**Graphs in JavaScript:**

Graphs are abstract data structures that consist of a set of vertices (nodes) connected by edges. They are widely used to model relationships between objects, such as networks, social connections, and dependencies.

**Directed Graphs:**

In a directed graph, edges have a direction, meaning they go from one vertex to another in a specific direction. It represents asymmetric relationships between vertices.

**Undirected Graphs:**

In an undirected graph, edges have no direction, meaning they connect vertices bidirectionally. It represents symmetric relationships between vertices.

**Weighted Graphs:**

In a weighted graph, each edge has a weight or cost associated with it. These weights can represent distances, costs, or any other metric relevant to the problem being modeled.

**Graph Traversal Algorithms:**

* Breadth-First Search (BFS):
  + BFS explores all the neighboring nodes at the present depth prior to moving on to the nodes at the next depth level. It's useful for finding the shortest path in an unweighted graph.
* Depth-First Search (DFS):
  + DFS explores as far as possible along each branch before backtracking. It's useful for topological sorting, cycle detection, and maze solving.

**Shortest Path Algorithms:**

* Dijkstra's Algorithm:
  + Dijkstra's algorithm is used to find the shortest path from a source vertex to all other vertices in a weighted graph with non-negative edge weights. It uses a priority queue to greedily select the vertex with the smallest distance.
* Bellman-Ford Algorithm:
  + Bellman-Ford algorithm is used to find the shortest path from a source vertex to all other vertices in a weighted graph, even if it contains negative edge weights. It iterates through all edges multiple times, relaxing the distances until the shortest paths are found.

**Minimum Spanning Tree Algorithms:**

* Prim's Algorithm:
  + Prim's algorithm is used to find the minimum spanning tree (MST) of a connected, undirected graph. It starts with an arbitrary vertex and grows the tree by adding the cheapest edge that connects a vertex in the tree to a vertex outside the tree.
* Kruskal's Algorithm:
  + Kruskal's algorithm is another algorithm used to find the minimum spanning tree of a connected, undirected graph. It starts with a forest of single-node trees and repeatedly adds the cheapest edge that does not produce a cycle until all vertices are connected.