Simulation of Team Sizes

Conclusion

Everyone can learn every aspect, but we need small isolated teams of **3-5** working properly on each aspect.

Reasons for upper bound as 5

Reason 1 - `Communication Overhead`

According to Josh Kauffman[1], high performance teams limit themselves to **3-12 members**, as anything larger will add communication overheads.

That is, larger teams have to spend more time ensuring that everyone is on the same page, thereby reducing efficiency.

Even Thahir's cousin said the same.

[1] author of The Personal MBA

Question time So why 5? Why not 6, 7, ... , 12? After all, Josh Kaufman said 3-12, right?

The reason is due to two essential **Economics principles**.

Reason 2 - `Law of Diminishing Marginal Productivity`

For every added unit of input, its corresponding added marginal productivity decreases.

Hence, apart from the communication overhead, I recommend upper bound as 5, as there is no task in any aspect of our project that will benefit from having more than 5 members.

Greater than 5 that will be recommended for technical work that will actually benefit from added hands, such as teams of doctors, surgeons, SWAT teams, etc.

Reason 3 - `Opportunity Cost`

It is a fraction that shows what you sacrificed to pick an option A instead of B. (We should minimize this)

$$\text{OC} = \frac{\text{what you sacrifice}}{\text{what you get}}$$

So, if we make large teams focused on a task, we are dedicating our scarce resource (here, limited members) on that task only, thereby sacrificing utilization of members for other tasks.

Methodology for Communication Overhead Analysis

Each team is taken as a **complete** graph, with each

- node/vertex as a person
- edge as an bidirectional interaction

Formula

If n is the number of vertices, then the number of edges in the complete graph is $\frac{n(n+1)}{2}$

In [1]:
```
import networkx as nx
import numpy as np

import matplotlib.pyplot as plt
%config InlineBackend.figure_formats = ['svg'] # makes everything svg by default
%matplotlib inline
```

In [2]:
```
def subgraph(n):
    vertices = np.arange(1, n+1)

    edges = []
    for i in range(1, n+1):
        for j in range(i+1, n+1):
            if(i != j):
                edges.append(
                    (i, j)
                )
    no_of_edges = len(edges)

    graph = nx.Graph()
    graph.add_nodes_from(vertices)
    graph.add_edges_from(edges)

    plt.figure(
      figsize=(0.5+n/4, 0.5+n/4),
      dpi=80
    )

    if(n<=12):
        node_color = "tab:blue"
        edge_color = "lightblue"
    else:
        node_color = "darkred"
        edge_color = "lightpink"

    nx.draw(
        graph,

        node_size = 30,
        node_color = node_color,

        width = 0.75,
        edge_color = edge_color
    )

    title = "Team of " + str(n) + " members"
    if(n==1):
        title = title[:-1]
    title += " - " + str(no_of_edges) + " interactions"
    if(no_of_edges==1):
        title = title[:-1]

    plt.title(title)
    plt.show()
```

In [3]:
```
def create_graph(n, divisions=1):
    division_size = int(n/divisions)

    if(divisions != 1):
        interactions_per_team = int(division_size*(division_size - 1)/2)
        total_interactions = interactions_per_team * divisions

        message = (
            "Team of " + str(n) +
            " divided into " + str(divisions) + " divisions " +
            "with " + str(division_size) + " members each" +
            "\n\n" +
            "Total interactions = " + str(total_interactions) + "\n" +
            "Interactions per team = " + str(interactions_per_team) +
            "\n\n" +
            "Each team will look like this"
        )
        print(message)

    subgraph(division_size)
```

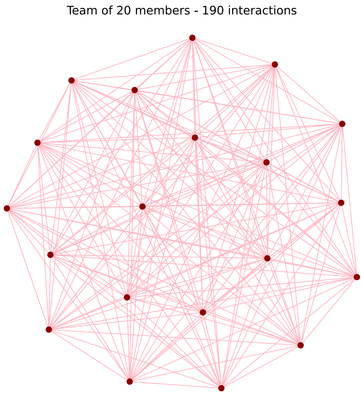
In [4]:
```
total = 100
divisions = [5, 20]

for i in divisions:
    create_graph(total, i)
```

```
Team of 100 divided into 5 divisions with 20 members each

Total interactions = 950
Interactions per team = 190

Each team will look like this
```




```
Team of 100 divided into 20 divisions with 5 members each

Total interactions = 200
Interactions per team = 10

Each team will look like this
```