GraphFrames is a framework used for graph processing.

Creating GraphFrames
This can be done using vertex and edge DataFrames

- A vertex dataframe should contain a special column named “id”
- An edge dataframe should contain two special columns “src” and “dst”

All other columns are optional and can be added as needed.

Querying graph and DataFrame
Made directly on the vertex and edge DataFrames since GraphFrames represent these graphs as pairs of the two. These DataFrames are listed as vertices and edges fields in the GraphFrame.

Motif Finding
These are just graph queries or graph pattern matching. The pattern is an expression used to define some connected vertices. Used to create more complex relationships of the edges and vertices. This will result in a new dataframe in which column names are motif keys. GraphFrames offers the motifs finding feature through find(pattern: String) method.

- Stateful queries: a more complex query that carries state along a path in the motif

Subgraph
Builds subgraphs by filtering on edges and vertices

BFS (Breadth-first Search)
Finds the shortest path(s) from one vertex (or a set of vertices) to another vertex (or a set of vertices)

Connected Components
Returns a DataFrame with each vertex assigned a component ID

Strongly connected components
Returns a DataFrame with each vertex assigned to the SCC containing that vertex

Label Propagation
Runs static Label Propagation Algorithm for detecting communities in networks

Shortest paths
Computes shortest paths to the given set of landmark vertices, where landmarks are specified by vertex ID

PageRank
This is the number and quality of links to a page inorder to determine its importance. More Important websites will have more links from other websites.
When a person is surfing on the internet, there is a point where the person will stop clicking on the links. The probability that at any point the person will continue is called the damping factor. THis can be set using the resetProbability parameter. Other important parameters are the tolerance (tol) and the maximum number of iterations maxIter.

TriangleCount
Counts the number of triangles passing through each vertex in a given graph. This is done by invoking the triangleCount function.