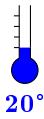


CHAÎNES DE CARACTÈRES

§1. Constructions



Exercice 1 — Écrire les programmes qui construisent les chaînes suivantes :

- 1) "ab ab ab ab ab ab" (avec n fois "ab"),
- 2) "abccccdddeeeee...z" avec 1 fois le "a", 2 fois le "b", etc., jusqu'à 26 fois le "z".



Exercice 2 — Écrire un programme `Grille(n)` qui construit la chaîne représentant un cadre, les bords étant matérialisés par des "*", et l'intérieur constitué de $n \times n$ caractères d'espacement.

```
>>> print(Grille(3))
*****
*   *
*   *
*   *
*****
```



Exercice 3

- 1) Écrire un programme qui étant donnée une chaîne (sans caractères accentués) construit et renvoie la chaîne en bas-de-casse contenant les mêmes caractères, sans les espaces et la ponctuation.

```
>>> Dépouiller("Bonjour le Monde!")
"bonjourlemonde"
```

- 2) En déduire un programme qui teste si une chaîne de caractères est un palindrome. Ce programme renverra un booléen.

```
>>> EstPalindrome("Esopelresteicietse repose.")
True
```

§2. Mots aléatoires



Exercice 4

- 1) Écrire un programme `VoyelleAléa()` qui renvoie une voyelle aléatoire.
- 2) Puis un programme `ConsonneAléa()`.
- 3) En déduire un programme `MotAléa()` qui renvoie un mot aléatoire, avec un nombre aléatoire (mais crédible) de lettres. On veillera à former un mot prononçable, c'est-à-dire qu'on alternera des voyelles et des consonnes.



50°

Exercice 5 — Écrire un programme `PhraseAléa()` qui construit une phrase aléatoire, avec des mots aléatoires, et des signes de ponctuation aléatoires.

```
>>> PhraseAléa()
"Thouraki j' lasc i quol az aij klô bimêts quarki , bla-to m' is trigpa chtu
coul i anê sta sporouskâ!"
```



60°

Exercice 6 — En déduire finalement un programme `TexteAléa()` qui produit un texte aléatoire, formé de paragraphes (de longueurs aléatoires), eux-mêmes formés de phrases aléatoires, etc.. On pourra inclure des dialogues, un titre, des chapitres, et que sais-je d'autre encore.

§3. Lectures

On donne les deux boîtes noires ci-dessous, qui permettent, pour la première, de récupérer le contenu d'un fichier-texte sous la forme d'une chaîne de caractères...

```
1 def Lire(Fichier) :
2     with open(Fichier, "r+b") as f :
3         s = f.read()
4     return s.decode("utf-8")
```

...et, pour la seconde, d'enregistrer dans un fichier-texte le contenu d'une chaîne de caractères.

```
1 def Écrire(Fichier, Texte) :
2     s = Texte.encode("utf-8")
3     with open(Fichier, "w+b") as f :
4         f.write(s)
```

L'argument `Fichier` est la chaîne qui désigne le « nom complet » du fichier (c'est-à-dire son emplacement sur le disque dur, son nom et son extension). Dans le programme `Lire`, si ce fichier n'existe pas, une erreur se produit.



60°

Exercice 7 — Écrire un programme qui étant donnée une chaîne de caractères construit et renvoie la liste de ses mots, en bas-de-casse.

```
>>> Découper("Bonjour, le Monde!")
["bonjour", "le", "monde"]
```



30°

Exercice 8 — Utiliser ce programme pour compter le nombre de mots dans « Notre Dame de Paris » de Victor Hugo.

§4. Statistiques



50°

Exercice 9

- 1) Écrire un programme `Caractères(s)` qui dresse la liste, avec leur fréquence, des caractères utilisés dans une chaîne `s`.
- 2) Quel est le signe le plus utilisé en français ?
- 3) Quelle est la lettre la plus utilisée en français ?



70°

Exercice 10

- 1) Écrire un programme `Mots(s)` qui dresse la liste, avec leur fréquence, des mots utilisés dans une chaîne `s`. On utilisera un dictionnaire pour gérer les mots déjà rencontrés.
- 2) Quel est le mot le plus utilisé en français ?