

.....

Session d'exercices – Les structures de contrôle

Jour de la semaine

1. [Difficulté: *]

Écrivez un programme appelé *week_day_single.py* qui demande un seul chiffre et affiche le nom du jour de la semaine correspondant. Pour un nombre en dehors de l'intervalle [1 – 7], le programme doit afficher le message

```
Number out of range. Retry....
```

Si l'entrée de l'utilisateur ne contient aucun chiffre, le programme doit afficher

```
Invalid number. Retry....
```

Pour obtenir une saisie utilisateur, nous avons vu qu'il faut utiliser la fonction `input()` de Python. Par défaut, cette fonction renvoie la saisie de l'utilisateur sous forme de chaîne de caractères. La méthode Python `isdigit()` permet de vérifier si une chaîne de caractères ne contient que des chiffres. Elle renvoie `True` si tous les caractères sont des chiffres, sinon elle renvoie `False`. Bien sûr, elle est utile uniquement lorsqu'on attend des nombres entiers (integers) supérieurs ou égaux à zéro. Exemples:

```
>>> my_input = '123456' # text/input returned by input()
>>> my_input.isdigit() # checking if that text/input, contains only digits
True
>>> my_input = '12.3456'
>>> my_input.isdigit()
False
# false because the input contained a period
>>> my_input = '-12.3456'
>>> my_input.isdigit()
False
# false because the input contained a period and minus sign
>>> my_input = '-123456'
>>> my_input.isdigit()
False
# false because the input contained minus sign
```

2. [Difficulté: *]

Écrivez un programme appelé `week_day_remaining`. Ce nouveau programme doit aussi demander un seul chiffre `n`, mais cette fois pour afficher les `n` derniers jours de la semaine. De plus, le programme doit vérifier si `n` est dans l'intervalle `[1 - 7]` de la même façon que dans l'exemple précédent.

```
Enter the number of the day [1-7]: 3
Friday
Saturday
Sunday
```

Figure 1: Exemple pour la valeur 3.