Mongo DB queries

1. Create a collection and insert documents into it using insertOne() and insertMany()

```
sampledb> db.suppliers.insertOne({
supplier_name: "Geeta",
city: "Delhi",
part_name: "P1",
color: "red",
price: 3000,
phoneno: 99887766"
});
```

```
sampledb> db.suppliers.insertMany([
 { supplier_name: "Ravi", city: "Mumbai", part_name: "P2", color: "green", price: 1500, phoneno: "88776655" },
 { supplier_name: "Sita", city: "Delhi", part_name: "P3", color: "blue", price: 6000, phoneno: "77665544" },
 { supplier_name: "Amit", city: "Chennai", part_name: "P4", color: "red", price: 1800, phoneno: "66555433" },
 { supplier_name: "Rekha", city: "Kolkata", part_name: "P5", color: "green", price: 2500, phoneno: "55443322" }
]);
```

2. Select all documents in collection

```
sampledb> db.suppliers.find();
```

3. Find the count of all suppliers

```
sampledb> db.suppliers.countDocuments();
```

4. Find all records that have city = 'Delhi'

```
sampledb> db.suppliers.find({ city: "Delhi" });
```

5. Retrieve all documents that have color equal to 'red' or 'green'

```
sampledb> db.suppliers.find({ color: { $in: ["red", "green"] } });
```

6. Retrieve all documents where part_name is 'P1' or price is less than 200.

```
sampledb> db.suppliers.find({
$or: [{ part_name: "P1" }, { price: { $lt: 200 } }]
 });
```

7. Update the record of 'Geeta', set city = 'Bombay' and phoneno ='11223344'

```
sampledb> db.suppliers.updateOne(
{ supplier_name: "Geeta" },
{ $set: { city: "Bombay", phoneno: "11223344" } }
);
```

8. Delete all records where price is greater than 5000

```
sampledb> db.suppliers.deleteMany({ price: { $gt: 5000 } });
```

9. Display only the name and city of the supplier

```
sampledb> db.suppliers.find({}, { supplier_name: 1, city: 1, id: 0 });
```

10. Sort all suppliers on city and display only the first two records.

```
sampledb> db.suppliers.find().sort({ city: 1 }).limit(2);
```