

23. Create index and fire queries with MongoDB

1. Import zip.json.
 2. Create single field, composite and multikey indexes.
 3. Fire queries given below again and write your analysis.
1. Display all cities having population above 1600.
 2. Display all cities in state "KS".
 3. Display location of city "TIMKEN"

//To work with this data in MongoDB,
let's assume the contents of the zip.json file have fields like city,
state, pop (population), and loc (location coordinates).

Query 1: import the zip.json file

```
mongoimport --db DYPIT --collection zips --file path/to/zip.json --jsonArray
```

Query 2: Create single field, composite and multikey indexes.

1.Single Field Index

```
db.zips.createIndex({ pop: 1 });
```

2.Composite (Compound) Index

```
db.zips.createIndex({ state: 1, city: 1 });
```

3.Multikey Index //If loc is an array (storing coordinates, for example),
MongoDB automatically creates a multikey index when indexing an array field.
Create an index on loc:

```
db.zips.createIndex({ loc: 1 });
```

Query 3: Fire queries given below again and write your analysis.

1. Display all cities having population above 1600.

```
db.zips.find({ pop: { $gt: 1600 } });
```

2. Display all cities in state "KS".

```
db.zips.find({ state: "KS" });
```

3. Display location of city "TIMKEN"

```
db.zips.find({ city: "TIMKEN" }, { loc: 1, _id: 0 });
```

```
-----  
-----  
-----  
-----  
-----  
-----
```