

## Support pour l'apprentissage de la pensée computationnelle

Adrian Holzer & Pascal Felber

Les compétences de base en programmation deviennent de plus en plus importantes et constituent un fondement pour de nombreux cours au sein de toutes les facultés.

Le but du projet est d'autonomiser les étudiant-e-s dans l'apprentissage de concepts de programmation à travers :

- de courtes capsules vidéos
- des ressources interactives
- des exercices auto-évalués.

### Capsules vidéos

4 Courtes capsules vidéos présentant les concepts de base de Python (en cours d'élaboration):

- Les conditions
- Les boucles
- Les listes
- Les fonctions

#### La liste

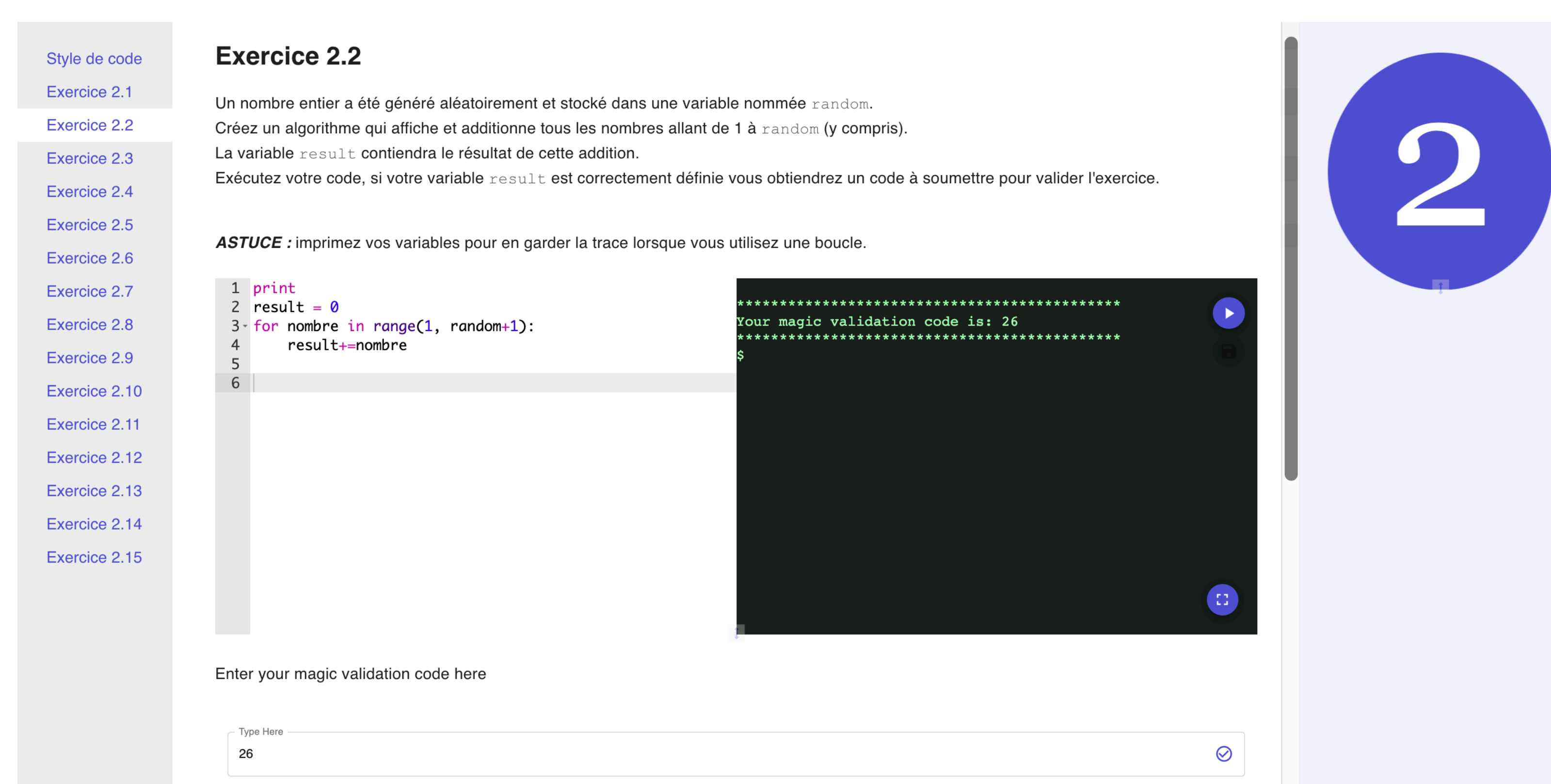
→ Modification

Pour modifier la liste en remplaçant un élément on peut utiliser la même notation avec l'index entre crochet



### Exercices auto-évalués

- 60 exercices de programmation Python auto-évalués sur Graasp
- Exécutable dans le navigateur sans besoin d'installer un logiciel
- Gamification (chaque exercice réussi ajoute des points au score) pour motiver les étudiant-e-s à faire plus d'exercices



### Ressources interactives

- Syllabus en ligne qui présente les bases de la programmation Python en utilisant des carnets interactifs Jupyter Notebook
- Mélange entre texte statique et code exécutable
- Directement accessible en ligne, pas besoin d'installer un logiciel

#### 4.3.1. Accéder aux caractères d'une chaîne

De la même façon que pour les listes, il est possible d'accéder à un ou plusieurs caractères d'une chaîne grâce à leur indice.

```

my_str = "chaîne_de_caractères"
print(my_str[0]) # 1er caractère uniquement
print(my_str[10:]) # tous les caractères à partir du 11e

```

```

c
caractères

```

La figure 4.2 ci-dessous illustre différentes manières d'accéder aux caractères d'une chaîne via leur indice. La chaîne `s` comprend 14 caractères d'indices 0 à 13. On peut accéder à des caractères individuels ou à des plages de caractères (sous-chaînes), comme par exemple `s[-3:]` pour désigner le préfixe composé des trois derniers caractères (équivalent ici à `s[11:]` ou `s[11:14]`).

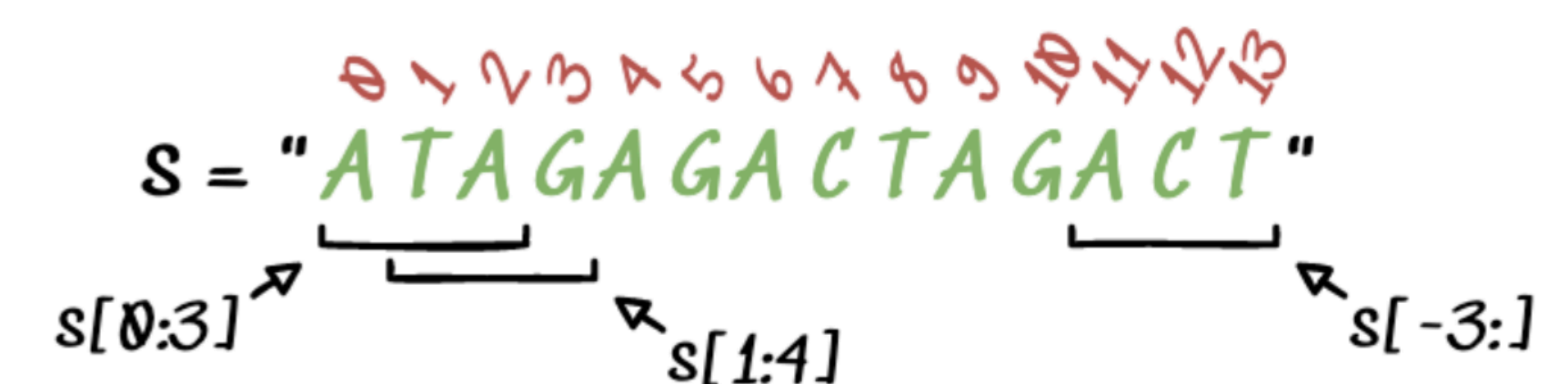


Fig. 4.2 Chaîne de caractères.

<http://members.unine.ch/pascal.felber/teaching/Python/Part-I/html/book/contents.html>  
<http://members.unine.ch/pascal.felber/teaching/Python/Part-II/html/book/contents.html>

### Conclusion

- Ce projet a permis de mettre sur pieds une série de nouvelles ressources qui seront disponibles en ligne pour toute personne intéressée.
- Les vidéos ainsi que les ressources interactives seront les plus faciles à partager et à intégrer dans d'autres cours (ou comme pré-requis) car elles peuvent être parcourues rapidement et il est possible d'en tirer des informations ciblées
- Les exercices auto-évalués sont plus lourds à intégrer dans un cours, car ils viennent en bloc, mais permettent aux étudiant-e-s de pratiquer la programmation de façon intensive mais relativement motivante grâce au feedback de l'auto-correction et du score général.