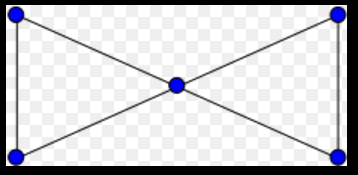
**Graphs Concepts**

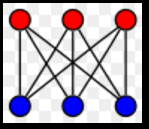
**Document Id 39**

* Edge, Vertex (node)
* Adjacent edge → connect 2 vertices.
* Vertex degree → number edges
* Path → chain of vertices connected by edges
* Cycle → path with the same first and last vertices
* Connected graph → path from every vertex to every other vertex
* Acyclic graph → no cycles

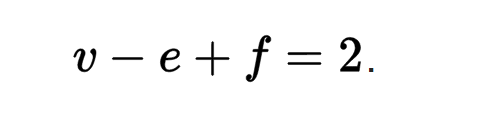
Planar Graph



Non Planar Graph



Planar GraphsEulers Formula v = num vertecies, e = number of edges, f = number of faces



**Graph database**

<https://en.wikipedia.org/wiki/Graph_database>

Simple but highly effective idea

In a relational database a relationship is expressed by a join

Generally we join on keys

Joins are a critical bottleneck for performance

In a graph database a relationship is an edge

A path in a graph expresses complex relationships

Queries on Graph databases come down to finding paths

**Neo4j**  well known popular

<https://neo4j.com/>

**GraphX**

http://spark.apache.org/graphx/

**Apache Jena** (not strict graph engine)

<https://jena.apache.org/>

**Directed Graph**

<https://en.wikipedia.org/wiki/Directed_graph>

Key points

* edges will now be ordered pairs
* We have 2 types of degree comining in (indegree)

and going out (outdegree)

Directed acyclic graph

https://en.wikipedia.org/wiki/Directed\_acyclic\_graph

no directed cycles

