**Javascript Commands**

connection = new Mongo('localhost:27017');

db = connection.getDB('Cruise1');

collection = db.getCollection('Cruises');

//1. Insert the documents into a database called Cruise1, collection Cruises.

doc = ({cruiseID: "1", destination: "Dubrovnik", shipName:"Spirit of the Sea",departDate:"20190502",returnDate:"20190513",

passenger: [{"passengerID" : "10", "name" : "Sue Smith", "address":"123 Sesame Street"},

{"passengerID" : "20", "name" : "Fran Jones", "address":"205 West Street"}]})

collection.insert(doc)

doc = ({cruiseID: "2", destination: "Dubrovnik", shipName:"Windjammer",departDate:"20190520",returnDate:"20190525",

passenger: [{"passengerID" : "20", "name" : "Fran Jones", "address":"205 West Street"},

{"passengerID" : "30", "name" : "George Jungle", "address":"270 North Fifth Street"}]})

collection.insert(doc)

doc = ({cruiseID: "3", destination: "Alaska", shipName:"Pacific Princess",departDate:"20190610",returnDate:"20190620",

passenger: [{"passengerID" : "30", "name" : "George Jungle", "address":"270 North Fifth Street"}]})

collection.insert(doc)

//2. Write a query to list all the json documents (pretty()) in the Cruises collection.

print("\n Print all json documents in Cruise1 DB\n\n");

cursor = collection.find()

while (cursor.hasNext()) {

doc = cursor.next();

printjson(doc);

}

**OUTPUT**

Print all json documents in Cruise1 DB

{

"\_id" : ObjectId("5eac6bf864ac12c7f34157b6"),

"cruiseID" : "1",

"destination" : "Dubrovnik",

"shipName" : "Spirit of the Sea",

"departDate" : "20190502",

"returnDate" : "20190513",

"passenger" : [

{

"passengerID" : "10",

"name" : "Sue Smith",

"address" : "123 Sesame Street"

},

{

"passengerID" : "20",

"name" : "Fran Jones",

"address" : "205 West Street"

}

]

}

{

"\_id" : ObjectId("5eac6bf864ac12c7f34157b7"),

"cruiseID" : "2",

"destination" : "Dubrovnik",

"shipName" : "Windjammer",

"departDate" : "20190520",

"returnDate" : "20190525",

"passenger" : [

{

"passengerID" : "20",

"name" : "Fran Jones",

"address" : "205 West Street"

},

{

"passengerID" : "30",

"name" : "George Jungle",

"address" : "270 North Fifth Street"

}

]

}

{

"\_id" : ObjectId("5eac6bf864ac12c7f34157b8"),

"cruiseID" : "3",

"destination" : "Alaska",

"shipName" : "Pacific Princess",

"departDate" : "20190610",

"returnDate" : "20190620",

"passenger" : [

{

"passengerID" : "30",

"name" : "George Jungle",

"address" : "270 North Fifth Street"

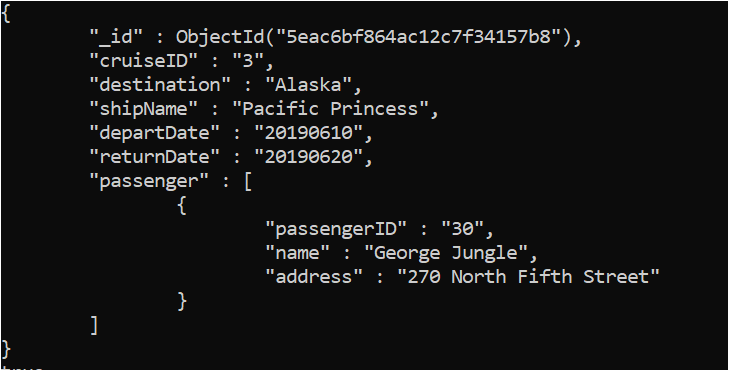
}

]

}

true

>

**Senario 2**

1.

Nodes label with cruise and Passenger are created.

There property is as follows:

CREATE(a:Cruise{cruiseId:'1',destination:'Dubrovnik',shipName:'Spirit of the Sea'})

CREATE(b:Cruise{cruiseId:'2',destination:'Saint Croix',shipName:'Windjammer'})

CREATE(c:Cruise{cruiseId:'3',destination:'Alaska',shipName:'Pacific Princess'})

**Passenger:**

CREATE(p1:Passenger{passengerID:'10',name:'Sue Smith',address:'123 Sesame Street',cruiseid:['1','2']}),

(p2:Students{passengerID:'20',name:'Fran Jones',address:'205 West Street',cruiseid:['2','3']}),

(p3:Students{passengerID:'30',name:'George Jungle',address:'270 North Fifth Street',cruiseid:'1'})

2.

**Relationships:**

Each passenger is added to its respective cruise. Property of relationship is departdate and returnDate.

Following is property of each passenger and each cruise.

MATCH(x{passengerID:'10'}),(y{cruiseId:'1'})CREATE(x)-[:CRUISE\_PASSENGER{departDate:'20190502',returnDate:'20190513'}]->(y)

MATCH(x{passengerID:'20'}),(y{cruiseId:'1'})CREATE(x)-[:CRUISE\_PASSENGER{departDate:'20190502',returnDate:'20190513'}]->(y)

MATCH(x{passengerID:'20'}),(y{cruiseId:'2'})CREATE(x)-[:CRUISE\_PASSENGER{departDate:'20190520',returnDate:'20190525'}]->(y)

MATCH(x{passengerID:'30'}),(y{cruiseId:'2'})CREATE(x)-[:CRUISE\_PASSENGER{departDate:'20190520',returnDate:'20190525'}]->(y)

MATCH(x{passengerID:'30'}),(y{cruiseId:'3'})CREATE(x)-[:CRUISE\_PASSENGER{departDate:'20190610',returnDate:'20190620'}]->(y)