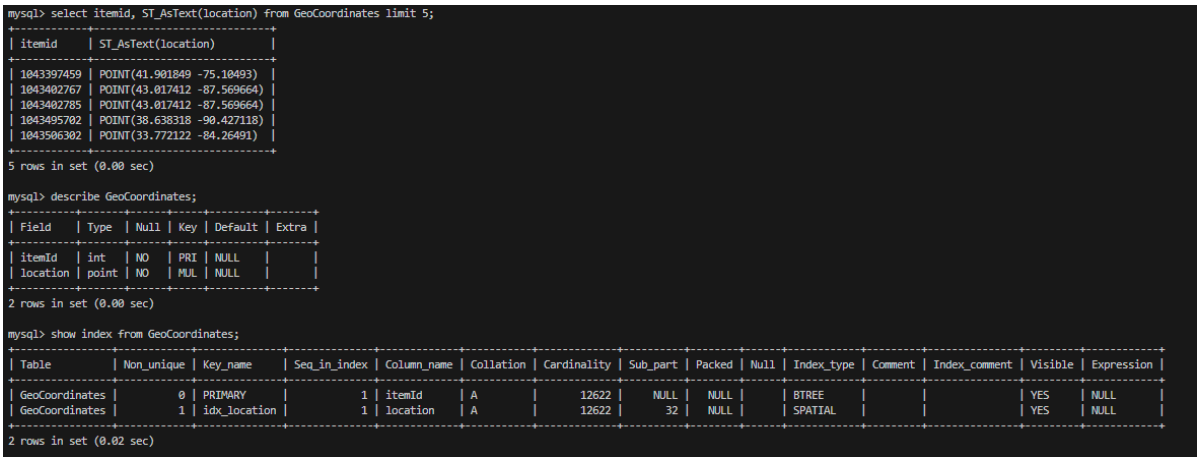
STMW - Programming Assignment 02

Part A: Create a Spatial Index in MySQL

MySQL validation



```
mysql> select itemid, ST_AsText(location) from GeoCoordinates limit 5;
+------------+-----------------------------+
| itemid     | ST_AsText(location)         |
+------------+-----------------------------+
| 1043397459 | POINT(41.901849 -75.10493)  |
| 1043402767 | POINT(43.017412 -87.569664) |
| 1043402785 | POINT(43.017412 -87.569664) |
| 1043495702 | POINT(38.638318 -90.427118) |
| 1043506302 | POINT(33.772122 -84.26491)  |
+------------+-----------------------------+
5 rows in set (0.00 sec)

mysql> describe GeoCoordinates;
+----------+-------+------+-----+---------+-------+
| Field    | Type  | Null | Key | Default | Extra |
+----------+-------+------+-----+---------+-------+
| itemId   | int   | NO   | PRI | NULL    |       |
| location | point | NO   | MUL | NULL    |       |
+----------+-------+------+-----+---------+-------+
2 rows in set (0.00 sec)

mysql> show index from GeoCoordinates;
+--------------+------------+--------------+--------------+-------------+-----------+-------------+----------+--------+------+------------+---------+---------------+---------+------------+
| Table        | Non_unique | Key_name     | Seq_in_index | Column_name | Collation | Cardinality | Sub_part | Packed | Null | Index_type | Comment | Index_comment | Visible | Expression |
+--------------+------------+--------------+--------------+-------------+-----------+-------------+----------+--------+------+------------+---------+---------------+---------+------------+
| GeoCoordinates |          0 | PRIMARY      |            1 | itemId      | A         |       12622 |     NULL | NULL   |      | BTREE      |         |               | YES     | NULL       |
| GeoCoordinates |          1 | idx_location |            1 | location    | A         |       12622 |       32 | NULL   |      | SPATIAL    |         |               | YES     | NULL       |
+--------------+------------+--------------+--------------+-------------+-----------+-------------+----------+--------+------+------------+---------+---------------+---------+------------+
2 rows in set (0.02 sec)
```

Part B: Create a Lucene Index

Using this SQL query for storing to Lucene index

```
SELECT i.itemId, i.name, GROUP_CONCAT(c.category SEPARATOR '; ') AS categories,
i.description, COALESCE(i.currently, 0) as price, COALESCE(ill.latitude, 0.00) as
latitude, COALESCE(ill.longitude, 0.00) as longitude FROM Items i INNER JOIN
Categories c ON i.itemId = c.itemId LEFT JOIN ItemLatLon ill ON i.itemId =
ill.itemId GROUP BY i.itemId, i.name, i.description, i.currently, ill.latitude,
ill.longitude;
```

Explanation:

- Getting `itemId`, `name`, `description`, `price` values from table `Items`
- Getting `category` values from table `Categories`
- Getting `latitude`, `longitude` values from table `ItemLatLon`
- Using `COALESCE`, we're replacing NULL value(s) with 0, for normalization of column `price`, `latitude`, `longitude`

Part C: Implement the Search Function

- Used {"name", "categories", "description"} `as the fields for text searching process using Lucene` SimpleAnalyzer `and` MultiFieldQueryParser`
- Added all required output fields
- Added extra arguments for `latitude`, `longitude`, and `width`
- Used `haversineDistance` mathematical formula to calculate geo-location distance

- Implemented ranking system using Java `sort` function

Part D: Automize

Implemented via `runLoad.sh`

Example process:

```
root@393b3dd196dc:/stmw/student_workspace# mysql < dropSpatialIndex.sql
root@393b3dd196dc:/stmw/student_workspace# ./runLoad.sh
Enlik - Programming Assignment 02
Part A: Create a Spatial Index in MySQL
Creating new table...
Tables created successfully.

Part B: Create a Lucene Index

WARNING: Using incubator modules: jdk.incubator.vector
Indexing to directory: indexes
Feb 14, 2025 4:14:41 AM org.apache.lucene.internal.vectorization.PanamaVectorizationProvider <init>
INFO: Java vector incubator API enabled; uses preferredBitSize=256; FMA enabled
Query: SELECT i.itemId, i.name, GROUP_CONCAT(c.category SEPARATOR '; ') AS categories, i.description, COALESCE(i.currently, 0) as price, COALESCE(
temId LEFT JOIN ItemLatLon ill ON i.itemId = ill.itemId GROUP BY i.itemId, i.name, i.description, i.currently, ill.latitude, ill.longitude;

Indexing finished

Part C: Implement the Search Function

WARNING: Using incubator modules: jdk.incubator.vector
Searching for: Marvel
Feb 14, 2025 4:14:56 AM org.apache.lucene.internal.vectorization.PanamaVectorizationProvider <init>
INFO: Java vector incubator API enabled; uses preferredBitSize=256; FMA enabled

#---------------------------------------#
Number of hits: 242
#---------------------------------------#
itemId: 1047970072, name: Captain Marvel # 32 1971, score: 5.037465572357178, price: 4.99
itemId: 1046598227, name: Marvel Comics - Wolverine, score: 4.3624982833862305, price: 2.00
itemId: 1049358715, name: Ms Marvel #7, score: 4.350419044494629, price: 2.00
itemId: 1046556864, name: Avengers 10 Marvel Comics 1964, score: 4.218635082244873, price: 8.99
itemId: 1049522832, name: *~* CAPTAIN MARVEL *~* #22 F, score: 4.138017654418945, price: 2.00
WARNING: Using incubator modules: jdk.incubator.vector
Searching for: Marvel
Geo-location search enabled: longitude=40.849879, latitude=-73.97501, width=100.0 km
Feb 14, 2025 4:14:57 AM org.apache.lucene.internal.vectorization.PanamaVectorizationProvider <init>
INFO: Java vector incubator API enabled; uses preferredBitSize=256; FMA enabled

#---------------------------------------#
Number of hits: 242
#---------------------------------------#
itemId: 1047970072, name: Captain Marvel # 32 1971, score: 5.037465572357178, distance: 15095.0 km, price: 4.99
itemId: 1046598227, name: Marvel Comics - Wolverine, score: 4.3624982833862305, distance: 15932.9 km, price: 2.00
itemId: 1049358715, name: Ms Marvel #7, score: 4.350419044494629, distance: 8667.31 km, price: 2.00
itemId: 1046556864, name: Avengers 10 Marvel Comics 1964, score: 4.218635082244873, distance: 15217.04 km, price: 8.99
itemId: 1049522832, name: *~* CAPTAIN MARVEL *~* #22 F, score: 4.138017654418945, distance: 14004.7 km, price: 2.00
```

Known Issues

- The geo-location distance calculation using `haversineDistance` somehow giving wrong calculation, need to be updated