MIS 375 Group Project – Nodak Hiring

**Milestone 4**: Advanced Database Implementation

**Due Date**: Friday, May 5th

Tables*:* Extend your existing MySQL database with new tables from the complete relational schema and make appropriate changes to the existing tables if necessary. Include the SQL DDL statements for creating the **new tables** in your report.

Sample Data*:* Make up some sample data (3-5 records) for each table.

**Company Table:**

mysql> select \* from Company;

+-----------+--------------------+------------------------+---------------+----------------+

| CompanyID | name | address | Contact\_name | Contact\_Number |

+-----------+--------------------+------------------------+---------------+----------------+

| 12543 | Water Works | 1853 Baltic Avenue | Jimothy Smith | 9523549876 |

| 14856 | Board Construction | 9627 Board Walk Street | Jimmy Ray | 6514856725 |

| 23758 | Electric Company | 1505 Lake Avenue N | Bill Jones | 6514782746 |

+-----------+--------------------+------------------------+---------------+----------------+

3 rows in set (0.00 sec)

**Job Assignment table:**

mysql> select \* from job\_assignment;

+------------+----------+------------+------------+

| job\_number | workerID | start\_date | end\_date |

+------------+----------+------------+------------+

| and22 | 46728 | 2023-05-04 | 2023-06-30 |

| and12 | 79364 | 2023-05-04 | 2023-06-30 |

| and32 | 79364 | 2023-05-05 | 2023-05-07 |

| and32 | 82743 | 2023-05-04 | 2023-06-30 |

+------------+----------+------------+------------+

4 rows in set (0.00 sec)

**Job Posting:**

mysql> select \* from job\_posting;

+-----------+----------+------------+-----------------+--------------+----------------+---------------------+

| CompanyID | job\_code | job\_number | job\_description | compensation | workers\_needed | workers\_skills |

+-----------+----------+------------+-----------------+--------------+----------------+---------------------+

| 12543 | job12 | and12 | coder | 5.00 | 1 | legend |

| 14856 | job50 | and22 | analyst | 10.00 | 2 | hard worker |

| 23758 | job69 | and32 | accountant | 15.00 | 3 | attention to detail |

+-----------+----------+------------+-----------------+--------------+----------------+---------------------+

3 rows in set (0.00 sec)

**Job\_type:**

mysql> select \* from job\_type;

+----------+-------------+

| job\_code | description |

+----------+-------------+

| blue1 | coder |

| job12 | coder |

| job50 | analyst |

| job69 | accountant |

| njn11 | team\_lead |

| red24 | developer |

+----------+-------------+

6 rows in set (0.01 sec)

**Qualified Candidates:**

mysql> select \* from qualified\_candidates;

+-------------+---------+------------+----------+----------------+-----------------+

| candidateID | staffID | job\_number | workerID | date\_contacted | worker\_response |

+-------------+---------+------------+----------+----------------+-----------------+

| cand1 | stf1 | and12 | 46728 | 2023-05-04 | no |

| cand2 | stf2 | and22 | 79364 | 2023-05-04 | yes |

| cand3 | stf3 | and32 | 82743 | 2023-05-04 | yes |

+-------------+---------+------------+----------+----------------+-----------------+

3 rows in set (0.00 sec)

**Required by:**

mysql> select \* from required\_by;

+------------+------------+

| job\_number | skill\_code |

+------------+------------+

| and12 | skil1 |

| and22 | skil2 |

| and32 | skil3 |

+------------+------------+

3 rows in set (0.00 sec)

**Sales Staff:**

mysql> select \* from sales\_staff;

+---------+------------+--------+

| staffID | commission | region |

+---------+------------+--------+

| stf1 | 500.00 | NA |

| stf2 | 700.00 | EU |

| stf3 | 1000.00 | NA |

+---------+------------+--------+

3 rows in set (0.00 sec)

**Skill:**

mysql> select \* from skill;

+------------+-------------+

| skill\_code | skill\_name |

+------------+-------------+

| skil1 | coding |

| skil2 | data master |

| skil3 | math genius |

+------------+-------------+

3 rows in set (0.00 sec)

**Staff:**

mysql> select \* from staff;

+---------+-----------+----------+---------------+-------------+--------------+-----------+--------------+

| staffID | firstName | lastname | date\_of\_birth | date\_joined | phone\_number | job\_title | address |

+---------+-----------+----------+---------------+-------------+--------------+-----------+--------------+

| stf1 | Janet | Jackson | 1970-10-20 | 2023-04-27 | 1234567890 | finder | 4th street N |

| stf2 | Ryan | Stone | 1980-03-10 | 2023-04-20 | 8889990000 | grader | 5th ave W |

| stf3 | Garret | Lionel | 1990-02-22 | 2023-03-06 | 1112223333 | poster | 8th street S |

+---------+-----------+----------+---------------+-------------+--------------+-----------+--------------+

3 rows in set (0.00 sec)

**Survey:**

mysql> select \* from survey;

+----------+-----------------+-------------+------------+----------+

| surveyID | completion\_date | link | job\_number | workerID |

+----------+-----------------+-------------+------------+----------+

| surv1 | 2023-05-04 | survey1\_org | and12 | 46728 |

| surv2 | 2023-05-04 | survey2\_org | and22 | 79364 |

| surv3 | 2023-05-04 | survey2\_org | and32 | 82743 |

+----------+-----------------+-------------+------------+----------+

3 rows in set (0.00 sec)

**Test:**

mysql> select \* from test;

+----------+---------+------------+-------+

| workerID | staffID | date | score |

+----------+---------+------------+-------+

| 46728 | stf1 | 2023-04-27 | 69 |

| 79364 | stf2 | 2023-04-27 | 98 |

| 82743 | stf2 | 2023-04-27 | 78 |

+----------+---------+------------+-------+

3 rows in set (0.01 sec)

**Worker:**

mysql> select \* from worker;

+----------+------------------+--------------+----------------------+--------------------+-------------+

| workerID | name | phone\_number | available\_start\_date | available\_end\_date | type\_of\_job |

+----------+------------------+--------------+----------------------+--------------------+-------------+

| 46728 | Reid MacLeod | 6512009784 | 2023-01-01 | 2024-05-23 | coder |

| 79364 | Andrew McMonagle | 9527457249 | 2023-05-08 | 9999-12-31 | analyst |

| 82743 | Nick Nuelle | 9523488273 | 2023-01-01 | 2025-07-16 | gardner |

+----------+------------------+--------------+----------------------+--------------------+-------------+

3 rows in set (0.00 sec)

**Worker job preference:**

mysql> select \* from worker\_job\_preference;

+----------+----------+

| workerID | job\_code |

+----------+----------+

| 79364 | job12 |

| 82743 | job50 |

| 46728 | job69 |

+----------+----------+

3 rows in set (0.00 sec)

**Worker Skill:**

mysql> select \* from worker\_skill;

+----------+------------+

| workerID | skill\_code |

+----------+------------+

| 79364 | skil1 |

| 46728 | skil2 |

| 82743 | skil3 |

+----------+------------+

3 rows in set (0.00 sec)

Queries: Write SQL statements to perform the following searches on your Nodak Hiring database. All the query statements and results should be reported.

**1)** Find all the jobs that match the skills of a particular worker. The query should use a Worker ID (such as 123) for the search condition.

mysql> SELECT jp.job\_number, jp.job\_description, jp.compensation

-> FROM job\_posting jp

-> INNER JOIN required\_by rb ON jp.job\_number = rb.job\_number

-> INNER JOIN worker\_skill ws ON rb.skill\_code = ws.skill\_code

-> WHERE ws.workerID = '46728';

+------------+-----------------+--------------+

| job\_number | job\_description | compensation |

+------------+-----------------+--------------+

| and22 | analyst | 10.00 |

+------------+-----------------+--------------+

1 row in set (0.00 sec)

**2)** List all the available workers for a particular job based on workers’s job type preferences. The query should use a job posting ID for the search condition.

mysql> SELECT w.workerID, w.name, w.phone\_number, w.available\_start\_date, w.available\_end\_date

-> FROM worker w

-> INNER JOIN worker\_job\_preference wjp ON w.workerID = wjp.workerID

-> INNER JOIN job\_posting jp ON wjp.job\_code = jp.job\_code

-> WHERE jp.job\_number = 'and32' AND w.available\_start\_date <= NOW() AND w.available\_end\_date >= NOW();

+----------+--------------+--------------+----------------------+--------------------+

| workerID | name | phone\_number | available\_start\_date | available\_end\_date |

+----------+--------------+--------------+----------------------+--------------------+

| 46728 | Reid MacLeod | 6512009784 | 2023-01-01 | 2024-05-23 |

+----------+--------------+--------------+----------------------+--------------------+

1 row in set (0.00 sec)

**3)** Generate a report of all the jobs including the following information: job ID, description, job type, start date, end date, and number of workers assigned.

mysql> SELECT jp.job\_number, jp.job\_description, jt.description AS job\_type, MIN(ja.start\_date) AS start\_date, MAX(ja.end\_date) AS end\_date, COUNT(ja.workerID) AS num\_workers\_assigned

-> FROM job\_posting jp

-> INNER JOIN job\_type jt ON jp.job\_code = jt.job\_code

-> LEFT JOIN job\_assignment ja ON jp.job\_number = ja.job\_number

-> GROUP BY jp.job\_number, jp.job\_description, jt.description

-> ORDER BY jp.job\_number;

+------------+-----------------+------------+------------+------------+----------------------+

| job\_number | job\_description | job\_type | start\_date | end\_date | num\_workers\_assigned |

+------------+-----------------+------------+------------+------------+----------------------+

| and12 | coder | coder | 2023-05-04 | 2023-06-30 | 1 |

| and22 | analyst | analyst | 2023-05-04 | 2023-06-30 | 1 |

| and32 | accountant | accountant | 2023-05-04 | 2023-06-30 | 2 |

+------------+-----------------+------------+------------+------------+----------------------+

3 rows in set (0.00 sec)

Trigger/Stored Procedure: Write a trigger or stored procedure for the following task:

1) Once a worker is assigned to a job, their availability and available dates need to be updated by a small piece of program (trigger or stored procedure).

**Trigger that updates a worker’s available start date to be the day after the end date and available end date of the job they are assigned to and their available end date to be set to a far future date i.e. ‘9999-12-31’.**

mysql> DELIMITER //

mysql>

mysql> CREATE TRIGGER update\_worker\_availability

-> AFTER INSERT ON job\_assignment

-> FOR EACH ROW

-> BEGIN

-> UPDATE worker

-> SET available\_start\_date = NEW.end\_date + INTERVAL 1 DAY,

-> available\_end\_date = '9999-12-31'

-> WHERE workerID = NEW.workerID;

-> END//

Query OK, 0 rows affected (0.08 sec)

mysql>

mysql> DELIMITER ;

**Insert record into job\_assignment:**

mysql> INSERT INTO job\_assignment (job\_number, workerID, start\_date, end\_date)

-> VALUES ('and32', '79364', '2023-05-05', '2023-05-07');

Query OK, 1 row affected (0.04 sec)

**Result:**

mysql> SELECT workerID, name, available\_start\_date, available\_end\_date

-> FROM worker

-> WHERE workerID = '79364';

+----------+------------------+----------------------+--------------------+

| workerID | name | available\_start\_date | available\_end\_date |

+----------+------------------+----------------------+--------------------+

| 79364 | Andrew McMonagle | 2023-05-08 | 9999-12-31 |

+----------+------------------+----------------------+--------------------+

1 row in set (0.00 sec)

**If you work on a case other than the given one, follow the instruction below:**

* Create three SQL queries and one stored procedure based on the user requirements.
* Test and run the queries and stored procedure in MySQL.

**Submission instruction**: Please submit the following documents to the blackboard by the due date:

* A list of all tables with sample data in your database (use SELECT \* FROM <table> to display each table and its content)
* The three SQL statements and the query results
* The trigger or stored procedure code and its runtime results