

Ce devoir contient deux exercices indépendants. Indiquez sur la copie votre parcours et le numéro de votre groupe. Sauf mention explicite du contraire, toutes les fonctions que vous réaliserez devront comporter une documentation avec contraintes et exemples d'utilisation.

Exercice 1 *Dates en chaînes*

Dans cet exercice, les dates sont représentées par des dictionnaires à trois clés : `'jour'`, `'mois'` et `'annee'` dont les valeurs associées sont des nombres entiers. Ainsi la date de ce jour est représentée par

```
>>> aujourdhui = creer_date(17, 2, 2018)
>>> aujourdhui
{'jour': 17, 'mois': 2, 'annee': 2018}
```

Question 1 En utilisant la méthode `split`, réalisez une fonction qui transforme une chaîne de caractères de la forme `'jj/mm/aaaa'` en une date sous forme d'un dictionnaire.

```
>>> chaine_en_date('17/2/2018')
{'jour': 17, 'mois': 2, 'annee': 2018}
```

Question 2 En utilisant la méthode `join`, réalisez la fonction réciproque de la précédente.

```
>>> date_en_chaine(aujourdhui)
'17/2/2018'
```

Exercice 2 *Jeux olympiques 2018*

Les XXIII-èmes jeux olympiques d'hiver se déroulent actuellement à Pyeongchang en Corée du Sud depuis le 9 février dernier. Au moment de rédiger ce sujet de DS quelques médailles ont été délivrées. Et le but de cet exercice est de gérer les résultats obtenus par pays.

Pour cela on décide de représenter ces résultats sous forme d'un dictionnaire dont les clés sont les pays (sous forme de chaînes de caractères), et les valeurs associées des listes de résultats, chaque résultat étant un couple de la forme (discipline, médaille). Voici un extrait du tableau des médailles tel qu'il est connu au moment de rédiger ces lignes.



```
TABLEAU_DES_MEDAILLES={
  'Allemagne' : [('Luge simple hommes', 'BRONZE')],
  'Corée du Sud' : [('Short track 1500m hommes', 'GOLD')],
  'Pays-Bas' : [('Short track 1500m hommes', 'SILVER'),
                ('Patinage de vitesse 5000m hommes', 'GOLD'),
                ('Patinage de vitesse 3000m femmes', 'GOLD'),
                ('Patinage de vitesse 3000m femmes', 'SILVER'),
                ('Patinage de vitesse 3000m femmes', 'BRONZE')],
  'Autriche' : [('Luge simple hommes', 'GOLD')],
  'Russie' : [('Short track 1500m hommes', 'BRONZE')],
  'Etats-Unis' : [('Luge simple hommes', 'SILVER')],
  'Canada' : [('Patinage de vitesse 5000m hommes', 'SILVER'),
```

```
( 'Ski acrobatique bosses femmes', 'SILVER' ]],  
'Norvège' : [( 'Patinage de vitesse 5000m hommes', 'BRONZE' ]],  
'Kazakhstan' : [( 'Ski acrobatique bosses femmes', 'BRONZE' ]],  
'Jamaïque' : []}]
```

Question 1 Comment utiliser ce tableau pour obtenir

1. le nombre de pays ayant obtenu des médailles ?
2. le nombre de médailles obtenues pas l'Allemagne ?
3. le nombre de médailles d'or obtenues par les Pays-Bas ?
4. le nombre total de médailles délivrées ?

Question 2 Réalisez une fonction `epreuves_courues(tableau)` qui renvoie l'ensemble des épreuves sportives dont les résultats figurent dans `tableau`.

```
>>> epreuves_courues(TABLEAU_DES_MEDAILLES)  
{ 'Patinage de vitesse 3000m femmes', 'Luge simple hommes', 'Short track 1500m hommes', 'Ski acrobatique bosses femmes' }
```

Question 3 Réalisez une procédure `ajouter_resultat(tableau, pays, sport, medaille)` qui a pour effet de bord d'ajouter au tableau des médailles passé en paramètre un résultat pour le pays `pays` dans le sport `sport` et la médaille `medaille`.

```
>>> ajouter_resultat(TABLEAU_DES_MEDAILLES, 'France', 'Ski acrobatique bosses femmes', 'GOLD')  
>>> TABLEAU_DES_MEDAILLES['France']  
[( 'Ski acrobatique bosses femmes', 'GOLD' )]
```

Question 4 Réalisez une fonction `medailles_par_sport(tableau, sport)` qui construit le triplet (`p1`, `p2`, `p3`) de pays médaillés dans le sport `sport`, `p1` étant le pays ayant obtenu la médaille d'or, `p2` la médaille d'argent et `p3` le bronze.

```
>>> medailles_par_sport(TABLEAU_DES_MEDAILLES, 'Luge simple hommes')  
( 'Autriche', 'Etats-Unis', 'Allemagne' )
```

Question 5 Réalisez une fonction `nombre_medailles(tableau, pays)` qui renvoie un triplet (`gold`, `silver`, `bronze`) constitué du nombre de médailles de chaque sorte obtenues par le pays passé en paramètre.

```
>>> nombre_medailles(TABLEAU_DES_MEDAILLES, 'Pays-Bas')  
(2, 2, 1)
```

Question 6 Réalisez une procédure `afficher_rang(tableau)` qui affiche un classement par pays. Par exemple :

```
>>> afficher_rang(TABLEAU_DES_MEDAILLES)  
rang n° 1 : (2, 2, 1) : Pays-Bas  
rang n° 2 : (1, 0, 0) : Autriche, Corée du Sud, France  
rang n° 5 : (0, 2, 0) : Canada  
rang n° 6 : (0, 1, 0) : Etats-Unis  
rang n° 7 : (0, 0, 1) : Allemagne, Kazakhstan, Norvège, Russie  
rang n° 11 : (0, 0, 0) : Jamaïque
```

remarque 1 : Le classement des triplets de nombres de médailles est dans l'ordre lexicographique inverse. Cela peut être obtenu en précisant `reverse=True` comme argument de la méthode `sort` appliquée une liste de triplets.

remarque 2 : Les pays qui possèdent exactement les mêmes nombres de médailles sont affichés dans l'ordre alphabétique et sont séparés par une virgule suivie d'un espace.

remarque 3 : Le numéro du rang tient compte du nombre total de pays mieux classés.