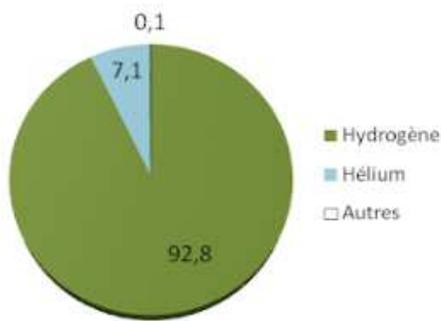
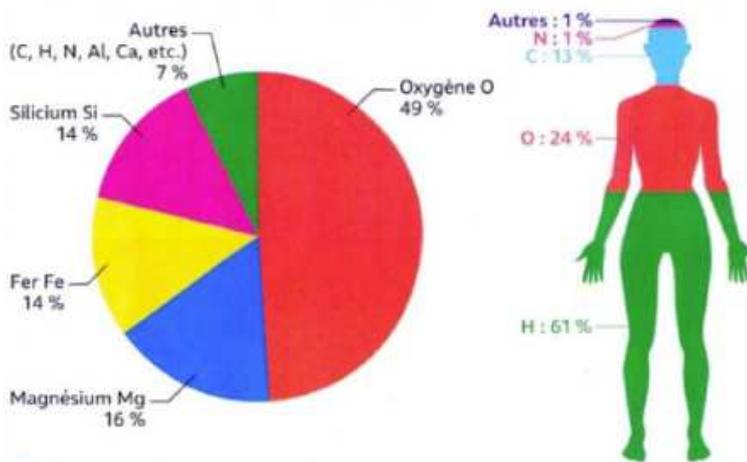


Activité 1 : Les proportions des éléments chimiques

Univers



Terre



Document : pourcentage en quantité de matière des éléments chimiques présents dans l'Univers, sur Terre et dans un être humain.

Questions :

- 1) A l'aide d'un tableur (Excel, Calc...), réaliser un histogramme groupé présentant les proportions des éléments chimiques dans l'Univers, sur Terre et chez un être humain.
- 2) A l'aide d'un tableur (Excel, Calc...), réaliser 3 diagrammes circulaires présentant chacun les proportions des éléments chimiques dans l'Univers, puis sur Terre et enfin chez un être humain.
- 3) A l'aide du langage de programmation Python (Edupython), proposer 3 programmes permettant de représenter 3 diagrammes circulaires présentant chacun les proportions des éléments chimiques dans l'Univers, puis sur Terre et enfin chez un être humain.
- 4) A l'aide du langage de programmation Python (Edupython), proposer un programme permettant de représenter un histogramme présentant les proportions des éléments chimiques dans l'Univers.
- 5) A l'aide du langage de programmation Python (Edupython), proposer un programme permettant de représenter un histogramme groupé présentant les proportions des éléments chimiques dans l'Univers et sur Terre.
- 6) A l'aide du langage de programmation Python (Edupython), proposer un programme permettant de représenter un histogramme groupé présentant les proportions des éléments chimiques dans l'Univers, sur Terre et chez un être humain.

ANNEXE :

Programme python pour faire un diagramme circulaire :

```
import matplotlib.pyplot as plt

labels = 'element1', 'element2', 'element3', 'element4', 'element5'

# Etiquettes de données à modifier

sizes = [10, 10, 10, 10, 10]

# valeurs à modifier

colors = ['yellowgreen', 'gold', 'lightskyblue', 'lightcoral', 'mediumpurple']

plt.pie(sizes, labels=labels, colors=colors,

        autopct='%1.1f%%', shadow=True, startangle=90)

plt.axis('equal')

plt.show()
```

Si on veut mettre en évidence une valeur, ajouter :

```
import matplotlib.pyplot as plt

labels = 'element1', 'element2', 'element3', 'element4', 'element5'

sizes = [10, 10, 10, 10, 10]

colors = ['yellowgreen', 'gold', 'lightskyblue', 'lightcoral', 'mediumpurple']

explode=(0, 0.15, 0, 0, 0)

plt.pie(sizes, explode=explode, labels=labels, colors=colors,

        autopct='%1.1f%%', shadow=True, startangle=90)

plt.axis('equal')

plt.show()
```

Programme python pour faire un histogramme :

```
import matplotlib.pyplot as plt

element = ["element1", "element2", "element3", "element4"]

# Etiquettes de données à modifier

pourcentage = [25, 60, 5, 10]

# valeurs à modifier

plt.bar(element, pourcentage, color = "green", edgecolor='black', linewidth=1.5)

plt.show()
```

Pour un histogramme comparatif de 2 compositions :

```
import matplotlib.pyplot as plt

import numpy as np

element = ["element1", "element2", "element3", "element4"]

pourcentage1 = [25, 60, 5, 10]

pourcentage2 = [30, 45, 10, 15]

# Position sur l'axe des x pour chaque étiquette

position = np.arange(len(element))

# Largeur des barres

largeur = 0.35

# Création de la figure et d'un set de sous-graphiques

fig, ax = plt.subplots()

r1 = ax.bar(position - largeur/2, pourcentage1, largeur, label = 'Planete1', edgecolor = "black",
linewidth = 1)

r2 = ax.bar(position + largeur/2, pourcentage2, largeur, label = 'Planete2', edgecolor = "black",
linewidth = 1)

# Modification des marques sur l'axe des x et de leurs étiquettes

ax.set_xticks(position)

ax.set_xticklabels(element)

plt.legend()

plt.show()
```