

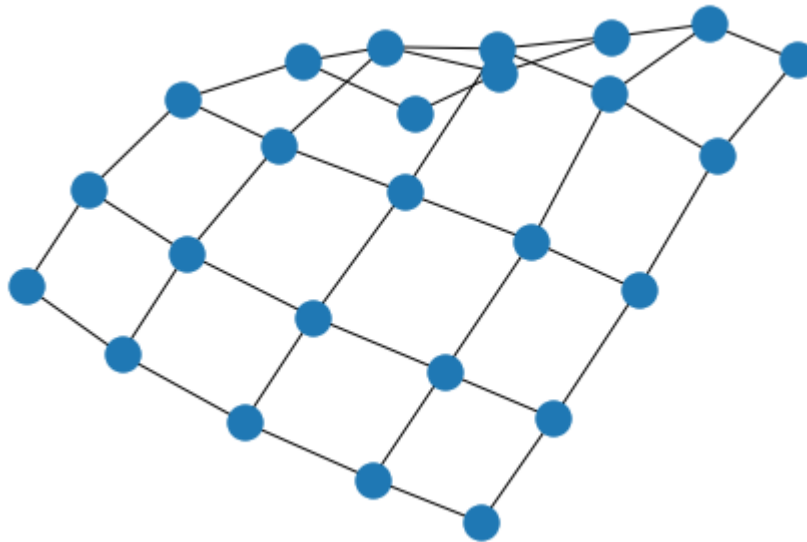
```
In [52]: import matplotlib.pyplot as plt
import networkx as nx

G = nx.grid_2d_graph(5, 5) # 5x5 grid

# print the adjacency list
# for line in nx.generate_adjlist(G):
#     print(line)
# write edgelist to grid.edgelist
# read edgelist from grid.edgelist
# nx.write_edgelist(G, path="grid.edgelist", delimiter=":")
nx.write_edgelist(G, path="grid.edgelist", delimiter=":")

H = nx.read_edgelist(path="grid.edgelist", delimiter=":")

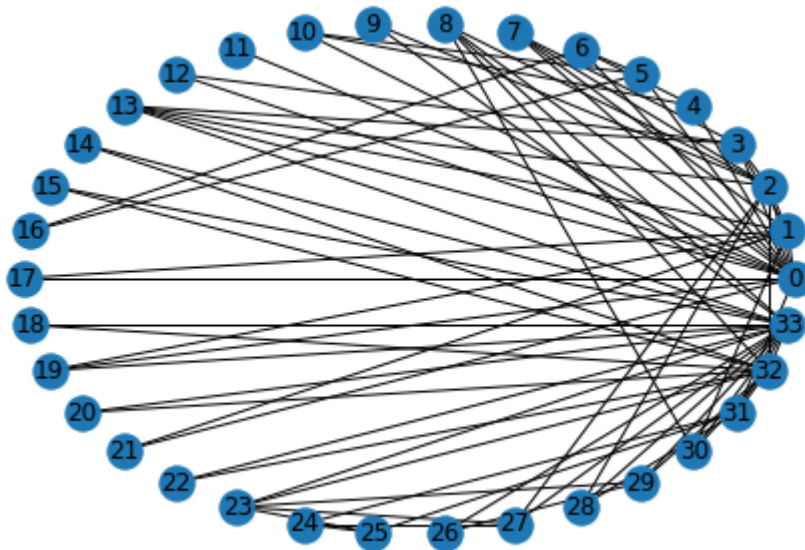
nx.draw(H)
plt.show()
```



```
In [51]: import matplotlib.pyplot as plt
import networkx as nx

G = nx.karate_club_graph()
#print("Node Degree")
#for v in G:
#    print('%s %s' % (v, G.degree(v)))

nx.draw_circular(G, with_labels=True)
plt.show()
```



```
In [12]: import networkx as nx

G=nx.Graph()
G.add_node("a")
G.add_nodes_from(["b", "c"])

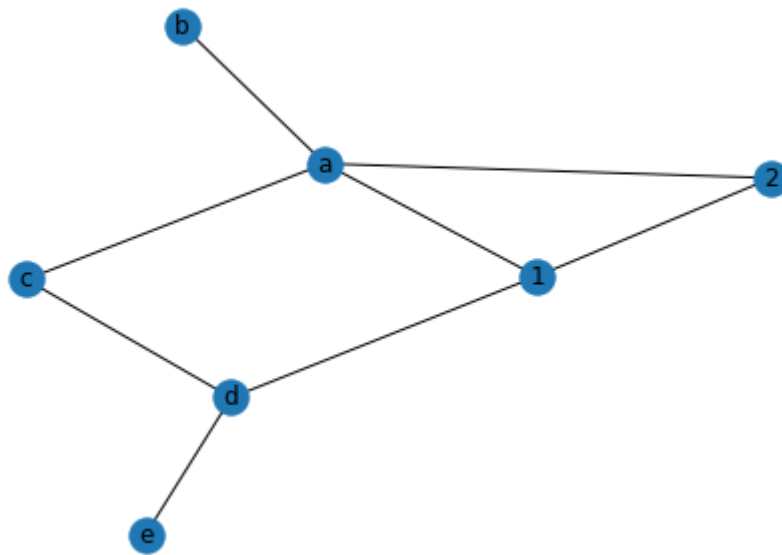
G.add_edge(1,2)
edge = ("d", "e")
G.add_edge(*edge)
edge = ("a", "b")
G.add_edge(*edge)

print("Nodes of graph: ")
print(G.nodes())
print("Edges of graph: ")
print(G.edges())
```

```
Nodes of graph:
['a', 'b', 'c', 1, 2, 'd', 'e']
Edges of graph:
[('a', 'b'), (1, 2), ('d', 'e')]
```

```
In [13]: # adding a list of edges:
G.add_edges_from([("a", "c"), ("c", "d"), ("a", 1), (1, "d"), ("a", 2)])
```

```
In [15]: nx.draw(G, with_labels=True)
plt.savefig("simple_path.png") # save as png
plt.show() # display
```



```
In [17]: import networkx as nx
import matplotlib.pyplot as plt

G=nx.path_graph(4)
cities = {0:"Toronto",1:"London",2:"Berlin",3:"New York"}

H=nx.relabel_nodes(G,cities)

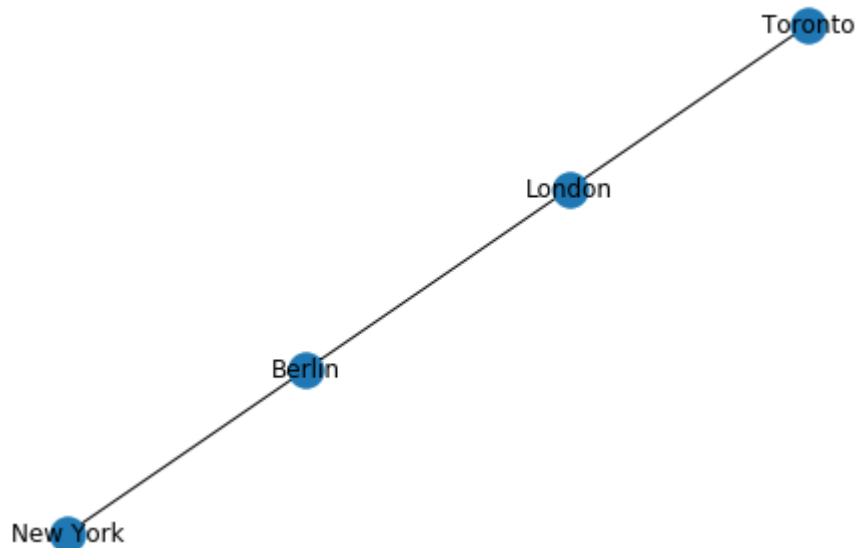
print("Nodes of graph: ")
print(H.nodes())
print("Edges of graph: ")
print(H.edges())
nx.draw(H,with_labels=True)
plt.savefig("path_graph_cities.png")
plt.show()
```

Nodes of graph:

['Toronto', 'London', 'Berlin', 'New York']

Edges of graph:

[('Toronto', 'London'), ('London', 'Berlin'), ('Berlin', 'New York')]



```
In [22]: import networkx as nx
import matplotlib.pyplot as plt

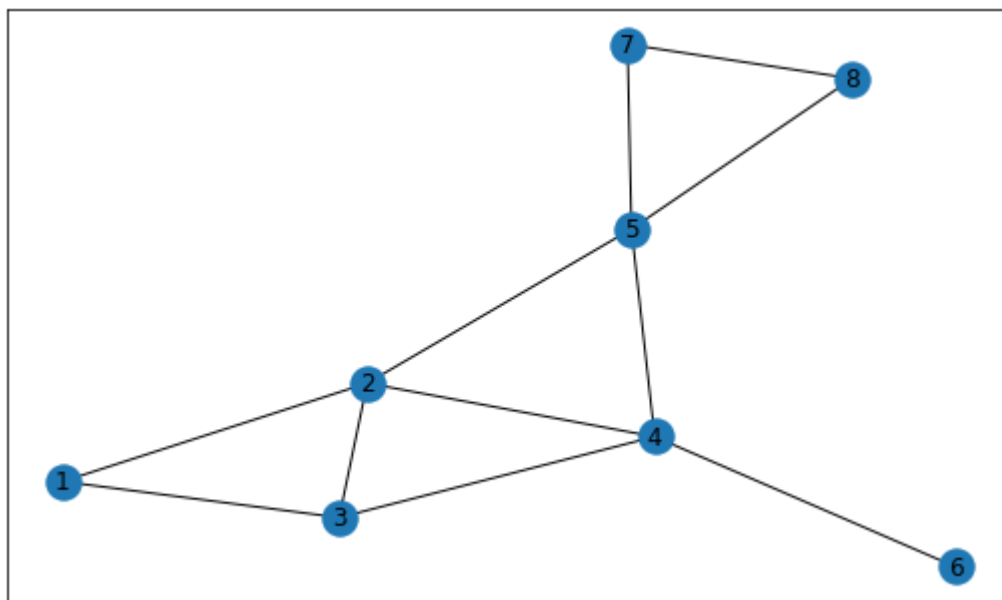
G = nx.Graph()

plt.figure(figsize =(9, 12))
G.add_edges_from([(1, 2), (1, 3), (2, 3), (2, 4), (2, 5), (3, 4),
                  (4, 5), (4, 6), (5, 7), (5, 8), (7, 8)])

# original Graph created
plt.subplot(211)
print("The original Graph:")

nx.draw_networkx(G)
```

The original Graph:



```
In [32]: import networkx as nx
import matplotlib.pyplot as plt

G = nx.Graph()

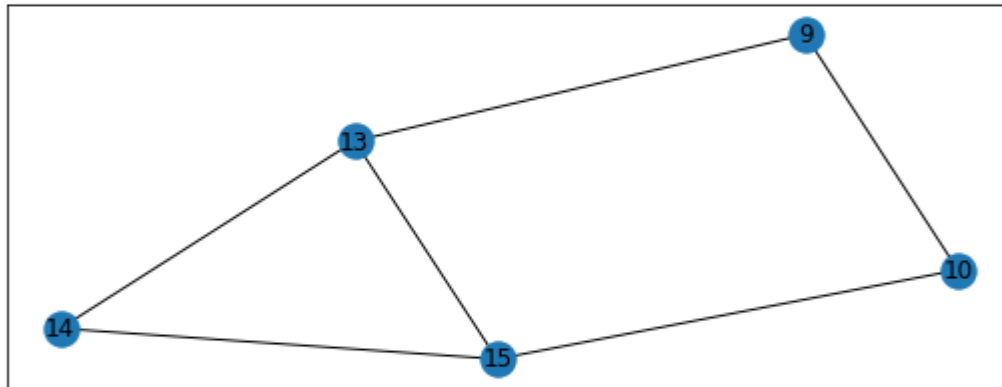
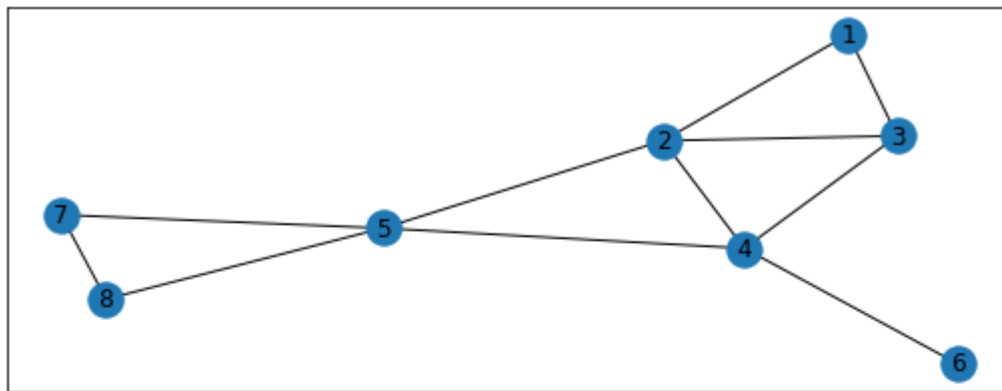
plt.figure(figsize =(9, 12))
G.add_edges_from([(1, 2), (1, 3), (2, 3), (2, 4), (2, 5), (3, 4),
                  (4, 5), (4, 6), (5, 7), (5, 8), (7, 8)])

# First Graph created
plt.subplot(311)
nx.draw_networkx(G)

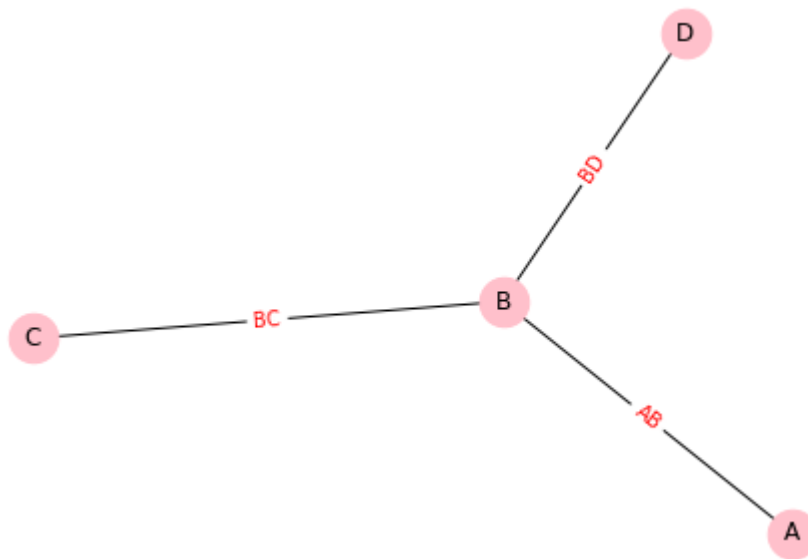
H = nx.Graph()
H.add_edges_from([(13, 14), (13, 15), (13, 9),
                  (14, 15), (15, 10), (9, 10)])

# Second Graph created
plt.subplot(312)
nx.draw_networkx(H)

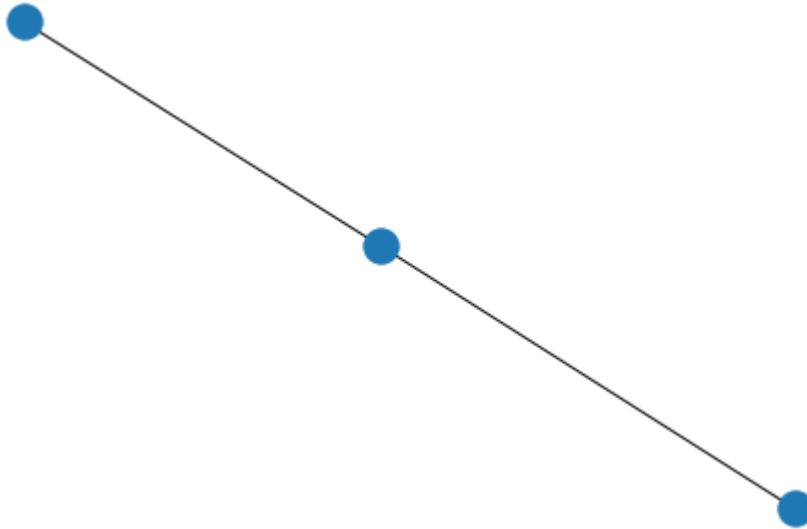
I = nx.union(G, H)
plt.subplot(313)
nx.draw_networkx(I)
```



```
In [31]: import networkx as nx
edges=[['A','B'],['B','C'],['B','D']]
G=nx.Graph()
G.add_edges_from(edges)
pos = nx.spring_layout(G)
plt.figure()
nx.draw(G,pos,edge_color='black',width=1,linewidths=1,\
node_size=600,node_color='pink',alpha=1.0,\
labels={node:node for node in G.nodes()})
nx.draw_networkx_edge_labels(G,pos,edge_labels={('A','B'):'AB',\
('B','C'):'BC',('B','D'):'BD'},font_color='red')
plt.axis('off')
plt.show()
```




```
In [38]: import networkx as nx  
g=nx.Graph()  
g.add_edge(1,3,weight=0.2)  
g.add_edge(3,2,weight=.3)  
nx.draw(g)
```



```
In [46]: import matplotlib.pyplot as plt
g2=nx.Graph()
g2.add_node('A',label='AA')
g2.add_node('B',label='BB')
g2.add_edge('A','B',weight=0.9)
g2.add_node('C',label='CC')
#nx.draw(g2)

pos = nx.spring_layout(g2) # compute graph layout
nx.draw(g2, pos, node_size=700) # draw nodes and edges
nx.draw_networkx_labels(g2, pos)

labels = nx.get_edge_attributes(g2, 'weight')
labels2 = nx.get_node_attributes(g2, 'label')
nx.draw_networkx_edge_labels(g2, pos, edge_labels=labels)

plt.show(g2)
print(labels2)
```



```
{'A': 'AA', 'B': 'BB', 'C': 'CC'}
```

```
In [48]: import matplotlib.pyplot as plt
g2=nx.Graph()
g2.add_node('A',label='AA')
g2.add_node('B',label='BB')
g2.add_edge('A','B',weight=0.9)
g2.add_node('C',label='CC')
#nx.draw(g2)

pos = nx.spring_layout(g2) # compute graph layout
nx.draw(g2, pos, node_size=700) # draw nodes and edges
nx.draw_networkx_labels(g2, pos)

labels = nx.get_edge_attributes(g2, 'weight')
labels2 = nx.get_node_attributes(g2, 'label')
nx.draw_networkx_edge_labels(g2, pos, edge_labels=labels)

plt.show(g2)
print(labels2)
```



```
{'A': 'AA', 'B': 'BB', 'C': 'CC'}
```

