Evan SPICKA, Guilhem POTIES TD III / TP 5

BUT Informatique Semestre 2 (2021/2022)

S2.04 - Exploitation d'une base de données

Etape 3 : Interrogation (dynamique) de la base

Partie 2



**Précondition du fonctionnement de l’application**

Avant de pouvoir lancer l’application, certaines variables sont à implémenter, certaines bibliothèques à importer et connexion à établir.

**Program.py :**

*#importation des bibliothèques*

*import pyodbc*

*import matplotlib.pyplot as plt*

*#Préparation de la connexion à la base de données ODBC*

*conn=pyodbc.connect('DSN=bd\_accident')*

*#Connexion à la base de données*

*cursor = conn.cursor()*

**Application.py :**

*#importation des fonctions du fichier « Program »*

*import Program as p*

**Interface de l’application**

Lors de l’usage de l’application, il est demandé à l’utilisateur de choisir entre 4 requêtes numérotées et nommées via un champ dédié.

**Code :**

*def choixRequete():*

*print("")*

*print("1 : Influence de la luminosité sur la gravité d’un accident")*

*print("2 : Quelle est la cause principale d’accident de nuit ou de jour peu éclairées")*

*print("3 : Les types de déplacement les plus impliqués en fonction du moment de la journée")*

*print("4 : Taux d’accident selon de jour ou de nuit entre 1984 et 1997")*

*choix = int(input(" quelle requête voulez-vous exécuter : "))*

*print("")*

*if choix == 1:*

*p.gravite()*

*if choix == 2:*

*p.cause()*

*if choix == 3:*

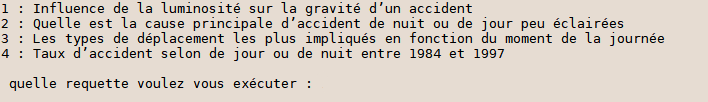
*p.implication()*

*if choix == 4:*

*p.date()*

*choixRequete()*

**Résultat :**

****

**Requête 1**

La première requête consiste à afficher le nombre d’accident sous différents types de luminosité selon un taux de gravité paramétrable.

**Code :**

*def gravite():*

*#Paramètre de la requête*

*val = int(input("Quel taux de gravité voulez-vous observer (de 0 à 3) ? "))*

*#Exécution de la requête*

*req = f"""*

*SELECT Lum.libelle, COUNT(Acc.gravite) AS Total*

*FROM MLuminosite AS Lum*

*JOIN MAccident AS Acc ON Lum.code = Acc.lum\_id*

*WHERE Acc.gravite = {val}*

*GROUP BY Lum.libelle*

*ORDER BY Total DESC*

*"""*

*cursor.execute(req)*

*#Création des listes contenant les informations de la requête*

*luminosite = []*

*nAccident = []*

*for row in cursor.fetchall():*

*luminosite.append(row[0])*

*nAccident.append(row[1])*

*#Création du graphique*

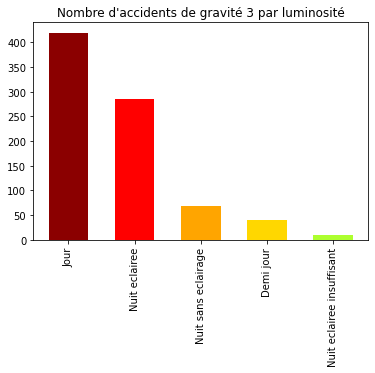
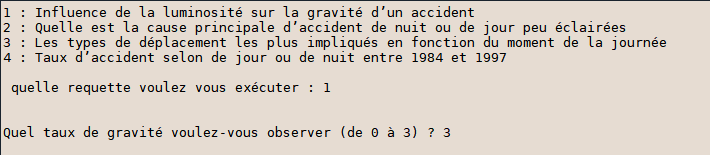
*plt.bar(range(len(luminosite)), nAccident, width = 0.6, color=['darkred','r','orange','gold','greenyellow'])*

*plt.xticks(range(len(luminosite)), luminosite, rotation='vertical')*

*plt.title(f"Nombre d'accidents de gravité {val} par luminosité")*

*plt.show*

**Résultat possible :**



**Requête 2**

La deuxième requête consiste à afficher le nombre d’accidents conséquents (> 400) selon une période de la journée paramétrable.

**Code :**

*def cause():*

*#Paramètre de la requête*

*val = int(input("Sur quelle période de la journée voulez-vous la requette (Jour : 1 | Nuit : 2) : "))*

*if val == 1:*

*param = 'Jour'*

*elif val == 2:*

*param = 'Nuit'*

*#Exécution de la requête*

*req = f"""*

*SELECT C.libelle AS LibelleCause, COUNT(C.Cause) AS SommeCauses*

*FROM MCause AS C*

*JOIN MAccident AS Acc ON C.Cause = Acc.cause\_id*

*JOIN MLuminosite AS Lum On Acc.lum\_id = Lum.code*

*WHERE Lum.type\_luminosite = {val}*

*GROUP BY C.libelle*

*HAVING SommeCauses > 400*

*ORDER BY SommeCauses DESC*

*"""*

*cursor.execute(req)*

*#Création des listes contenant les informations de la requête*

*cause = []*

*nAccident = []*

*for row in cursor.fetchall():*

*cause.append(row[0])*

*nAccident.append(row[1])*

*#Création du graphique*

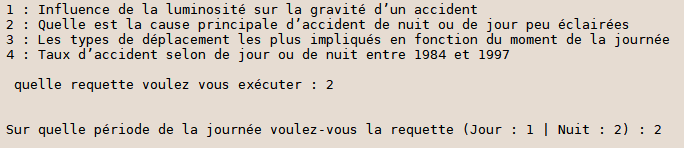
*plt.bar(range(len(cause)), nAccident)*

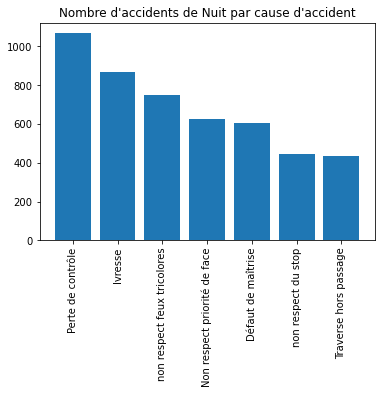
*plt.xticks(range(len(cause)), cause, rotation='vertical')*

*plt.title(f"Nombre d'accidents de {param} par cause d'accident")*

*plt.show*

**Résultat possible :**





**Requête 3**

La troisième requête consiste à afficher la proportion d’accidents par rapport au type de véhicule/déplacement impliqué selon une période de la journée paramétrable.

**Code :**

*def implication():*

*#Paramètre de la requête*

*val = int(input("Sur quelle période de la journée voulez-vous la requette (Jour : 1 | Nuit : 2) : "))*

*if val == 1:*

*param = 'Jour'*

*elif val == 2:*

*param = 'Nuit'*

*#Exécution de la requête*

*req = f"""*

*SELECT TypeImp.libelleType, COUNT(TypeImp.libelleType) AS Total*

*FROM MLuminosite AS Lum*

*JOIN MAccident AS Acc ON Lum.code = Acc.lum\_id*

*JOIN MImplique AS Imp ON Acc.impliq\_id = Imp.code*

*JOIN MTypeImplication AS TypeImp ON Imp.code = TypeImp.id*

*WHERE Lum.type\_luminosite = {val}*

*GROUP BY TypeImp.libelleType*

*"""*

*cursor.execute(req)*

*#Création des listes contenant les informations de la requête*

*periode = []*

*nAccident = []*

*for row in cursor.fetchall():*

*periode.append(row[0])*

*nAccident.append(row[1])*

*#Création du graphique*

*plt.pie(nAccident, labels = periode, colors=['r', 'y', 'g', 'greenyellow'],*

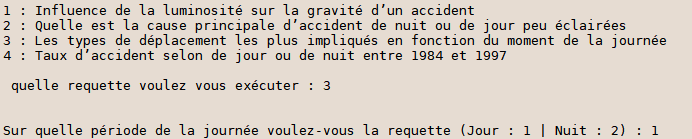
*startangle=90, explode = (0.05, 0.05, 0.05, 0.05),*

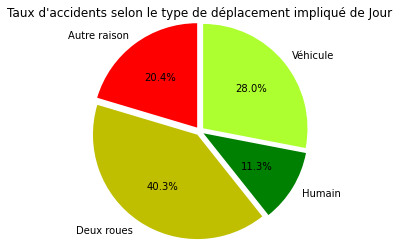
*radius = 1.2, autopct = '%1.1f%%')*

*plt.title(f"Taux d'accidents selon le type de déplacement impliqué de {param}")*

*plt.show*

**Résultat possible :**





**Requête 4**

La quatrième requête consiste à afficher le nombre d’accidents conséquents (> 400) selon une période de la journée paramétrable.

**Code :**

*def date():*

*#Paramètres de la requête*

*print(" année minimale : 1984 | année maximele : 1998")*

*param1 = input("A partir de quelle année voulez vous voir les accidents : ")*

*param2 = input("Jusqu'à quelle année voulez vous voir les accidents : ")*

*#Exécution des deux requettes*

*req1 = f"""*

*SELECT YEAR(d.DateFormatStandard) AS Année, COUNT(Acc.date\_id) AS NombresAccidents*

*FROM MDate AS d*

*JOIN MAccident AS Acc ON d.date\_id = Acc.date\_id*

*JOIN MLuminosite AS Lum ON Acc.lum\_id = Lum.code*

*WHERE Lum.libelle\_luminosite = 'jour' AND YEAR(d.DateFormatStandard) BETWEEN {param1} AND {param2}*

*GROUP BY YEAR(d.DateFormatStandard)*

*"""*

*req2 = f"""*

*SELECT YEAR(d.DateFormatStandard) AS Année, COUNT(Acc.date\_id) AS NombresAccidents*

*FROM MDate AS d*

*JOIN MAccident AS Acc ON d.date\_id = Acc.date\_id*

*JOIN MLuminosite AS Lum ON Acc.lum\_id = Lum.code*

*WHERE Lum.libelle\_luminosite = 'nuit' AND YEAR(d.DateFormatStandard) BETWEEN {param1} AND {param2}*

*GROUP BY YEAR(d.DateFormatStandard)*

*"""*

*cursor.execute(req1)*

*#Création des listes contenant les informations de la requête*

*annees = []*

*nAccidentJour = []*

*nAccidentNuit = []*

*for row in cursor.fetchall():*

*annees.append(str(row[0]))*

*nAccidentJour.append(row[1])*

*cursor.execute(req2)*

*for row in cursor.fetchall():*

*nAccidentNuit.append(row[1])*

*#Création du graphique*

*plt.plot(annees, nAccidentJour, marker="o", label = "Jour", color='y')*

*plt.plot(annees, nAccidentNuit, marker="o", label = "Nuit",color='b')*

*plt.ylim(0,2250)*

*plt.legend()*

*plt.title(f"Nombre d'accidents de jour et de nuit de {param1} à {param2}")*

*plt.show*

**Résultat possible :**

