

TRAVAUX PRATIQUES
La programmation en Python
T.P. n°4 : Méthodes sur les chaines de caractères

BTS CIEL
2 ^{ère} année
Page 1 sur 2

Rappel

```

maChaine = "" # 1 : Création d'une chaîne vide
print(type(maChaine)) # 2 : Affichage du type de la variable maChaine
maChaine = str() # 3 : Autre méthode de création d'une chaîne vide
print(type(maChaine)) # 4 : Affichage du type de la variable maChaine
maChaine = "Bonjour les BTS CIEL !" # 5 : Affectation d'une chaîne à la variable maChaine
print(type(maChaine)) # 6 : Affichage du type de la variable maChaine
print(maChaine) # 7 : Affichage du contenu de la variable maChaine
print(maChaine.upper()) # 8 : Mettre en majuscules le contenu de la variable maChaine
print(maChaine.lower()) # 9 : Mettre en minuscules le contenu de la variable maChaine
print(maChaine.capitalize()) # 10 : Mettre en majuscule la première lettre de la variable
print(maChaine.title()) # 11 : Mettre en majuscule la première lettre de chaque mot
print(len(maChaine)) # 12 : Afficher la longueur (le nombre de caractères)
print(maChaine[6]) # 13 : Afficher le 6ème caractère de la variable
print(maChaine[:7]) # 14 : Afficher les caractères du 1er au 6ème
print("#####") # 15 : Affichage des # pour séparer les sections

uneChaine = "Python" # 17 : Affectation de la variable uneChaine à "Python"
for element in uneChaine: # 18 : Parcourir la variable uneChaine caractère par caractère
    print(element) # 19 : Afficher les caractères un par un
print("#####") # 20 : Affichage des # pour séparer les sections

i = 0 # 22 : Création et initialisation de la variable i
while i < len(uneChaine): # 23 : Tant que i est à la longueur de la chaîne
    print(uneChaine[i]) # 24 : Affiche le caractère à la position i (Premier = 0)
    i += 1 # 25 : Incrémenter la variable d'itération i
print("#####") # 26 : Affichage des # pour séparer les sections

suffixe = "Joursoir" # 28 : Création d'une variable et lui affecter "Joursoir"
debut = "Bon" # 29 : Création d'une variable et lui affecter "Bon"
salutation = debut + suffixe[:4] # 30 : Concaténation avec les 4 premiers caractères de suffixe
print(salutation) # 31 : Affichage du contenu de la variable salutation
salutation = debut + suffixe[-4:] # 32 : Concaténation de debut avec la partie finale de suffixe
print(salutation) # 33 : Affichage du contenu de la variable salutation
salutation = debut + suffixe[:-5] # 34 : Concaténation
print(salutation) # 35 : Affichage du contenu de la variable salutation
maChaine = "Bonjour tout le monde"
print(maChaine.startswith("Bonjour")) # 37: Résultat : True
print(maChaine.endswith("monde")) # 38: Résultat : True
print(maChaine.isalpha()) # 39: Résultat : True
print(maChaine.isdigit()) # 40: Résultat : False
print(maChaine.isalnum()) # 41: Résultat : False
maChaine = "42"
print(maChaine.zfill(5)) # 43: Résultat : "00042"
maChaine = "Bonjour tout tout le monde"
print(maChaine.count("tout")) # 45: Résultat : 2 Compte le nom d'occurrence

```

TRAVAUX PRATIQUES
La programmation en Python
T.P. n°4 : Méthodes sur les chaînes de caractères

BTS CIEL
2 ^{ème} année
Page 2 sur 2

Exercice 1 : Vérification de format d'adresse IP

Dans le cadre d'une vérification d'adresses IP, crée un programme qui demande à l'utilisateur d'entrer une adresse IP et qui vérifie si chaque octet de l'adresse est compris entre 0 et 255.

Contraintes :

- L'adresse IP doit comporter exactement trois points "." pour séparer les quatre octets.
- Chaque octet doit être un nombre compris entre 0 et 255

Exercice 2 : Validation d'une adresse MAC

Écris un programme qui vérifie si une chaîne de caractères entrée par l'utilisateur correspond au format d'une adresse MAC valide. Le format attendu est XX:XX:XX:XX:XX:XX, où chaque X est un caractère hexadécimal (0-9, A-F).

Contraintes :

- Chaque partie entre les : doit comporter exactement deux caractères.
- Les caractères doivent être des chiffres (0-9) ou des lettres (A-F) majuscules

Exercice 3 : Vérification de nom de domaine

Écris un programme qui demande à l'utilisateur de saisir un nom de domaine et vérifie si celui-ci respecte les règles suivantes :

- Il doit comporter au moins un point (.).
- Chaque partie du nom de domaine (entre les points) doit contenir uniquement des lettres ou des chiffres.
- Le nom de domaine doit se terminer par une extension de 2 à 6 caractères (comme .com, .org, .edu, etc.).

Exercice 4 : Vérification d'adresse email

Écris un programme qui demande à l'utilisateur de saisir une adresse email et vérifie si celle-ci est valide selon les règles suivantes :

- L'email doit contenir un seul symbole @.
- Il doit y avoir au moins un point (.) après le symbole @.
- Le nom de domaine (la partie après le dernier point) doit avoir au moins 2 caractères.