INSA INSTITUT NATIONAL DES SCIENCES APPLIQUÉES TOULOUSE

Leçon sur les listes

Université de Toulouse

Rappel sur ce que vous savez…

- **Maîtriser les aspects syntaxiques du langage**
- **Créer et gérer des variables simples**
  - Identifiez dans un problème les informations qui doivent être mises en variable
- **Utiliser et créer des fonctions utilisateurs**
  - Mécanismes de passage d'arguments
  - Notion de variables locales à une fonction
  - Découper dans un problème, les sous-problèmes *codables* en fonction

Rappel sur ce que vous savez…

- **Conditionner des parties de programme**
  - Alternative simple ou alternative composée
  - Imbrication d'alternatives
  - Exprimer des conditions complexes par condition logique (ET, OU, NOT,…)
  - Repérer dans un problème les parties à conditionner et exprimer les conditions afférentes
- **Comprendre et mettre en place des boucles algorithmiques**
  - Faire la différence entre un *Tant Que* et un *Pour*
  - Repérer dans un problème les éléments qui sont itératifs

Liste, tableaux…Où est le problème ?

- **Limites d'une variable simple :**
  - Ne peut conserver qu'une seule valeur.
  - Dans une gestion de boucle « profondeur » qu'à un pas…
  - ... à moins de créer N variables! Problème si N n'est pas connu
- **La solution :**
  - Créer des *tableaux* = entassement de N variables de même type.
  - Créer des listes = entassement de variables de type différents.
- **En Python :**
  - Pas de notion de tableaux, sauf pour les chaines de caractères (**str**).
  - Généralisation du concept de liste lié à l'aspect « objet » du langage.

Commençons par les chaines

- On a déjà utilisé des str directement (affichage) :

```
Texto = 'Bonjour'
print(Texto,'à toutes et à tous')
```

- ou dans une boucle for

```
Nb_e = 0
Texto = 'Je ramasse les e'
for Car in Texto :
        if (Car == 'e') :
                Nb_e = Nb_e +1
print('Il y a ',Nb_e,' e dans Texto')
```

- En python , il existe des opérateurs, des fonctions et des méthodes pour travailler sur les chaines….

Des méthodes …késako ?

- La notion de méthode est liée à la notion de programmation objet
- C'est une sorte de fonction, associée à un objet (telle qu'une chaine de caractères, par exemple :

```
NewChaine = Texto.upper()
print(NewChaine)
```

```
JE RAMASSE LES E
```

- Nous ne verrons pas plus dans ce cours l'utilisation de ces méthodes….même si je peux les utiliser à l'occasion.
- Nous en resterons à la notion de fonctions et d'opérateurs

Exemples de fonctions

- **Vous en connaissez déjà un certain nombre**
  - Var = **input**(prompt) : prend une chaine et retourne une chaine
  - Valeur = **int**(Var) : transforme la chaine var en valeur entière
- **Une autre fonction classique (et utile !) est celle qui permet de récupérer la longueur d'une chaine :**

```
Longue = len(Texto)
print('Texto fait', Longue, 'caractères')
```

```
Texto fait 16 caractères
```

- **Peu d'autres fonctions (mais beaucoup de méthodes).**

Exemple d'opérateurs

- **On peut « additionner » deux chaines, ce qui en soi n'est pas trivial :**

```
Prenom = 'Jean'
Nom = 'Peuxplus'
Identite = Prenom + ' ' + Nom
print(Identite)
```

```
Jean Peuxplus
```

- **On peut surtout accéder aux différents éléments de la chaine avec des crochets :**

le numéro i est appelé l'indice du caractère dans la chaine

  - **chaine[i] : retourne le $i^{ème}$ caractère de chaine**
  - **chaine[i : j] : retourne une chaine de caractères contenant les caractères i à j-1 de chaine**

Le premier élément d'une chaine a pour indice 0

Exemple utilisation des indices

- **Exemple précédent de recherche de ‘e’**

```
Nb_e = 0
Texto = 'Je ramasse les e'
for Indi in range(0,len(Texto)) :
        if (Texto[Indi] == 'e') :
                Nb_e = Nb_e + 1
print('Il y a ',Nb_e,' e dans Texto')
```

La syntaxe déjà vue :
for Car in Texto
est plus adaptée pour cette application

Application directe 1

- **Ecrire un programme qui demande une phrase à l'utilisateur qui recherche dans la chaine saisie la première fois où il rencontre la lettre 's' – A chaque tour de recherche il affichera la lettre et l'indice trouvé**

- Que se passe t-il si le texte saisie ne comporte pas de 's'?
- Que se passe t-il si l'indice dépasse la longueur de la chaine?

Modification d'une chaine

- **Tout d'abord on peut dupliquer une chaine par simple affectation :**

```
Texto = ' Bonjour les grands '
Texto_bis = Texto
Texto_bis = Texto_bis + ' dupliqué'
print(Texto,'\n',Texto_bis)
```

```
Bonjour les grands
Bonjour les grands dupliqué
```

- **Mais on ne peut pas modifier un élément !**

```
Texto[13] = 'l'
```

```
Texto[13] = 'l'
TypeError: 'str' object does not support item assignment
```

Modification d'une chaine (2)

- **Pour modifier un élément d'une chaine il faut la *refabriquer* :**

```
Long = len(Texto)
Texto = Texto[0:13]+'L'+ Texto[14:Long]
print(Texto)
```

```
Bonjour les gLands
```

- **Technique similaire pour l'insertion de caractères :**

```
Texto = Texto[:13]+'OE'+ Texto[13:]
print(Texto)
```

```
Bonjour les gOELands
```

Faire attention à la syntaxe !
[0:13] est la même chose que [:13] - Les 13 caractères d'indice 0 à 12
Même constat pour les caractères de fin de chaine.

Application directe 2

- Ecrire un programme qui demande une phrase à l'utilisateur puis qui remplace dans cette chaine toutes les lettres 'e' par la lettre 'i'

Récapitulatif sur les chaines

- Une chaine de caractères (string, type str) est un tableau
- Chaque lettre est repérée par un indice qui commence à 0
- La lettre d'indice i par la syntaxe [i]
- Récupération des lettres indicées i à j par la syntaxe [i:j+1]
- On ne peut pas modifier directement un élément particulier d'une chaine
- Sauf cas particulier il est plus pratique de parcourir une chaine à l'aide de la syntaxe for car in chaine