

Correction des exercices

Savoir gérer les exceptions (try... except...) est indispensable lorsqu'on touche aux fichiers... mais dans l'objectif des concours, tant qu'on ne vous le demande pas, ne vous dispersez pas !

Exercice 1**1.**

```
def temperature():
    tableau = interroge_bdd('SELECT nom_abrege,ptm,temperature FROM
                            polymeres')

    for ligne in tableau:
        if ligne[1]== 'cristallin':
            print(ligne[0],"température de fusion : ",ligne[2])
        else:
            print(ligne[0],"température de transition vitreuse : ",ligne[2])
```

2.

```
def fichier_applications():
    tableau = interroge_bdd('SELECT nom_abrege,nom_application
                            FROM polymeres JOIN applications
                            ON polymeres.id = applications.id_polymere
                            ORDER BY nom_abrege')

    polymere = tableau[0][0]
    ligne_fichier = polymere
    with open('applications.txt','w') as fichier:
        for ligne in tableau:
            if ligne[0] == polymere:
                ligne_fichier += "  "+ligne[1]
            else:
                fichier.write(ligne_fichier+"\n")
                polymere = ligne[0]
                ligne_fichier = polymere+ "  "+ligne[1]
    fichier.write(ligne_fichier)
```

3.

```
def demande_info(polymere):
    with open('production.txt','r') as fichier:
        ligne = fichier.readline()
        numero_ligne = 1
        while ligne != "\n" and ligne.find(polymere) != 0:
            ligne = fichier.readline()
            numero_ligne += 1
        if ligne != "\n":
            ligne_debut = int(ligne[len(polymere)+1:])
            ligne = fichier.readline()
            numero_ligne += 1
            if ligne == "\n":
                ligne_fin = -1
            else:
                ligne_fin = int(ligne[ligne.find('')+1:])-1
        while numero_ligne != ligne_debut-1:
            ligne = fichier.readline()
            numero_ligne += 1
        if ligne_fin == -1:
            print(fichier.read())
        else:
            while numero_ligne != ligne_fin:
                print(fichier.readline())
                numero_ligne += 1
```

Exercice 2

1.

```
def file_to_table(nom_fichier):  
    with open(nom_fichier,'r') as fichier:  
        fichier.readline()  
        dimensions = fichier.readline() # dimensions est la ligne 2 du fichier  
        dim = dimensions.split()      # transforme la chaine en liste  
        hauteur, largeur = int(dim[0]), int(dim[1])  
        fichier.readline()  
        tableau = [[[0]*3 for j in range(largeur)] for i in range(hauteur)]  
        for i in range(hauteur):  
            ligne = (fichier.readline()).split()  
            for j in range(largeur):      # largeur = nombre de pixels  
                for k in range(3):      # pour les trois couleurs  
                    pixel[i][j][k] = int(ligne[3*j+k])  
        return tableau
```

Variante à partir de tableau = ... :

```
tableau = []  
for i in range(hauteur):  
    ligne_fichier = fichier.readline()  
    ligne = ligne_fichier.split() # convertit la chaine en liste  
    ligne_tableau = []  
    for j in range(largeur):      # largeur = nombre de pixels  
        pixel = [int(ligne[3*j]), int(ligne[3*j+1]), int(ligne[3*j+2])]   
        ligne_tableau.append(pixel)  
    tableau.append(ligne_tableau)  
return tableau
```

2.

```
def test(pixel):  
    R, G, B = pixel  
    return R-G > 79 and R-B > 79 and abs(G-B) < 20
```

```
def recherche_yeux_rouges(tableau):  
    hauteur = len(tableau)  
    largeur = len(tableau[0])  
    for i in range(hauteur-1):      # on ne teste pas la dernière ligne ni  
        for j in range(largeur-1): # colonne car il faut un bloc 2x2  
            if (test(tableau[i][j]) and test(tableau[i+1][j]) and  
                test(tableau[i][j+1]) and test(tableau[i+1][j+1])):  
                return i,j  
    return None
```

ou

```
def recherche_yeux_rouges(tableau):  
    hauteur = len(tableau)  
    largeur = len(tableau[0])  
    for i in range(hauteur-1):  
        for j in range(largeur-1):  
            yeuxrouges = True  
            for di, dj in [(0,0),(0,1),(1,0),(1,1)]:  
                if not test(tableau[i + di][j + dj]):  
                    yeuxrouges = False  
                break  
            if yeuxrouges:  
                return i,j  
    return None
```