

▼ SNT - TP Python 1 : Modélisation d'un réseau social



Pour commencer, exécutez le script suivant pour importer deux modules qui nous seront nécessaires pendant tout le TP.

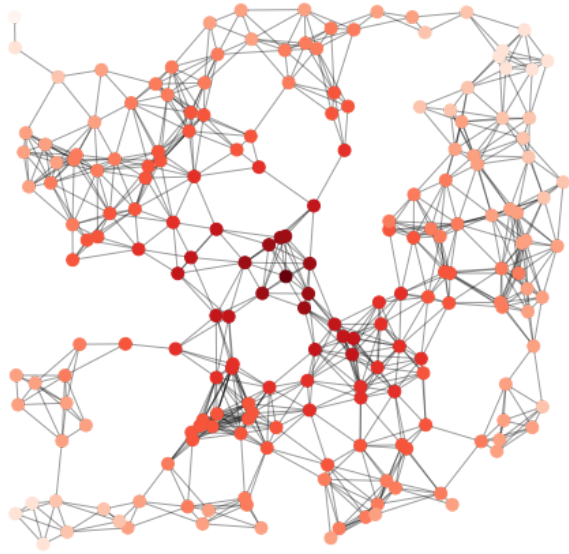
```
1 # importation du module "matplotlib" pour créer des graphiques
2 import matplotlib.pyplot as plt
3 # importation du module "networkx" pour simuler un réseau social
4 import networkx as nx
```

▼ 1 Introduction

Exécuter plusieurs fois le script Python ci-dessous (sans chercher pour le moment à en comprendre le code) :

```
1 G = nx.random_geometric_graph(200, 0.125)
2 pos = nx.get_node_attributes(G, 'pos')
3 dmin = 1
4 ncenter = 0
5 for n in pos:
6     x, y = pos[n]
7     d = (x - 0.5)**2 + (y - 0.5)**2
8     if d < dmin:
9         ncenter = n
10        dmin = d
11 p = dict(nx.single_source_shortest_path_length(G, ncenter))
12 plt.figure(figsize=(8, 8))
13 nx.draw_networkx_edges(G, pos, nodelist=[ncenter], alpha=0.4)
14 nx.draw_networkx_nodes(G, pos, nodelist=list(p.keys()),
15                        node_size=80,
16                        node_color=list(p.values()),
17                        cmap=plt.cm.Reds_r)
18 plt.xlim(-0.05, 1.05)
19 plt.ylim(-0.05, 1.05)
20 plt.axis('off')
21 plt.show()
```





Exercice 1 : Expliquer pourquoi la figure générée nous permet de modéliser un réseau social.

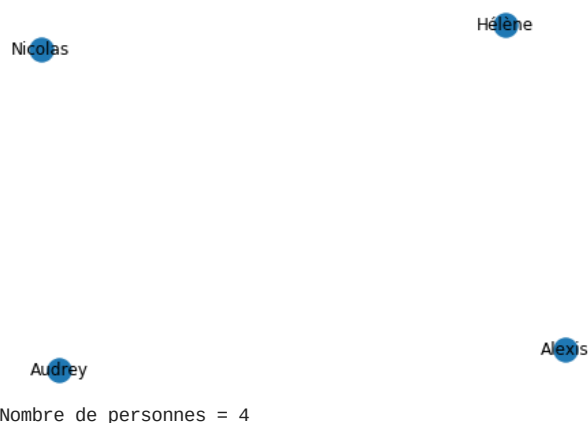
---> votre réponse dans cette cellule...

▼ 2 Découverte du module networkx

Simulons un réseau social avec le module Python networkx.

Lire puis exécuter plusieurs fois le code suivant :

```
1 # création d'un graphe vide non orienté appelé "monReseau"
2 monReseau = nx.Graph()
3
4 monReseau.add_node("Audrey")
5 monReseau.add_node("Nicolas")
6 monReseau.add_node("Alexis")
7 monReseau.add_node("Hélène")
8
9 # dessine (draw) et affiche (show) le graphe "monReseau"
10 nx.draw(monReseau, with_labels=True)
11 plt.show()
12
13 print("Nombre de personnes =", monReseau.number_of_nodes())
```



Exercice 2 : Que fait le code de la ligne 6 : `monReseau.add_node("Alexis")` ?

---> votre réponse dans cette cellule...

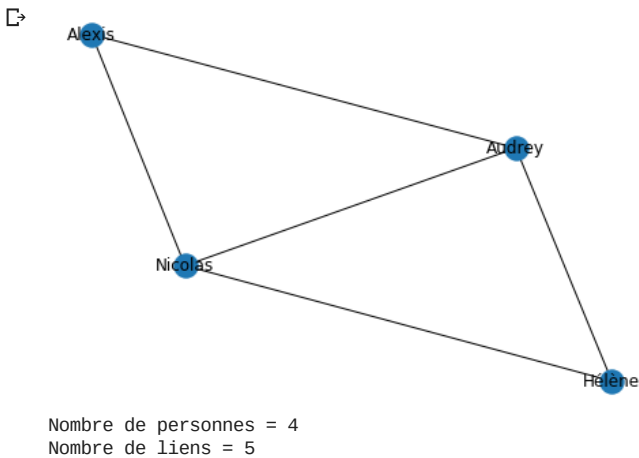
Lire puis exécuter plusieurs fois le code suivant :

```
1 monReseau = nx.Graph()
2
```

```

3 monReseau.add_node("Audrey")
4 monReseau.add_node("Nicolas")
5 monReseau.add_node("Alexis")
6 monReseau.add_node("Hélène")
7
8 monReseau.add_edge("Audrey", "Nicolas")
9 monReseau.add_edge("Audrey", "Alexis")
10 monReseau.add_edge("Nicolas", "Alexis")
11 monReseau.add_edge("Hélène", "Audrey")
12 monReseau.add_edge("Hélène", "Nicolas")
13
14 nx.draw(monReseau, with_labels=True)
15 plt.show()
16
17 print("Nombre de personnes =", monReseau.number_of_nodes())
18 print("Nombre de liens =", monReseau.number_of_edges())

```



Exercice 3 : Que fait le code de la ligne 8 : `monReseau.add_edge("Audrey", "Nicolas")` ?

---> votre réponse dans cette cellule...

Exercice 4 : Quelles sont les deux personnes qui ne sont pas directement en relation ? Que faudrait-il faire pour les mettre en relation ?

---> votre réponse dans cette cellule...

▼ 3 Un graphe orienté

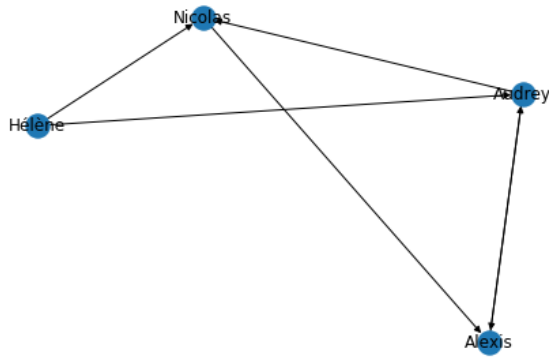
Lire puis exécuter plusieurs fois le code suivant, qui est presque le même que le code précédent :

```

1 monReseau = nx.DiGraph() # création d'un graphe orienté
2
3 monReseau.add_node("Audrey")
4 monReseau.add_node("Nicolas")
5 monReseau.add_node("Alexis")
6 monReseau.add_node("Hélène")
7
8 monReseau.add_edge("Audrey", "Nicolas") # lien de Audrey vers Nicolas
9 monReseau.add_edge("Audrey", "Alexis")
10 monReseau.add_edge("Nicolas", "Alexis")
11 monReseau.add_edge("Hélène", "Audrey")
12 monReseau.add_edge("Hélène", "Nicolas")
13 monReseau.add_edge("Alexis", "Audrey")
14
15 nx.draw(monReseau, with_labels=True)
16 plt.show()
17
18 print("Nombre de personnes =", monReseau.number_of_nodes())
19 print("Nombre de liens =", monReseau.number_of_edges())

```





Nombre de personnes = 4
Nombre de liens = 6

Exercice 5 : Qu'est-ce qui a changé par rapport au graphe précédent ?

---> votre réponse dans cette cellule...

Exercice 6 : Pourquoi la relation entre Audrey et Alexis est-elle différente des autres ? Pourquoi voit-on 5 segments mais 6 liens ?

---> votre réponse dans cette cellule...

Exercice 7 : Ce dernier graphe est dit **orienté**, par opposition au graphe du paragraphe précédent qui est dit **non orienté**. Parmi Facebook, Twitter, Instagram, WhatsApp, Snapchat, citer un réseau social orienté et un réseau social non orienté.

---> votre réponse dans cette cellule...

▼ 4 Mon petit réseau social

Exercice 8 : Choisir six prénoms de votre choix et simuler graphiquement un réseau social **non orienté** entre ces personnes, à l'aide du code appris précédemment.

```

1  # votre code ici
2  ...
3  pass

```

▼ 5 Un réseau social où tout le monde est amis

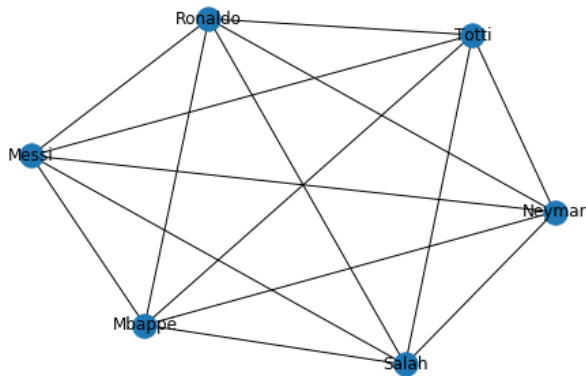
Lire puis exécuter plusieurs fois le programme suivant.

```

1  monRezo = nx.Graph()
2
3  Liste = ["Ronaldo", "Messi", "Totti", "Salah", "Mbappé", "Neymar"]
4
5  for a in Liste:
6      monRezo.add_node(a)
7
8  for b in Liste:
9      for c in Liste:
10         monRezo.add_edge(b, c)
11
12  nx.draw(monRezo, with_labels = True)
13  plt.show()
14
15  print("Nombre de personnes =", monRezo.number_of_nodes())
16  print("Nombre de liens =", monRezo.number_of_edges())

```





Nombre de personnes = 6
Nombre de liens = 15

Exercice 9 : Expliquer ce que font les lignes 8, 9 et 10 du programme.

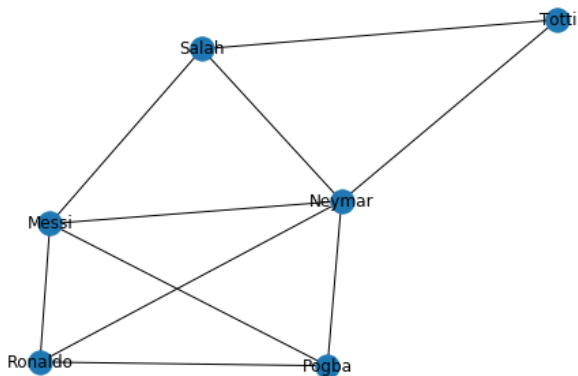
---> votre réponse dans cette cellule...

▼ 6 Un réseau social plus ou moins social

La fonction `random()` renvoie un nombre réel au hasard entre 0 et 1.

Lire puis exécutez plusieurs fois de suite le programme suivant :

```
1  from random import random
2
3  monRezo = nx.Graph()
4
5  Liste = ["Ronaldo", "Messi", "Totti", "Salah", "Pogba", "Neymar"]
6  p = 0.3
7
8  for a in Liste:
9      monRezo.add_node(a)
10
11  for b in Liste:
12      for c in Liste:
13          if random() < p:
14              monRezo.add_edge(b, c)
15
16  nx.draw(monRezo, with_labels = True)
17  plt.show()
18
19  print("Nombre de personnes =", monRezo.number_of_nodes())
20  print("Nombre de liens =", monRezo.number_of_edges())
```



Nombre de personnes = 6
Nombre de liens = 8

Exercice 10 : Par défaut $p = 0.3$ (ligne 6 du programme). Donnez à p la valeur 0.1 puis exécutez plusieurs fois le programme. Donnez à p la valeur 0.9 puis exécutez plusieurs fois le programme. Que constatez-vous ? Pourquoi ?

---> votre réponse dans cette cellule...

▼ 7 D'autres réseaux ?

Bonus : Nous venons de simuler un réseau social avec le module Python `networkx`. Quels autres types de réseaux pourrait-on modéliser avec `networkx` ?

---> votre réponse dans cette cellule...
