

✓ **Congratulations! You passed!**

Next Item



1 / 1  
points

1.

To build GraphFrame you need two DataFrames one for vertices let's call it "vertexDF" and the second for edges let's call it "edgeDF". What obligatory columns should both of them have?

- ☐ both DataFrames should not contain any obligatory columns and can contain arbitrary columns. Those columns can represent vertex and edge attributes.
- ☒ vertexDF should have the obligatory column "id" and edgeDF should have two obligatory columns "src" and "dst"

Correct

True. A vertex DataFrame should contain a special column named "id" which specifies unique IDs for each vertex in the graph. An edge DataFrame should contain two special columns: "src" (source vertex ID of edge) and "dst" (destination vertex ID of the edge)

- ☐ vertexDF should not have any obligatory columns and edgeDF should have two obligatory columns "src" and "dst"
- ☐ vertexDF should have the obligatory column "id" and edgeDF shouldn't contain any special columns



1 / 1  
points

2.

Does GraphFrame represent directed or undirected graph?

- ☐ It represents an undirected graph

## Motif Finding

Practice Quiz, 9 questions



It represents a directed graph

**Correct**

9/9 points (100.00%)

True. Because by default each edge in the GraphFrames library has a direction from the source vertex to the destination vertex

---



1 / 1  
points

3.

You have created GraphFrame with the name `g` from the vertex DataFrame named `vertexDF` and the edges DataFrame named `edgeDF`. How can you get the original DataFrames `vertexDF` and `edgeDF` from `g`?



by `g.vertices` and `g.edges`

**Correct**

True. The original vertices and edges dataframes could be obtained by `g.vertices` and `g.edges` respectively.



by `g.vertexDF` and `g.edgesDF`



it's impossible

---



1 / 1  
points

4.

What will `g.inDegrees` built-in method generate?



DataFrame with two columns "id" and "inDegree", where "inDegree" will contain the sum of numbers of incoming edges and of outgoing edges for each vertex



DataFrame with two columns "id" and "inDegree", where "inDegree" will contain number of incoming edges for each vertex

**Correct**

true. You are learning very fast!

---

# Motif Finding

1 / 1  
points

9/9 points (100.00%)

Practice Quiz, 9 questions

5. True or false? Mark only the correct statements about GraphFrames's motif-finding DSL for structural patterns

- ☒ The vertices are put into parentheses (a), while the edges are denoted by square brackets [e].



**Correct**

True. You are absolutely right

- ☐ The vertices are put into square brackets [a], while the edges are denoted by parentheses (e).



1 / 1  
points

6. How many columns will be in the resulting DataFrame after proceeding the pattern "(a)-[e]->(b); (b)-[e2]->(c)"?

- ☐ 4
- ☐ 3
- ☒ 5



**Correct**

True. The Resulting Data Frame will contain the columns "a", "e", "b", "e2" and "c"



1 / 1  
points

7. If a motif contains named vertex a, then the resulting DataFrame will contain a column "a" which is a StructType. To what will the sub-fields of this StructType be equivalent to?

- ☒ To the schema (columns) of the original GraphFrame.vertices



## Motif Finding

Practice Quiz, 9 questions

**Correct**

True. If a motif contains named vertex a, then the result DataFrame will contain a column "a" which is a StructType with sub-fields equivalent to the schema (columns) of GraphFrame.vertices.

**9/9 points (100.00%)**



To the schema with only one field "Id"



To the schema (columns) of the original GraphFrame

---



1 / 1  
points

8.

Pattern "(a)-[e]->(b); !(b)-[e2]->(a)" finds:



The edges from a to b for which there is edge from b to a.



The edges from a to b for which there is no edge from b to a.

**Correct**

True. An exclamation mark before an edge means that the edge should not be present in the graph.

---



1 / 1  
points

9.

What amount of shuffles will be done proceeding this "(A)-[]->(B); (B)-[]->(C); (A)-[]->(C)" pattern?



6

**Correct**

True. This pattern string will be transformed in the following collection of patterns:

[NamedVertex("A"),

AnonymousEdge(NamedVertex("A"), NamedVertex("B")),

NamedVertex("B"),

NamedVertex("B"),

**Motif Finding** AnonymousEdge(NamedVertex("B"), NamedVertex("C")),

**9/9 points (100.00%)**

Practice Quiz, 9 questions

NamedVertex("C"),

NamedVertex("A"),

AnonymousEdge(NamedVertex("A"), NamedVertex("C")),

NamedVertex("C")]

To proceed it with motif finding algorithm you will need to do 6 joins.

☐ 5

☐ 7

