GEO-LOCATION

1. Insert a geolocation structure.

* db.places.insertOne({name: 'California Academy of Sciences', location: {type: 'Point', coordinates: [37.7642095, -122.4703587]}})

1. Adding a geo-spatial index.

* db.places.createIndex({location: '2dsphere'})

1. Query the places that are near my location.

* db.places.find({location: {$near: {$geometry: {type: 'Point', coordinates: [-122.471114, 37.771104]}, $maxDistance: 800, $minDistance: 10}}})

1. Query the points that are within a polygon.

* db.places.find({location: {$geoWithin: {$geometry: {type: 'Polygon', coordinates: [[p1,p2,p3,p4,p1]]}}}}).pretty()

1. Find out if the point provided intersects with the area stored. We can also find out if another area intersects with an area stored.

* db.areas.find({area:{$geoIntersects: {$geometry: {type:'Point', coordinates: [-122.48446, 37.77776]}}}}).pretty()

1. Find out what points are in a certain radius from an epicenter.

* db.places.find({location: {$geoWithin: {$centerSphere: [[-122.46203,37.77286], 1/6378.1]}}}).pretty()