values("Nombre d'accidents par lieu",
ascending = False)
    graphiqueAccidentLieu.plot(kind="bar")

    plt.title("Répartition des lieux par leur nombre d'accident")
    plt.ylabel("Nombre")
    plt.legend("",frameon=False)
    plt.show()

```

Le menu est d'abord une présentation de nos 4 requêtes, proposant à l'utilisateur de taper le numéro de la requête qu'il souhaite.

```
.... Bienvenue ....

Veuillez choisir l'une des possible incohérence :
1 | Les accidents avec un sol incoherent par rapport au temps
2 | Les accidents sous la pluie ou non
3 | Les accidents dû à un éblouissement par phare en plein jour
4 | Les accidents ayant lieu toujours au même endroit

Entrez 1, 2, 3, ou 4 :
```

(fig.1)

Ensuite, l'utilisateur peut choisir l'année qui l'intéresse, s'il ne souhaite voir aucune année en particulier, il peut taper 0 pour voir le graphique de toutes les années

```
.... Bienvenue ....

Veuillez choisir l'une des possible incohérence :
1 | Les accidents avec un sol incoherent par rapport au temps
2 | Les accidents sous la pluie ou non
3 | Les accidents dû à un éblouissement par phare en plein jour
4 | Les accidents ayant lieu toujours au même endroit

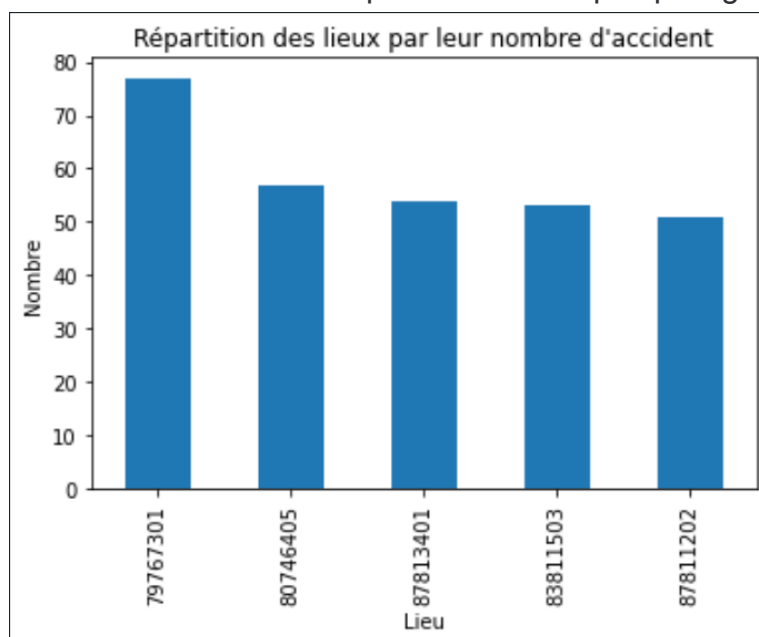
Entrez 1, 2, 3, ou 4 : 4

Selectionnez une année entre 1984 et 1998 ou toutes(0) : 1986

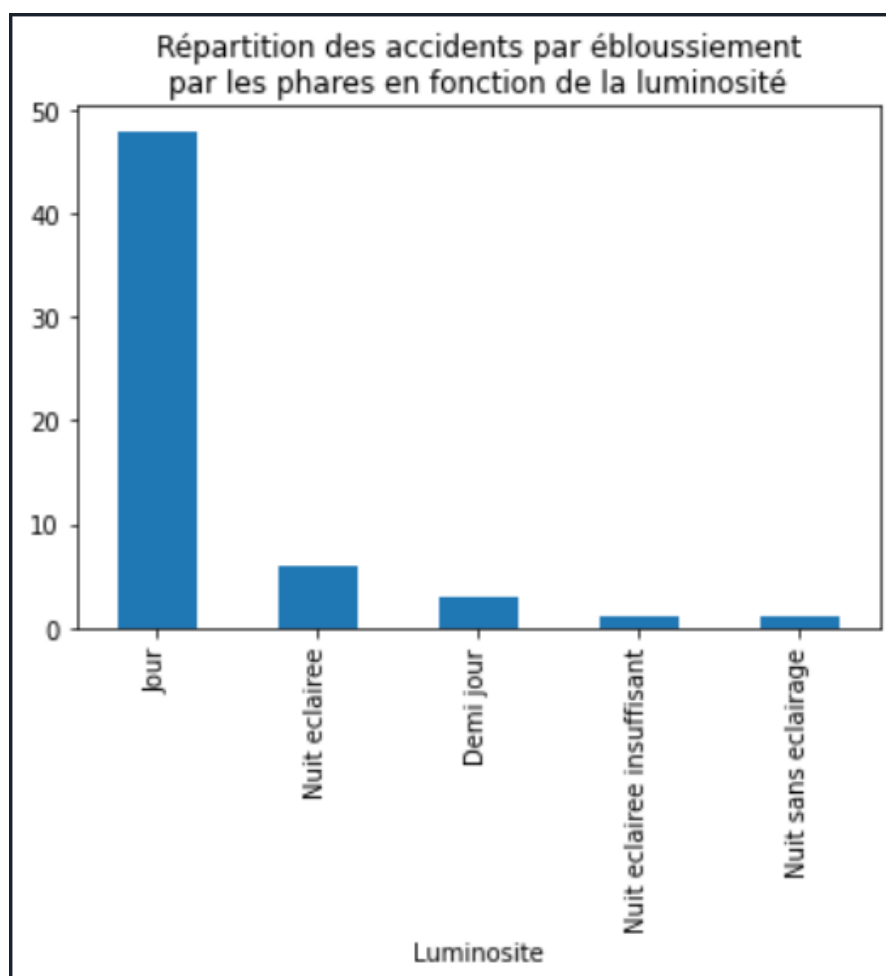
Accédez a l'accueil(1) ou quittez(0) : 1|
```

(fig.2)

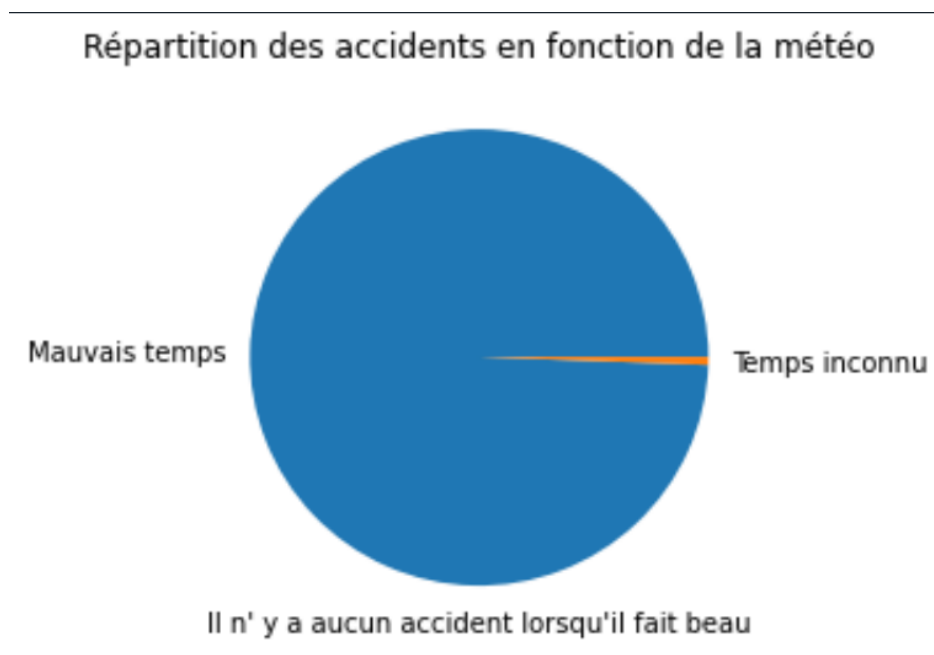
Voici les résultats fournis par le choix de quelques graphiques :



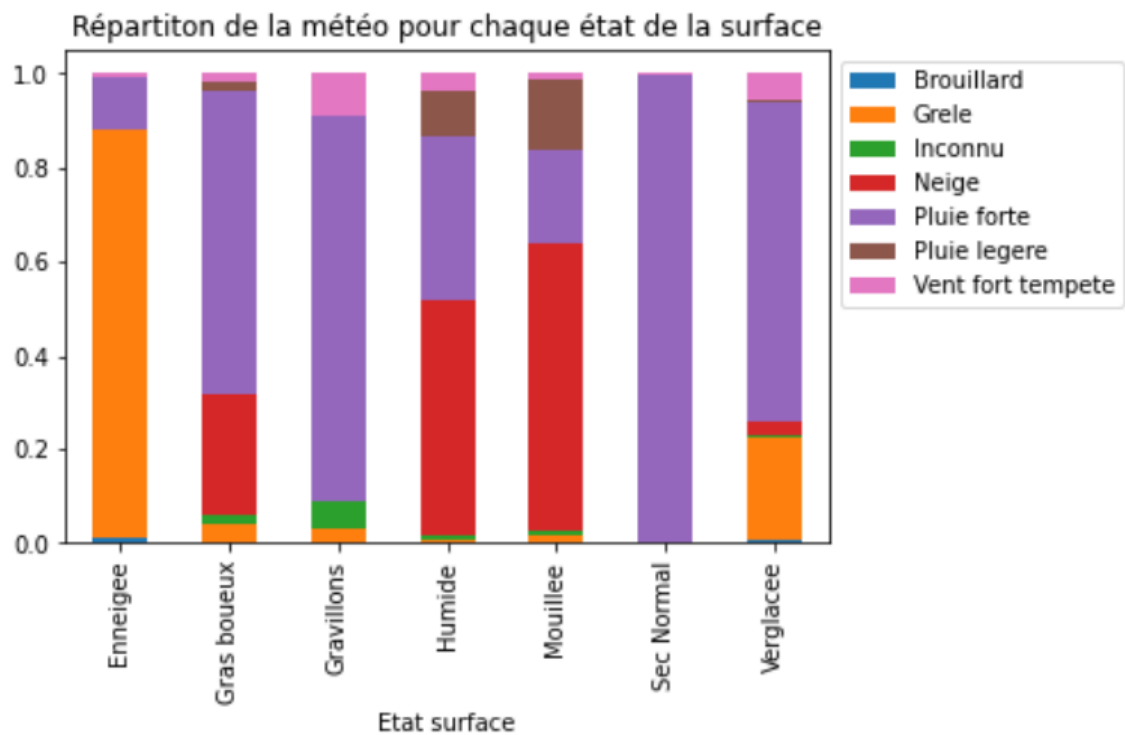
(fig.3)



(fig.4)



(fig.5)



(fig.6)

Une fois le graphique visualisé, l'utilisateur peut choisir de revenir au menu, ou de quitter l'application (voir fig.2).