

NOM :  
Prénom :  
Classe :

Lycée Jeanne d'Arc  
Clermont-Ferrand  
2024 – 2025

## ***Fascicule de SNT (Seconde)***

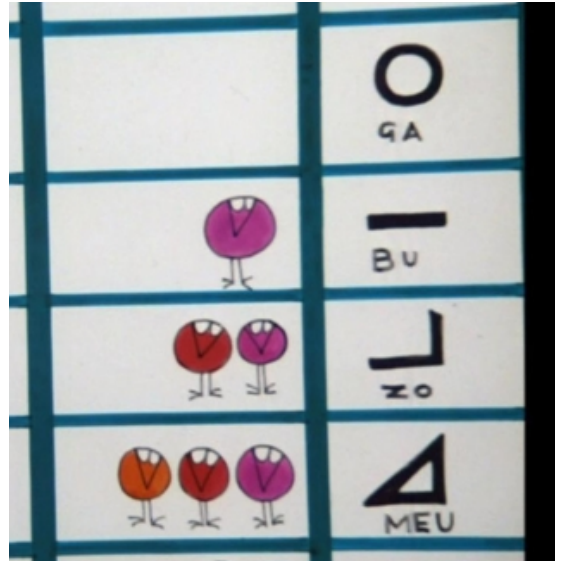
<b>CHAPITRE I : Les systèmes de numération.....2</b>	Exercice 15 : Comparer des valeurs.....11
Exercice 1 : Compter comme un Shadok.....2	Exercice 16 : Convertisseur de température.....11
Exercice 2 : Convertir vers le système binaire.....2	Exercice 17 : La divisibilité d'un nombre.....11
Exercice 3 : Convertir vers le système hexadécimal.....2	Exercice 18 : Je me répète.....11
Exercice 4 : Convertir vers le système décimal.....2	Exercice 19 : Pair ou impair.....11
Exercice 5 : Calculer sans convertir.....2	Exercice 20 : Une séquence d'ADN.....11
<b>CHAPITRE II : Stocker et coder des données.....3</b>	Exercice 21 : Nombre de jours.....11
Exercice 6 : Un proverbe codé.....3	TP 6 : Dessiner un sapin de Noël.....12
Exercice 7 : Blague de robot.....3	TP 7 : Jouer au juste prix.....13
Exercice 8 : Déchiffrer un texte binaire.....4	TP 8 : Jeu de comparaison.....13
Exercice 9 : Coloriage RGB.....4	<b>CHAPITRE V : Internet et le Web.....14</b>
Exercice 10 : Le codage Morse.....4	TP 9 : Simulation du protocole TCP.....14
TP 1 : Le codage des documents texte.....5	TP 10 : Découverte des langages du Web.....14
TP 2 : Les couleurs des pixels.....6	<b>Annexe 1 : Les instructions en Python.....15</b>
TP 3 : Les métadonnées des photos.....7	Les opérations en Python.....15
<b>CHAPITRE III : Traiter des données structurées.....8</b>	Les conditions en Python.....15
TP 4 : Traiter les données des secours français.....8	Les autres mots-clés en Python.....16
TP 5 : Une exploitation forestière.....9	Les fonctions de base en Python.....16
<b>CHAPITRE IV : Le langage Python.....10</b>	Les fonctions de la bibliothèque « random ».....17
Exercice 11 : La calculatrice de Python.....10	Les fonctions de la bibliothèque « math ».....17
Exercice 12 : Les types de variables.....10	Les fonctions de la bibliothèque « time ».....17
Exercice 13 : Mon premier programme.....10	<b>Annexe 2 : Les principales balises en HTML.....18</b>
Exercice 14 : Convertisseur de prix.....10	

# CHAPITRE I : Les systèmes de numération

## Exercice 1 : Compter comme un Shadok

Regardez l'extrait du dessin animé des Shadoks puis répondez aux questions suivantes.

- 1) Quelle est la base du système de numération des Shadoks ?
- 2) Combien valent les nombres suivants dans le système décimal ?  
BU BU = ?  
ZO ZO ZO = ?  
MEU BU ZO ZO GA = ?  
BU GA ZO GA MEU = ?
- 3) Comment prononce-t-on en Shadok les nombres suivants ?  
25 = ?  
73 = ?  
100 = ?  
999 = ?



Extrait du dessin animé des Shadoks

## Exercice 2 : Convertir vers le système binaire

Convertis en binaire et justifie tes réponses par un calcul.

- $77_{10} = ?$   
 $200_{10} = ?$   
 $680_{10} = ?$   
 $345_{10} = ?$   
 $512_{10} = ?$

## Exercice 3 : Convertir vers le système hexadécimal

Convertis en hexadécimal et justifie tes réponses par un calcul.

- $184_{10} = ?$   
 $222_{10} = ?$   
 $546_{10} = ?$   
 $359_{10} = ?$   
 $789_{10} = ?$

## Exercice 4 : Convertir vers le système décimal

Convertis en décimal et justifie tes réponses par un calcul.

- $110\ 011_2 = ?$   
 $10\ 101\ 010_2 = ?$   
 $1\ 011\ 011_2 = ?$   
 $D_{16} = ?$   
 $7AC_{16} = ?$   
 $9B7_{16} = ?$

## Exercice 5 : Calculer sans convertir

1) Effectue les additions de nombres binaires suivantes :

$$1\ 011_2 + 1\ 101_2 = ?$$

$$100\ 101_2 + 11\ 110_2 = ?$$

$$110\ 011_2 + 10\ 101_2 = ?$$

2) Effectue les additions de nombres hexadécimaux suivants :

$$3A_{16} + 7C_{16} = ?$$

$$B9_{16} + 4D_{16} = ?$$

$$1C7_{16} + 8A_{16} = ?$$

Pour vous entraîner à faire des conversions et vérifier vos calculs :  
<https://sebastienguillon.com/test/javascript/convertisseur.html>

# CHAPITRE II : Stocker et coder des données

## Exercice 6 : Un proverbe codé

1) À l'aide de la table ASCII ci-dessous, déchiffrez le texte suivant :

43 68 6F 73 65 20 70 72 6F 6D 69 73 65 2C 20

2) Complétez la phrase précédente en chiffrant le texte suivant en hexadécimal :

chose due.

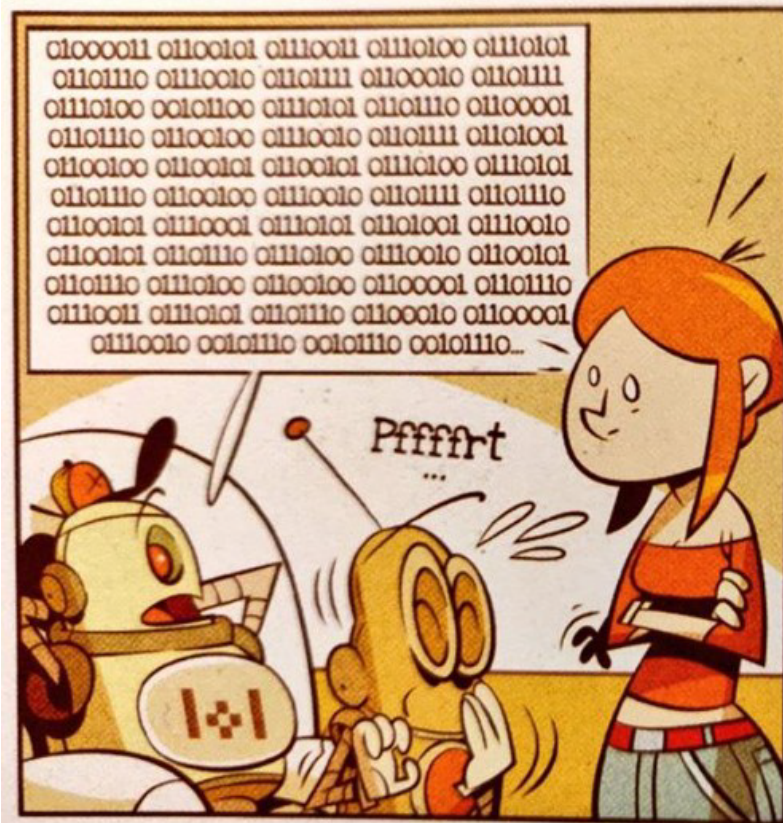
	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
0	NUL	SOH	STH	ETH	EOT	ENQ	ACK	BEL	BS	HT	LF	VT	FF	CR	SO	SI
1	DLE	DC1	CD2	DC3	DC4	NAK	SYN	ETB	CAN	EM	SUB	ESC	FS	GS	RS	US
2	spc	!	"	#	\$	%	&	'	(	)	*	+	,	-	.	/
3	0	1	2	3	4	5	6	7	8	9	:	;	<	=	>	?
4	@	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O
5	P	Q	R	S	T	U	V	W	X	Y	Z	[	\	]	^	_
6	`	a	b	c	d	e	f	g	h	i	j	k	l	m	n	o
7	p	q	r	s	t	u	v	w	x	y	z	{		}	~	DEL

**La table de caractères ASCII (codage hexadécimal)**

« spc » est le caractère d'espace

## Exercice 7 : Blague de robot

Traduisez cette blague (Rob Niveau 3, James, Boris Miroir, éd. Delcourt, p. 56). Regroupez les chiffres par groupes de 4 pour convertir les nombres binaires directement dans le système hexadécimal. Utilisez ensuite la table ASCII de l'exercice précédent pour décoder les caractères.



### Exercice 8 : Déchiffrer un texte binaire

Déchiffre le texte suivant en utilisant une table ASCII. Que remarques-tu de singulier dans ce message ?

```
01000010 01110010 01100001 01110110 01101111 00101100 00100000 01110100
01110101 00100000 01100001 01110011 00100000 01110000 01110010 01100101
01110011 01110001 01110101 01100101 00100000 01110100 01101111 01110101
01110100 00100000 01110100 01110010 01101111 01110101 01110110 11101001
00101110 00101110 00101110
```

### Exercice 9 : Coloriage RGB

Identifie chacune des couleurs ci-dessous, puis colorie le cadrillage ci-contre en respectant le code de chaque case.

- 0 = rgb(0,0,0)
- 1 = rgb(255,255,255)
- 2 = rgb(255,0,0)
- 3 = rgb(0,255,0)
- 4 = rgb(0,0,255)
- 5 = #FFFF00
- 6 = #FF00FF
- 7 = #00FFFF
- 8 = #808080
- 9 = #8080FF

1	1	1	1	1	5	5	5	5	5	5	1	1	1	1	
1	1	1	5	5	5	5	5	5	5	5	5	1	1	1	
1	1	5	5	5	5	5	5	5	5	5	5	5	1	1	
1	5	5	5	5	5	5	5	5	5	5	5	5	5	1	
1	5	5	4	4	4	4	5	5	4	4	4	4	5	5	1
1	5	5	4	0	1	4	5	5	4	0	1	4	5	5	1
1	5	5	4	0	0	4	5	5	4	0	0	4	5	5	1
1	5	5	4	4	4	4	5	5	4	4	4	4	5	5	1
1	5	5	5	5	5	5	5	5	5	5	5	5	5	5	1
1	5	5	5	5	5	5	5	5	5	5	5	5	5	5	1
1	5	5	5	2	5	5	5	5	5	5	2	5	5	5	1
1	5	5	2	2	2	1	1	1	1	2	2	2	5	5	1
1	5	5	2	2	2	2	2	2	2	2	2	2	5	5	1
1	1	5	5	2	2	2	9	9	2	2	2	5	5	1	1
1	1	1	5	5	2	9	9	9	9	2	5	5	1	1	1
1	1	1	1	5	5	5	5	5	5	5	5	1	1	1	1

### Exercice 10 : Le codage Morse

Le codage Morse, développé dans les années 1830 par Samuel Morse et Alfred Vail, est une méthode de transmission d'informations utilisant des séquences de signaux courts et longs. Historiquement, il joua un rôle crucial dans la communication à longue distance, notamment en télégraphie, et marqua les prémices de la communication codée. Il s'agit d'un précurseur des codages binaires plus récents comme l'ASCII.

Un tiret (signal long) est trois fois plus long qu'un point (signal court). Le silence entre deux éléments d'une même lettre dure aussi longtemps qu'un point.

Le silence entre deux lettres dure aussi longtemps qu'un tiret. Le silence marquant une espace entre deux mots est 7 fois plus long qu'un point, et il se note avec une barre oblique (/).

A	• —	U	• • —
B	— • • •	V	• • • —
C	— • — •	W	• — • —
D	— • •	X	— • • —
E	•	Y	• • — • —
F	• • — •	Z	— • — • •
G	— — •		
H	• • • •		
I	• •		
J	• — — —		
K	— • — —		
L	• — • •	1	• — — — —
M	— —	2	• • — — —
N	— •	3	• • • — —
O	— — —	4	• • • • —
P	• — — •	5	• • • • •
Q	— • — •	6	— • • • •
R	• — • •	7	— — • • •
S	• • •	8	— — — • •
T	—	9	— — — — •
		0	— — — — —

- 1) Chiffrez le message suivant en Morse : SOS
- 2) Faites de même avec ce message : PROGRES TECHNOLOGIQUE
- 3) Déchiffrez le message en Morse suivant :  
- . . . - - - - - . . - . . . - . - . . - - - - - / . . . - - . . . / . . . . . . . . . .
- 4) Imaginez une manière simple de coder un message en Morse avec des bits informatiques.
- 5) Codez le message SOS avec votre système de Morse binaire et comparez-le à son codage ASCII binaire.

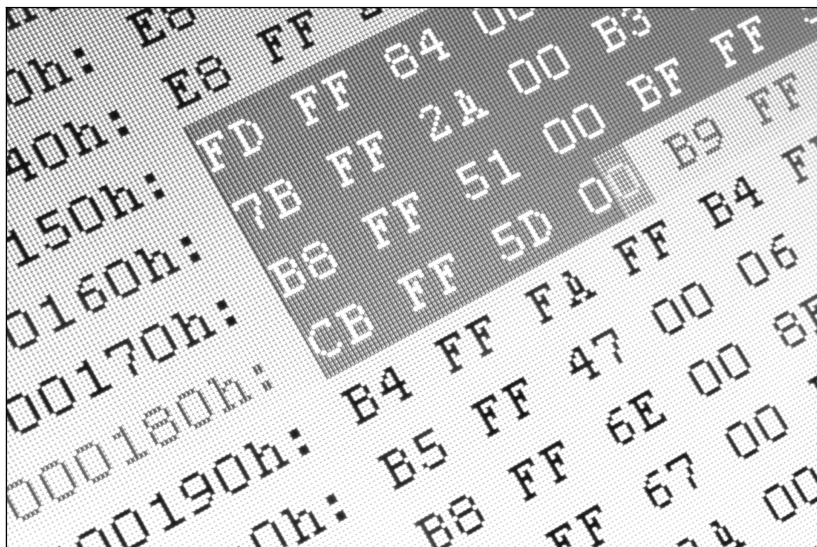
# TP 1 : Le codage des documents texte

Nous allons explorer la manière dont les fichiers sont codés dans un ordinateur.

1) Connectez-vous à l'ENT, puis dans l'espace des classes téléchargez le fichier « bienvenue.zip » sur le bureau. Ce fichier est une archive. Faites un clic droit dessus puis allez dans le menu « 7-zip » pour extraire les fichiers compressés.

2) Pour chacun des trois fichiers que vous avez extraits faites un clic droit et allez dans « Propriétés ». Notez le type de chaque fichier ainsi que sa taille.

3) Ouvrez les trois fichiers avec LibreOffice Writer. Quelles différences constatez-vous ?



4) Rendez-vous sur le site <https://hexed.it/> pour explorer le codage de ces fichiers. Ouvrez les trois fichiers à l'aide de cette application en ligne.

5) Commençons par explorer le fichier « bienvenue.txt ».

À combien de chiffres hexadécimaux correspondent un caractère du texte ?

6) Expliquez le lien entre la longueur de ce texte et la taille du fichier que vous avez notée plus tôt. Un octet est un nombre hexadécimal à combien de chiffres ?

7) Modifiez l'un des nombres hexadécimaux. Que constatez-vous ?

Faites plusieurs essais pour retrouver les nombres correspondant à chacune des 26 lettres en minuscule de l'alphabet.

8) Explorez maintenant « bienvenue.rtf ». Quelle différence constatez-vous avec le fichier précédent ? Retrouvez dans ce code l'emplacement des mots du texte. Quelle est l'utilité du code qui entoure chaque mot ?

9) Explorez enfin « bienvenue.odt ». Pouvez-vous retrouver les mots du texte ? Pourquoi ? À quoi sert un logiciel comme LibreOffice Writer ?

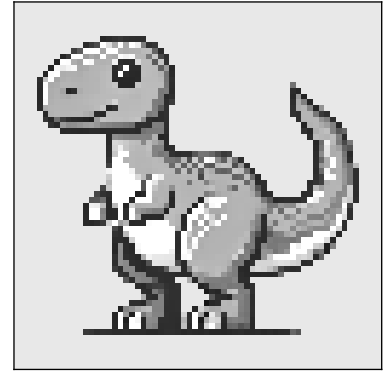
10) Créez un fichier texte (au format TXT) sur le bureau à l'aide d'un éditeur de texte comme Bloc-notes. Nommez ce fichier « LISEZMOI ». Inscrivez dans ce fichier votre nom, votre prénom, votre lycée, votre classe, et l'année scolaire en cours. Placez ce fichier à la racine de votre clé USB. **Recommencez cette question pour l'autre camarade qui compose votre binôme.**



## TP 2 : Les couleurs des pixels

1) Constituez un binôme, allumez l'ordinateur et ouvrez une session. Sur le bureau, cliquez sur « Ordinateur » et naviguez pour accéder au fichier suivant, puis copiez-le sur le bureau :

P:\Diffusion\SNT\PENNEQUIN\dino.bmp



2) Ouvrez ce fichier avec Paint. Quelles sont ses dimensions ? Sachant que chaque pixel est codé par 3 octets successifs, combien d'octets devrait comporter ce fichier ? Affichez les propriétés et relevez la taille exacte du fichiers en nombre d'octets. Combien y a-t-il d'octets de différence par rapport à votre prédiction ?

3) Rendez-vous sur le site <https://hexed.it/> pour explorer le codage de ce fichier. Les octets supplémentaires que vous avez calculés dans la question précédente forment une entête au début du fichier avant le code du premier pixel. Vérifiez que cette entête est bien présente, qu'elle a la bonne taille, et repérez les 3 octets du premier pixel.

4) Les 3 octets d'un pixel de couleur sont dans l'ordre Bleu, Vert, Rouge dans un fichier au format BMP. Modifiez la composante verte du code du premier pixel (écrivez 00 à la place de l'octet correspondant) et enregistrez le nouveau fichier sous le nom de « dino2.bmp ». Ouvrez ce nouveau fichier avec Paint. Où se trouve le premier pixel dans l'image ?

5) Modifiez maintenant le code du deuxième pixel avec HexEd.it et enregistrez le nouveau fichier. Ouvrez-le avec Paint. Où se trouve-t-il par rapport au premier ? Les pixels sont-ils regroupés colonne par colonne ou bien ligne par ligne ?

6) Cherchez maintenant le code du premier pixel de la deuxième ligne en partant du bas. Pour cela calculez le nombre total d'octets occupés par la première ligne et ajoutez le nombre d'octets de l'entête du fichier. Pour vous rendre dans le fichier à la position que vous venez de calculer utilisez dans le volet de droite le champ « Aller à » et tapez simplement le nombre décimal correspondant à cette position. Enregistrez et ouvrez le nouveau fichier avec Paint pour vérifier que vous avez bien modifié le bon pixel.

7) L'œil du dinosaure comporte deux pixels clairs. On veut modifier leur couleur en orange. Ouvrez le fichier dans Paint et repérez les coordonnées de ces deux pixels. Calculez le nombre de lignes pleines qui précèdent ce pixel dans le fichier, sachant qu'il faut les compter à partir du bas de l'image et non du haut. Aidez-vous de cette formule simple :

(Nombre de lignes pleines à partir du début du fichier) = 63 – (La coordonnée y du pixel obtenue sur Paint)

Déduisez-en comme précédemment les positions des premiers octets de ces deux points dans le fichier en ajoutant la coordonnée x, en multipliant par 3 puisqu'il y a 3 octets par pixel, et en ajoutant les octets de l'entête. Rendez-vous à ces positions et modifiez la couleur. On rappelle que les 3 octets sont ici dans l'ordre Bleu, Vert, Rouge et que la couleur orange est un mélange de rouge et d'un peu de vert. Enregistrez le fichier et rouvrez-le avec Paint pour vérifier que vous avez modifier les bons pixels.

8) Comparez maintenant l'image sur laquelle vous avez travaillé avec « dino\_compressé.jpg ». La taille de ce fichier est-elle différente ? Qu'en est-il de la qualité de l'image ? Est-il possible de trouver les octets correspondant à chaque pixel individuel avec HexEd.it ?

9) Remplacez au hasard un octet par 00 au milieu du fichier JPG et enregistrez le nouveau fichier. Comparez attentivement l'ancienne image compressée et la nouvelle. Quel effet a eu votre modification ?

## TP 3 : Les métadonnées des photos

---

1) Constituez un binôme, allumez l'ordinateur et ouvrez une session. Sur le bureau, cliquez sur « Ordinateur » et naviguez dans le système de fichiers pour accéder au dossier suivant, puis copiez les fichiers qui s'y trouvent sur le bureau :

P:\Diffusion\SNT\PENNEQUIN\

2) On s'intéresse à « pelican.jpg ». Quel est le nom du fichier ? Quelle est son extension ? Quel type de contenu renferme-t-il ?

3) Faites un clic droit sur le fichier « pelican.jpg » et rendez-vous dans les Propriétés pour consulter les métadonnées.

Quel est la taille du fichier ?

À quelle date et à quelle heure a été effectuée la prise de vue ?

Avec quel appareil photo (modèle et marque) ?

Le flash a-t-il été utilisé ?

4) Notez les coordonnées GPS de la prise de vue, les différents nombres correspondent dans l'ordre aux degrés (°), minutes d'angles (') et secondes d'angles ("). Arrondissez à un seul chiffre après la virgule pour les secondes d'angles.

5) À l'aide du navigateur rendez-vous sur <https://www.google.fr/maps> et rentrez les coordonnées GPS, latitude puis longitude séparés par une virgule. Utilisez un point à la place de la virgule dans les nombres.

Exemple : Coordonnées GPS du lycée Jeanne d'Arc 45°46'37.3",3°5'42.3"

Où a été prise la photographie ?

6) Créez de nouveaux dossiers et réarrangez vos fichiers selon l'arborescence suivante :

```
TP3
|
+--poemes
|  |
|  +--lepelican.txt
|  |
|  +--thenewcolossus.txt
|
+--photos
|
|  +--joconde.jpg
|
|  +--pelican.jpg
|
|  +--statue-de-la-liberte.jpg
```

7) Sélectionnez le dossier TP3 et faites un clic-droit dessus. Allez dans le menu « 7-Zip » pour créer une archive au format ZIP. L'archive est-elle un fichier ou bien un dossier ?

8) Renommez l'archive avec les initiales en majuscules des deux membres de votre binôme. Placez-la sur l'ENT dans les « Dossiers partagés » dans la catégorie « Informatique ». **Vérification du professeur.**

# CHAPITRE III : Traiter des données structurées

## TP 4 : Traiter les données des secours français

L'État français rend publique les données qu'il a développé dans le cadre de ses missions et qu'il juge non sensibles à l'adresse suivante : <https://www.data.gouv.fr>



1) Allez sur le site des données publiques de l'État français. Dans le moteur de recherche de ce site tapez le mot « secours » afin d'obtenir la table de données sur les « Interventions réalisées par les services d'incendie et de secours ». Téléchargez le fichier le plus récent (au format CSV) et placez-le sur le bureau.

2) Ouvrez le fichier avec le logiciel Bloc-notes. Le tableau est-il facilement lisible ? Quel caractère est utilisé pour séparer les colonnes ? S'agit-il du séparateur standard ? Est-il possible de traiter les données avec ce logiciel ?

3) Ouvrez le fichier avec LibreOffice Calc (au moment de l'ouverture, choisissez le jeu de caractères « Latin 3 » et le séparateur approprié).

4) Quels sont les 7 départements où il y a eu le plus d'interventions au total l'année pour laquelle vous avez obtenues des données ? Commencez par repérer le descripteur « Total interventions », sélectionnez la colonne correspondant à ce descripteur, choisissez dans le menu « Données » l'option « Trier... », une fenêtre apparait, faites « Étendre la sélection » pour que l'intégralité des objets soient triés et pas seulement la colonne sélectionnée. Lisez enfin le nom des 7 départements où il y a eu le plus d'interventions.

5) Que remarquez-vous concernant ces départements et leurs numéros ? Cherchez la signification de ces sigles sur Internet et formulez une explication.

6) Quels sont les 7 départements de catégorie C (donc les moins peuplés) ayant connu le plus d'interventions pour des accidents sur les lieux de travail ? Il y a un tri suivant deux critères à effectuer. Cette fois-ci, lorsque la fenêtre apparait lors du tri, utilisez deux clés de tris en les choisissant de manière pertinente pour répondre à la question posée.

7) Parmi les départements où il y a eu des secours en mer, quels sont ceux où il n'y a pas eu de fausse alerte ?

8) Quel est le nombre total d'interventions pour « Secours à victime » sur toute la France ? Vous pouvez utiliser la fonction SOMME du tableur. Combien cela fait-il d'interventions par jour en moyenne (arrondissez le résultat à l'unité) ?

9) Quelle la moyenne du nombre d'interventions pour les accidents de circulation ? Vous pouvez utiliser la fonction MOYENNE du tableur.



# TP 5 : Une exploitation forestière

Dans une exploitation forestière de cerisiers noirs *Prunus serotina*, on veut estimer le volume de bois que ces arbres pourront produire si on les abattaient, mais avant de le faire.





Nous utilisons pour cela un jeu de données contenant des mesures effectuées sur 31 arbres abattus de différentes tailles, et dont le volume de bois est connu.

1) Constituez un binôme, allumez l'ordinateur et ouvrez une session. Sur le bureau, cliquez sur « Ordinateur », puis copiez le fichier suivant sur le bureau :

```
P:\Diffusion\SNT\PENNEQUIN\  
arbres_cerisier_noir.csv
```

2) Ouvrez ce fichier avec LibreOffice Calc. Quelles données contiennent-ils ? Quelles mesures pourraient nous permettre d'estimer le volume de bois avant l'abattage ?

3) Pour tester vos hypothèses précédentes nous allons construire deux graphiques. Allez dans le menu « Insertion » puis « Diagramme » pour créer un premier graphique. Cliquez sur « XY (dispersion) » puis « Terminer ». Déplacez le graphique pour qu'il ne gêne pas la lecture du tableau.

4) Double-cliquez sur votre graphique pour retourner dans ses menus de conception. Tout en haut, cliquez sur le bouton « Plage de données » . Supprimez toutes les séries de données éventuellement déjà présentes et créez une nouvelle série en cliquant sur « Ajouter ». Cliquez sur cette nouvelle série puis sélectionnez « Valeurs X » pour choisir les données à mettre en abscisse. Cliquez sur le petit bouton  à droite de « Plage pour Valeurs X » puis sélectionnez avec la souris les valeurs (et uniquement les valeurs) de la colonne « Hauteur ». De même, pour les ordonnées vous devez sélectionner la colonne « Volume ». Cliquez enfin sur « OK » pour fermer la fenêtre. Cliquez enfin sur le bouton « Type de diagramme »  tout en haut et sélectionnez « Points seuls » pour supprimer les segments reliant les points sur le graphique. Ajoutez un titre au graphique ainsi qu'aux deux axes après avoir cliqué sur le bouton « Titres » .

5) Créez un deuxième graphique en recommençant la procédure décrite dans les deux questions précédentes. Mais cette fois-ci vous devez choisir la colonne « Circonférence » comme données à placer en abscisse.

6) Maintenant que les deux graphiques ont été créés nous voulons tracer deux droites qui passent au milieu de ces deux nuages de points pour estimer dans chaque cas la corrélation entre le paramètre mesuré et le volume de bois. Double-cliquez sur un graphique pour l'éditer, puis faites un clic-droit sur l'un des points pour ouvrir le menu et choisissez « Insérer une courbe de tendance... ». Choisissez un type linéaire, ne forcez pas l'ordonnée à l'origine, mais cochez bien afficher « Afficher l'équation » et « Afficher le coefficient de détermination ( $R^2$ ) », puis fermez la fenêtre en cliquant sur « OK ». Faites de même avec le deuxième graphique.

7) Le coefficient de détermination  $R^2$  estime combien les points sont dispersés et éloignés de la courbe de tendance (0 = points aléatoires ; 1 = points parfaitement alignés). **Dans lequel des deux graphiques  $R^2$  est-il le plus élevé ? Vaut-il mieux se servir de la hauteur ou bien de la circonférence des arbres pour estimer le volume de bois que l'on obtiendra après les avoir abattus ?**

8) À l'aide de l'équation que vous avez obtenue, **calculez le volume que l'on obtiendrait si l'on abattait 100 cerisiers noirs ayant une circonférence moyenne de 40 cm.**



# CHAPITRE IV : Le langage Python

## Exercice 11 : La calculatrice de Python

**Notez sur une feuille les instructions que vous avez tapées correctement sur l'ordinateur.**

1) Dans la console de Python, affectez la valeur de l'année en cours à la variable `date`, affectez votre année de naissance à la variable `naissance`, enfin soustrayez les deux variables pour calculer l'âge que vous avez eu ou aurez cette année lors de votre anniversaire. Enregistrez ce résultat dans la variable `age`.

2) Créez une variable `annee` contenant le nombre de jours dans une année. Créez une variable `jour` contenant le nombre d'heures dans une journée. De même créez une variable `heure` et une variable `minute`. Enfin, calculez grâce à ces variables le nombre de secondes dans une année et enregistrez le résultat dans une nouvelle variable `anneesec`.

3) Utilisez les variables précédemment calculées `age` et `anneesec` pour calculer le nombre de secondes que vous avez déjà vécu.

4) En utilisant le symbole de la division entière (`//`), calculez le nombre d'années entières que l'on peut mettre dans un milliard de secondes. Vous ne devez pas trouver un nombre décimal !

## Exercice 12 : Les types de variables

**Notez sur une feuille les instructions que vous avez tapées correctement sur l'ordinateur.**

1) Dans la console de Python, affectez votre âge à la variable `age` et votre prénom à la variable `prenom`, puis affichez le type de ces deux variables.

2) Définissez une variable `presentation` dans laquelle vous rédigez une phrase de présentation incluant par concaténation les deux variables définies précédemment. Attention à bien convertir le nombre en une chaîne de caractères avec la fonction appropriée.

3) Affichez la longueur de votre phrase de présentation grâce à la fonction adéquate.

4) À l'aide de la fonction appropriée déterminez le numéro ASCII du caractère "a". Sachant que les lettres sont dans l'ordre alphabétique dans la table ASCII, calculez le numéro de la 16<sup>e</sup> lettre de l'alphabet (sans chercher à savoir quelle est cette lettre). Maintenant utilisez ce numéro ASCII pour afficher cette lettre à l'aide de la fonction appropriée.

## Exercice 13 : Mon premier programme

Écrivez un programme `bonjour.py` qui dit bonjour et qui demande successivement de saisir son prénom puis son nom de famille à l'utilisateur, et enfin qui utilise ces informations pour faire une phrase de réponse. Testez votre programme en l'exécutant.

## Exercice 14 : Convertisseur de prix

Cherchez sur internet le taux de change actuel pour convertir des dollars américains en euros. Écrivez un programme `dollareuro.py` qui demande à l'utilisateur quel prix en dollars il souhaite convertir en euros, puis qui effectue cette conversion et l'affiche avec un arrondi à deux chiffres après la virgule.

### **Exercice 15 : Comparer des valeurs**

- 1) Écrivez un programme `nombremin.py` qui demande de saisir deux valeurs et qui affiche ensuite la plus petite des deux valeurs.
- 2) Écrivez un programme `chainemax.py` qui demande de saisir deux chaînes de caractères et qui affiche ensuite la plus grande des deux chaînes (celle qui a le plus de caractères).
- 3) Écrivez un programme `lettrepresente.py` qui demande de saisir une lettre puis qui demande de saisir un texte, et enfin qui cherche si cette lettre est présente dans le texte ou non. La réponse est alors affichée pour l'utilisateur.

### **Exercice 16 : Convertisseur de température**

Cherchez sur Internet la formule permettant de convertir des degrés Celsius en Fahrenheit et inversement. Écrivez un programme `temperature.py` qui demande à l'utilisateur s'il veut convertir une température de Celsius en Fahrenheit ou l'inverse. En fonction du choix, le programme demande ensuite la température à convertir, effectue la conversion et affiche le résultat avec un arrondi à deux chiffres après la virgule.

### **Exercice 17 : La divisibilité d'un nombre**

Écrivez le programme `multiplede3.py` qui teste si un nombre est entier ou décimal, puis qui teste dans le premier cas si ce nombre est pair ou impair, puis qui teste dans ces deux cas s'il s'agit ou non d'un multiple de trois. Vous devez afficher un compte-rendu précis de tous ces tests pour l'utilisateur.

### **Exercice 18 : Je me répète**

Écrivez un programme `bureau.py` qui affiche 10 fois « Je dois ranger mon bureau » à l'aide du mot-clé `for`. Faites de même avec le mot-clé `while`.

### **Exercice 19 : Pair ou impair**

Écrivez un programme `pairimpair.py` qui affiche les nombres de 2 en 2 de 0 jusqu'à 100 inclus avec un `for`, puis les nombres de 2 en 2 de 1 jusqu'à 99 inclus avec un `while`.

### **Exercice 20 : Une séquence d'ADN**

Écrivez un programme `sequenceADN.py` qui construit une séquence d'ADN aléatoire. La séquence d'ADN doit avoir une longueur déterminée par l'utilisateur grâce à `input()`. On utilise ensuite une boucle `for` pour ajouter les lettres une par une à une séquence initialement vide. On utilisera la fonction `choice` de la bibliothèque `random` pour choisir la lettre aléatoirement parmi "ATCG".

### **Exercice 21 : Nombre de jours**

Écrivez un programme `joursentredates.py` qui demande à l'utilisateur de saisir deux dates au format `jj/mm/aaaa` et qui calcule le nombre de jours entre ces deux dates (sans tenir compte des années bissextiles et en considérant que les mois ont une durée moyenne de 30 jours).

**Astuce :** Pour extraire une partie d'une chaîne de caractères vous pouvez utiliser les crochets.

```
date = "22/06/1999"  
mois = date[3:5]
```

Ce programme enregistre "06" dans la variable `mois` car le 3 entre crochets est la position du début de la chaîne à extraire (0 renvoie la première position) et le 5 est la fin de la chaîne à extraire que l'on exclut.

# TP 6 : Dessiner un sapin de Noël

Nous allons créer un programme capable de dessiner un joli sapin de Noël, dont la taille est donnée par l'utilisateur.

Exemple de sapin pour une taille de 10 lignes :

```
  ^
  ^^
  ^^^
  ^^^^
  ^^^^^
  ^^^^^^
  ^^^^^^
  ^^^^^^
  ^^^^^^
  ^^^^^^
  ^^^^^^
```

1) Créer un nouveau programme Python sous le nom de `sapinnoel.py` que vous enregistrerez sur votre clé USB.

2) Utiliser la fonction `input()` pour demander à l'utilisateur combien de lignes il souhaite afficher pour le sapin. Enregistrer le résultat dans une variable appelée « ligne ». Convertir cette variable en nombre entier grâce à la fonction `int()`.

3) Nous allons créer une boucle pour afficher une par une les lignes du sapin de Noël. Utiliser pour cela le mot-clé `for`.

4) Dans votre boucle `for` calculer deux variables appelées « chapeau » et « espace » correspondant respectivement au nombre d'accents circonflexes et au nombre d'espaces disposés avant les accents circonflexes pour chaque ligne.

5) Utiliser l'opérateur de la multiplication `*` pour concaténer d'un seul coup plusieurs caractères espace « » et plusieurs caractères circonflexe « ^ » en vous servant des variables « espace » et « chapeau » précédemment calculées (Exemple : `"a"*5` est égal à `"aaaaa"`). Afficher le tout avec la fonction `print()`.

6) Bonus : Créer deux nouveaux programmes qui, sur le même principe, sont capables d'afficher des dessins de croix comme sur les exemples ci-dessous.

Exemple de croix droite avec des branches d'une longueur de 3 plus :

```
  +
  +
  +
+++++++
  +
  +
  +
```

Exemple de croix penchée avec des branches d'une longueur de 3 lettres x :

```
  x   x
   x  x
    x x
     x
    x x
   x  x
  x   x
 x    x
```

## TP 7 : Jouer au juste prix

---

**Nous allons créer un programme capable de faire jouer l'utilisateur au jeu du juste prix. L'objectif sera de deviner le prix du superbe ordinateur ci-contre !**



- 1) Créez un nouveau programme Python sous le nom de « justeprix.py » que vous enregistrerez sur le bureau.
- 2) Dans votre programme commencez par créer une variable appelée « prix » et égale à un nombre entier aléatoire entre 300 et 1999.
- 3) Expliquez à l'utilisateur qu'il va devoir deviner le prix d'un ordinateur, celui-ci étant compris entre 300 € et 1999 €. Créez une variable appelée « devinette » et initialement égale à 0. Créez une boucle qui continuera à tourner tant que « prix » et « devinette » ne sont pas égaux.
- 4) Dans la boucle, demandez à l'utilisateur de choisir un nombre entier entre 300 et 1999 et enregistrez le résultat dans « devinette ».
- 5) Testez si « devinette » est inférieure à « prix ». Si c'est le cas, affichez que l'estimation est trop faible. De même, testez si « devinette » est supérieure à « prix ». Dans ce cas, affichez que l'estimation est trop élevée.
- 6) À la fin du programme, affichez un message de félicitations pour l'utilisateur qui a enfin trouvé le juste prix.
- 7) Rajoutez dans votre programme une variable appelée « essai » qui permettra de compter le nombre de fois que l'utilisateur aura fait une estimation avant de réussir à trouver le juste prix et ajoutez un message de fin indiquant ce nombre d'essais.

## TP 8 : Jeu de comparaison

---

**Nous allons créer un jeu éducatif à destination des enfants de 6 à 8 ans (niveaux CP/CE1). L'utilisateur doit comparer la somme de deux nombres de deux chiffres à un troisième nombre.**

- 1) Créez le fichier de votre programme, nommez-le `comparaison.py` et placez-le sur le bureau.
- 2) Commencez par importer le module `random`, puis utilisez la fonction `print()` pour expliquer la règle de ce jeu éducatif : l'utilisateur doit répondre à chaque question avec l'un des 3 symboles suivants : `>`, `<`, ou `=`.
- 3) Initialisez une variable appelée `note` à 0 pour suivre le score, puis créez une boucle `for` qui se répète exactement 10 fois pour pouvoir poser 10 questions.
- 4) Dans cette boucle utilisez la fonction `randint()` pour choisir 3 nombres entiers aléatoires, entre 10 et 50 pour les deux premiers, et entre 20 et 100 pour le 3<sup>e</sup>. Stockez bien ces nombres dans 3 variables différentes.
- 5) Utilisez la fonction `print()` pour afficher un message contenant les 3 nombres, avec le symbole `+` entre le premier et le deuxième, et le symbole `?` entre le deuxième et le troisième. Utilisez ensuite la fonction `input()` contenant le texte « ? doit être remplacé par » pour récupérer la réponse de l'utilisateur.
- 6) Utilisez des séries de conditions pour vérifier si la réponse de l'utilisateur est correcte. Affichez le message « Correct ! » si c'est le cas et augmentez la note de 1, sinon affichez le message « Faux ! ».
- 7) Après la boucle principale du jeu, affichez la note finale obtenue par l'utilisateur.



# CHAPITRE V : Internet et le Web

## TP 9 : Simulation du protocole TCP

**Le silence doit être total. Vous n'avez pas le droit de parler.**

Le but est de transmettre votre message complet à votre destinataire.

Vous ne pouvez transmettre qu'UN SEUL mot à la fois. Si votre message fait 6 mots vous devez donc le découper en 6 paquets. Chaque paquet est un petit morceau de papier.

Vous ne pouvez transmettre vos paquets qu'à la personne se trouvant à votre droite.

Si vous recevez un paquet qui ne vous est pas destiné vous devez le passer à la prochaine personne.

Si vous recevez un paquet qui vous est destiné vous devez émettre un accusé de réception.

Si vous ne recevez pas l'accusé de réception d'un paquet vous devez le renvoyer.

Si vous recevez un paquet en double vous devez à nouveau émettre un accusé de réception.

Le professeur passera dans les rangs pour intercepter des paquets au hasard.

Vous avez perdu si votre destinataire ne parvient pas à reconstituer la totalité du message dans le bon ordre.

Vous perdez aussi si l'expéditeur du message que vous avez reçu n'a pas tous les accusés de réceptions.

**Chaque paquet doit contenir les informations suivantes :**

1) L'expéditeur du paquet  
2) Le destinataire du paquet  
3) Le numéro du paquet  
4) Le nombre total de paquets du message  
5) Une donnée : un seul et unique mot de votre message

OU BIEN

1) L'expéditeur du paquet  
2) Le destinataire du paquet  
3) L'accusé de réception « Bien reçu ! »  
4) Le numéro du paquet dont vous voulez accuser la réception

## TP 10 : Découverte des langages du Web

1) Allez chercher le dossier « TP10 » sur le serveur à l'adresse suivante, puis copiez-le sur votre bureau :

P:\Diffusion\SNT\PENNEQUIN\

2) Repérez et corrigez les erreurs de codage et de mise en page dans les fichiers de ce site Web :

- Le bandeau de l'entête de la page d'accueil n'est pas coloré en vert comme il le devrait.
- Les titres des pages ne sont pas centrés.
- L'image de la page d'accueil ne s'affiche pas.
- Le lien vers la page du parc des Écrins dans la page d'accueil est mort.
- La page de contact n'est pas du tout mise en forme.
- Sur la page du parc des Écrins le paragraphe sur les activités n'utilise pas la bonne mise en forme.

3) Changez la police du site, utilisez par exemple « Arial » ou « Times New Roman ».

4) Créez une toute nouvelle page pour l'un des parcs nationaux et insérez un lien vers celle-ci depuis « index.html ».

# Annexe 1 : Les instructions en Python

## Les opérations en Python

Opérateur	Rôle	Exemple
=	<b>Affecte</b> une valeur à une variable	nbeleve = 36
#	Insère un <b>commentaire</b>	nbeleve = 36 # Ceci n'est pas du code
"	Encadre une chaîne de <b>caractères</b>	appreciation = "Très bien !"
+	<b>Additionne</b> deux nombres ou bien <b>concatène</b> deux chaînes de caractères	somme = 13 + 27 presentation = "Je m'appelle " + prenom
-	<b>Soustrait</b> deux nombres	presents = nbeleve - absents
/	<b>Divise</b> un nombre par un autre	vitesse = distance / temps
*	<b>Multiplie</b> deux nombres ou bien <b>concatène</b> une chaîne de caractères avec elle-même le nombre de fois indiqué	airerectangle = longueur * largeur rire = "ha" * 3 # Enregistre "hahaha"
( et )	<b>Priorité</b> des opérations	moyenne = (note1 + note2) / 2
[ et ]	<b>Extrait</b> une partie d'une chaîne de caractères	chaîne = "science" chaîne[2] # Renvoie "i" chaîne[1:4] # Renvoie "cie" chaîne[4:] # Renvoie "nce" chaîne[:5] # Renvoie "scien"
//	Donne le <b>résultat</b> de la <b>division entière</b>	part = 15 // 4 # Donne 3
%	Donne le <b>reste</b> (modulo) de la <b>division entière</b>	Reste = 30 % 7 # Donne 2
**	Élève un nombre à la <b>puissance</b> d'un autre nombre	airecube = 2 ** 3 # Donne 8

## Les conditions en Python

Opérateur	Rôle	Exemple
if	Exécute les <i>instructions indentées</i> qui suivent <b>seulement si la condition testée est vraie</b>	a = 5 b = 5 c = 7
==	Teste si deux valeurs sont <b>égales</b> (ou identiques)	if a == b : # Ceci sera affiché print("a et b sont égaux")
!=	Teste si deux valeurs sont <b>différentes</b>	if a != c : # Ceci ne sera pas affiché print("a et c sont différents")
>	Teste si une variable est <b>strictement supérieure</b> à une autre	if age > 17 : print("vous êtes majeur")
>=	Teste si une variable est <b>supérieure ou égale</b> à une autre	if age >= 18 : print("vous êtes majeur")
<	Teste si une variable est <b>strictement inférieure</b> à une autre	if age < 18 : print("vous êtes mineur")
<=	Teste si une variable est <b>inférieure ou égale</b> à une autre	if age <= 17 : print("vous êtes mineur")
else	Exécute les <i>instructions indentées</i> qui suivent <b>seulement si la condition testée par if est fausse</b>	if note >= 14 : print("Bien")
elif	Exécute les <i>instructions indentées</i> qui suivent <b>seulement si la condition testée par if est fausse et que la condition testée après elif est vraie</b>	elif note >= 10 : print("Passable") else : print("Échec")

## Les autres mots-clés en Python

Mot-clé	Rôle	Exemple
<b>and</b>	Donne un résultat <b>vrai</b> si et seulement si <b>les deux valeurs sont vraies</b>	<pre>a = 5 b = 5 c = 2 d = 2</pre>
<b>or</b>	Donne un résultat <b>faux</b> si et seulement si <b>les deux valeurs sont fausses</b>	<pre>if a == b and c == d :     print("Ceci s'affiche") if a &lt; c or b &gt; d :     print("Ceci s'affiche aussi")</pre>
<b>in</b>	Teste l' <b>appartenance</b> d'une valeur à une liste ou bien d'un caractère à une chaîne de caractères	<pre>5 in range(10) # Ceci est vrai "e" in "doigt" # Ceci est faux</pre>
<b>for</b>	Exécute en <b>boucle</b> les <i>instructions indentées</i> qui suivent <b>pour chaque valeur de la liste cible</b>	<pre>for i in range(5) :     print(i ** 2) # Affiche 0, 1, 4, 9, 16</pre>
<b>while</b>	Exécute en <b>boucle</b> les <i>instructions indentées</i> qui suivent <b>tant que la condition testée reste vraie</b> (testée à chaque début de boucle)	<pre>i = 3 while i &lt; 5 :     print(i) # La boucle n'est     i = i + 1 # exécutée que 2 fois</pre>
<b>from</b> et <b>import</b>	Permet d'importer les fonctions d'un <b>module</b>	<pre># Importe toutes les fonctions # de la bibliothèque random from random import *</pre>
<b>def</b>	Permet de définir une nouvelle <b>fonction personnalisée</b>	<pre>def cubemoinscarre(mavariabale) :     x = mavariabale**3 - mavariabale**2     return x print(cubemoinscarre(3)) # Affiche 18</pre>
<b>return</b>	<b>Termine</b> l'exécution d'une fonction et <b>renvoie</b> une valeur	

## Les fonctions de base en Python

Fonction	Rôle	Exemple
<b>type()</b>	Renvoie le <b>type</b> de la variable (nombre entier, nombre décimal, chaîne de caractères, etc.)	<pre>nbeleve = 36 type(nbeleve) # Renvoie "int"</pre>
<b>print()</b>	<b>Affiche</b> une valeur sur la console (nombre ou bien chaîne de caractères)	<pre>print("Bonjour") # Affiche "Bonjour"</pre>
<b>str()</b>	<b>Convertit</b> un nombre en une <b>chaîne de caractères</b>	<pre>k = str(36) # Enregistre la chaîne "36"</pre>
<b>int()</b>	<b>Convertit</b> une chaîne de caractères ou un nombre décimal en un <b>nombre entier</b>	<pre>nb = int(36.5) # Donne le nombre 36</pre>
<b>float()</b>	<b>Convertit</b> un nombre entier ou une chaîne de caractères en un <b>nombre décimal</b>	<pre>nb = float(36) # Enregistre 36.0 nb = float("36") # Donne aussi 36.0</pre>
<b>len()</b>	Donne la <b>longueur</b> d'une chaîne de caractères	<pre>longueur = len("chien") # Enregistre 5</pre>
<b>chr()</b>	Renvoie le <b>caractère</b> d'une valeur décimale ASCII ou Unicode donnée	<pre>car = chr(65) # Enregistre "A"</pre>
<b>ord()</b>	Renvoie la <b>valeur décimale ASCII ou Unicode</b> d'un caractère	<pre>num = ord("A") # Enregistre 65</pre>
<b>round()</b>	<b>Arrondit</b> un nombre décimal à n chiffres après la virgule	<pre>resultat = round(8.36,1) # Donne 8.4</pre>
<b>input()</b>	Demande à l'utilisateur de <b>saisir</b> un texte au clavier (renvoie toujours une chaîne de caractères)	<pre>reponse = input("Quel âge as-tu ? ") # Enregistre la chaîne "15" # et pas le nombre 15</pre>
<b>range()</b>	Renvoie une <b>liste de n nombres entiers</b> de 0 jusqu'à n-1	<pre>range(5) # Donne la liste des cinq # nombres suivants 0, 1, 2, 3, 4</pre>
<b>replace()</b>	<b>Remplace</b> une chaîne de caractère par une autre	<pre>txt = "j'aime les bananes" x = txt.replace("bananes", "pommes") print(x) # Affiche "j'aime les pommes"</pre>

### Les fonctions de la bibliothèque « random »

Fonction	Rôle	Exemple
<b>random()</b>	Renvoie un <b>nombre décimal aléatoire</b> entre 0 et 1	<code>nb = random()</code> # Donne par exemple 0,386452114529...
<b>randint()</b>	Renvoie un <b>nombre entier aléatoire</b> entre deux bornes	<code>nb = randint(1,6)</code> # Donne par exemple 5
<b>choice()</b>	Renvoie aléatoirement <b>l'un des caractères</b> d'une chaîne	<code>lettre = choice("abcdefghij")</code> # Donne par exemple "h"

### Les fonctions de la bibliothèque « math »

Fonction	Rôle	Exemple
<b>pi</b>	Constante renvoyant le <b>nombre <math>\pi</math></b>	<code>x = pi</code> # Enregistre 3.14159... dans x
<b>cos()</b>	Renvoie le <b>cosinus</b> d'un angle exprimé en radians	<code>x = cos(pi/3)</code> # Enregistre 0.5
<b>sin()</b>	Renvoie le <b>sinus</b> d'un angle exprimé en radians	<code>x = sin(pi/6)</code> # Enregistre 0.5
<b>tan()</b>	Renvoie la <b>tangente</b> d'un angle exprimé en radians	<code>x = tan(pi/4)</code> # Enregistre 1
<b>sqrt()</b>	Renvoie la <b>racine carrée</b> d'un nombre positif	<code>x = sqrt(9)</code> # Enregistre 3

### Les fonctions de la bibliothèque « time »

Fonction	Rôle	Exemple
<b>sleep()</b>	Met l'exécution du programme en <b>pause</b>	<code>sleep(3)</code> # On attend 3 secondes

# Annexe 2 : Les principales balises en HTML

Balise	Rôle
<html>	Racine du document HTML (tout le document doit être inclus dans cette balise)
<head>	Métadonnées de la page, non affichées à l'écran
<title>	Titre à afficher dans l'onglet du navigateur, pas dans la page
<meta />	Métadonnées, par exemple le jeu de caractères à utiliser pour afficher correctement la page <meta charset="utf-8">
<link />	Lien vers des fichiers externes, comme des feuilles de style CSS <link rel="stylesheet" href="styles/monstyle.css" />
<script>	Lien vers du code JavaScript <script type="text/javascript" src="scripts/code.js"></script>
<body>	Contenu de la page qui sera affiché
<h1>	Titre de la page
<h2>	Titre d'une section
<h3>	Sous-titre d'une section
<ul>	Liste non ordonnée
<ol>	Liste ordonnée
<li>	Élément d'une liste
<p>	Paragraphe
<em>	Texte important (italique par défaut)
<strong>	Texte très important (gras par défaut)
 	Saut de ligne
<a>	Lien hypertexte <a href="https://www.exemple.com">Texte du lien</a>
<img />	Image 
<button>	Bouton cliquable
<table>	Tableau
<tr>	Ligne d'un tableau
<td>	Cellule d'une ligne d'un tableau
<header>	Entête de la page
<footer>	Pied de page