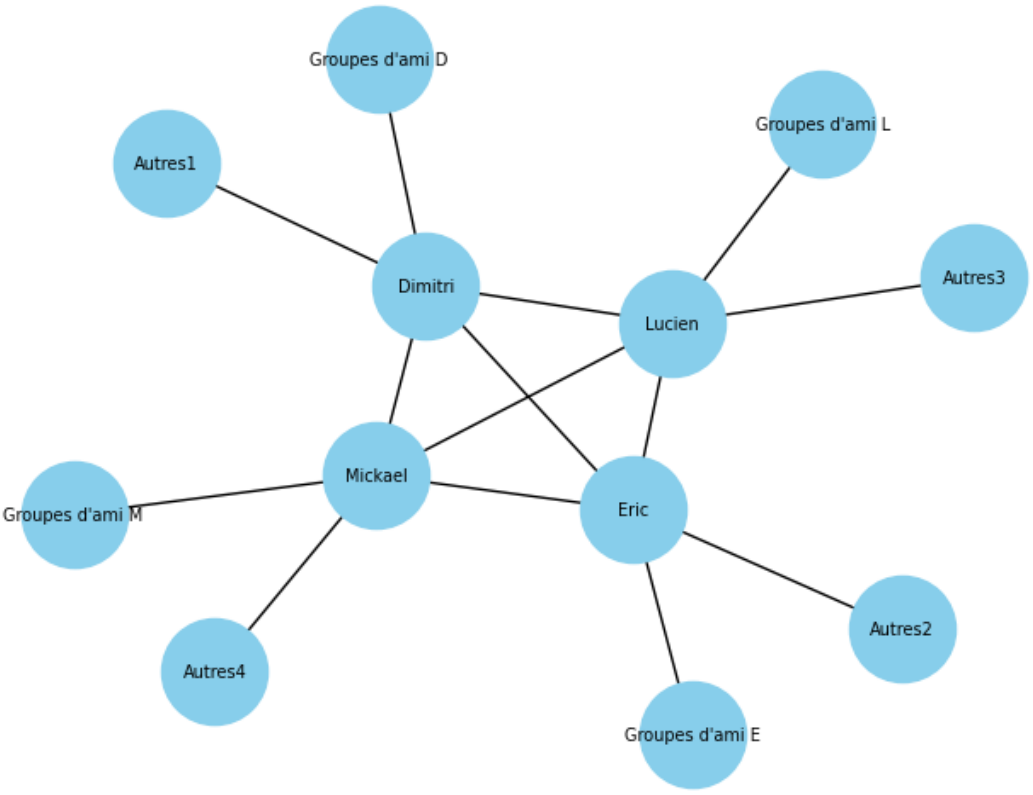
graphe

November 25, 2024

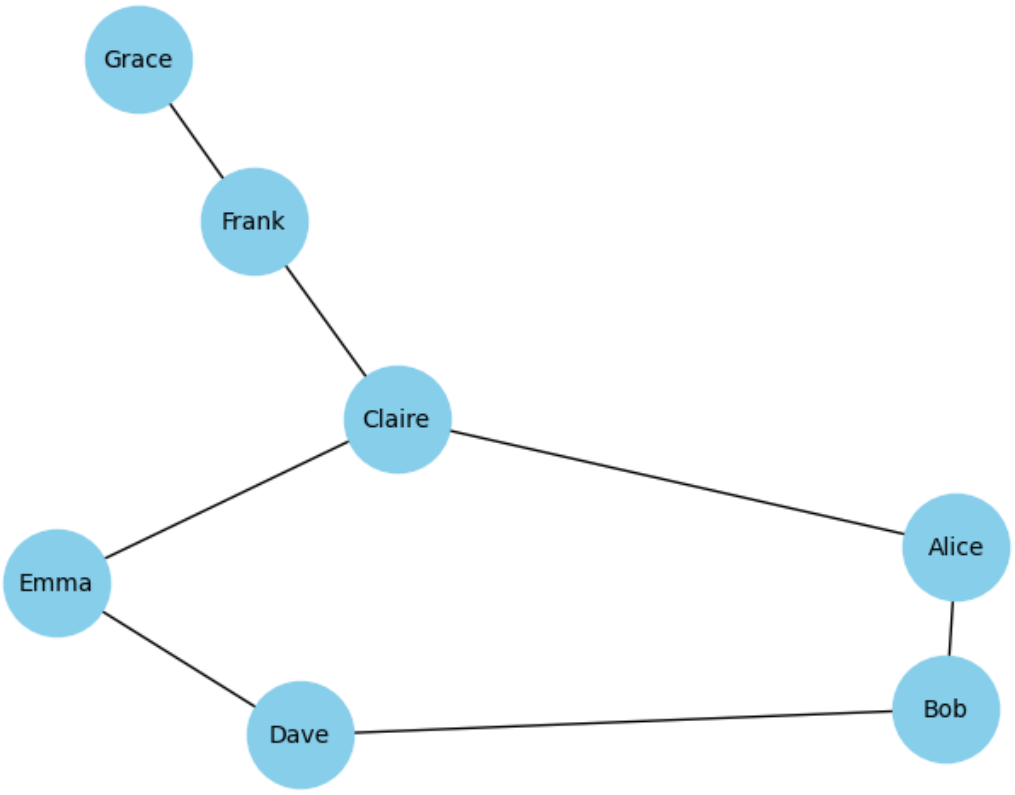
[57]:
```
import networkx as nx
import matplotlib.pyplot as plt

graph = nx.Graph()
nodes = ["Dimitri", "Eric", "Lucien", "Mickael","Autres", "Groupes d'ami D",␣
 ↪"Groupes d'ami E", "Groupes d'ami L", "Groupes d'ami M" ]
edges = [("Dimitri", "Groupes d'ami D"),
    ("Eric", "Groupes d'ami E"),
    ("Lucien", "Groupes d'ami L"),
    ("Mickael", "Groupes d'ami M"),
    ("Dimitri", "Eric"),
    ("Dimitri", "Lucien"),
    ("Dimitri", "Mickael"),
    ("Eric", "Lucien"),
    ("Eric", "Mickael"),
    ("Lucien", "Mickael"),
    ("Dimitri", "Autres1"),
    ("Eric", "Autres2"),
    ("Lucien", "Autres3"),
    ("Mickael", "Autres4")
]
graph.add_edges_from(edges)
nx.draw(graph, with_labels=True, node_color = 'skyblue', node_size = 2000,␣
 ↪font_size = 7)
plt.show()
```



[63]:
```python
import networkx as nx

# Création du graphe
G = nx.Graph()
G.add_nodes_from(["Alice", "Bob", "Claire", "Dave", "Emma", "Frank", "Grace"])
edges = [
    ("Alice", "Bob"),
    ("Alice", "Claire"),
    ("Bob", "Dave"),
    ("Claire", "Emma"),
    ("Dave", "Emma"),
    ("Frank", "Claire"),
    ("Frank", "Grace"),
]

G.add_edges_from(edges)

def trier(L):
    for i in range(len(L)-1):
        max = 0
```

```
        num_max = 0
        for j in range(i, len(L), 1):
            if max < L[j][1]:
                max = L[j][1]
                num_max = j
        L[i], L[num_max] = L[num_max], L[i]


def recommend_algo(graph, user):

    user_friends = set(graph.neighbors(user))
    recommendations = []

    for node in graph:
        if node != user and node not in user_friends:
            score = 0
            for i in graph.neighbors(node):
                if i in user_friends:
                    score += 1
            if score > 0:
                recommendations.append((node, score))
    trier(recommendations)
    return recommendations

user = "Alice"
recommendations = recommend_algo(G, user)

print("Recommandations d'amis pour %s :" %user)
for friend, score in recommendations:
    print("- %s (amis en commun : %d)" %(friend, score))

nx.draw(G, with_labels = True, node_color = "skyblue", node_size = 2000,␣
 ↪font_size = 10)
plt.show()
```

```
Recommandations d'amis pour Alice :
- Dave (amis en commun : 1)
- Emma (amis en commun : 1)
- Frank (amis en commun : 1)
```


Grace
Frank
Claire
Alice
Emma
Bob
Dave