Project Report

|  |  |
| --- | --- |
| Product Name | Advanced Certificate in Web Development |
| Qualification Name (NICF) | NICF-Advanced Certificate in Infocomm Technology (Software & Applications) |
| Product Name | Database Design and Implementation |
| Module Name (NICF) | ITSF-Database Design and Implementation |

|  |  |  |  |
| --- | --- | --- | --- |
| Student name | | Assessor name | |
| **Muhammad Kemal / BDSE-0922-084** | |  | |
| Date issued | Completion date | | Submitted on |
|  |  | |  |
|  | |  | |
| Project title | Design, Implement, Test & Document Community Portal Database. | | |

|  |
| --- |
| Learner declaration |
| I certify that the work submitted for this assignment is my own and research sources are fully acknowledged.  Student signature:  Date: 30 January 2023 |

Content

1. Project background
2. Project Objectives
3. Database Requirement Specifications
4. Task 1
5. Task 2
6. Task 3
7. Task 4
8. Task 5
9. Task 6

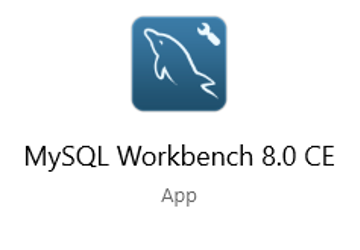
Project Background

This Project is used for Summative Assessment of student in the Module ‘Database Design & Implementation’ of the ITSF Course ‘NICF-Advanced Certificate in Web Development” This Project considers the skills required to design, develop & implement a MySQL Database.

Project Objective

Tools & platform used

MySQL Workbench – for making database, EERD and phpMyAdmin – for making data dictionary



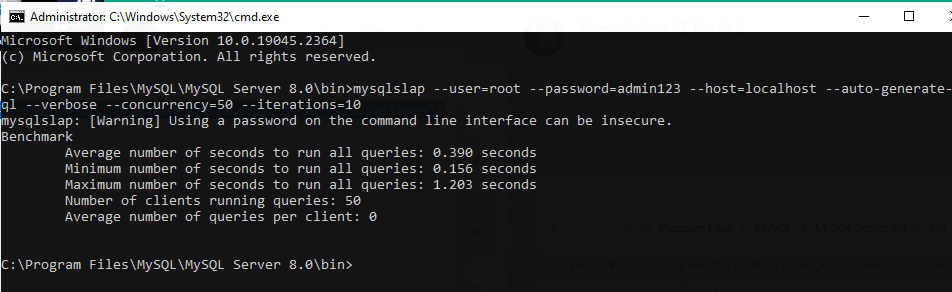
ERDPlus – for making ER Diagram and generating relational schema and sql query.



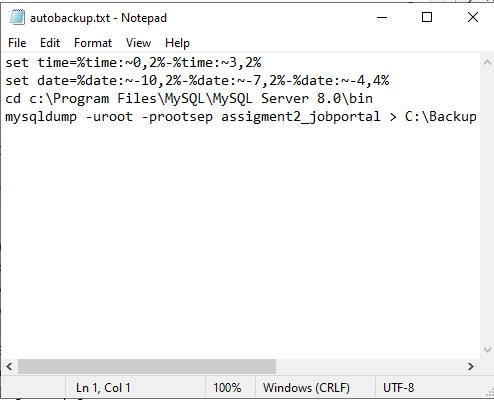
Mockaroo.com:- To generate sample data



Command prompt – for load testing



Notepad – for making batch file



# Task 1

Task Statement:

1. Create a database requirements specification document for the project scenario.

2. Include it as part of Project Report. (A1)

Solution:

Hardware requirements

**-** Gen Intel(R) Core(TM) i3

**-** Ram 4,00 GB

Software requirements

* MySQL Server (Database Server)
* MySQL Workbench (Database Client)
* XAMPP (Apache Server )
* phpMyAdmin (Data Dictionary)
* Web Browser (Google Chrome)
* Task Scheduler Windows
* Command Prompt
* Notepad

Database requirements (A1)

|  |  |
| --- | --- |
| Entity | **Description** |
| User | It includes details on how to sign up and log in. |
| Administrator | Administrators can manage accounts and send out mass emails. |
| Email | to keep track of each user's email details who has signed up with the system. |
| Job | It gives information on the user's employment chances. |
| Messages | Messages can be sent and received by users. |
| Thread | is a page or portion of a website that shows a particular dialogue between numerous users, also referred to as a "thread." |

Entities Relationship Table

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| No | Entity one | Entity two | Relationship | Note/Description |
| 1 | User | Administrator | one to many | One Admin can administer many User |
| 2 | Admin | Bulk email | One to many | One admin can send many bulk emails |
| 3 | User | Thread | Many to many | One user can create and participate in numerous threads, and multiple users can be in a single thread. |
| 4 | User | Messages | One to Many | One message can only have one user linked with it; however, one user can have several messages. |
| 5 | User | Job | Many to Many | Both a single user and a single job can have several users linked with them. |

# Task 2

Task Statement

# 1. Create a database design document.

# 2. Create a Conceptual Design for the proposed Database.

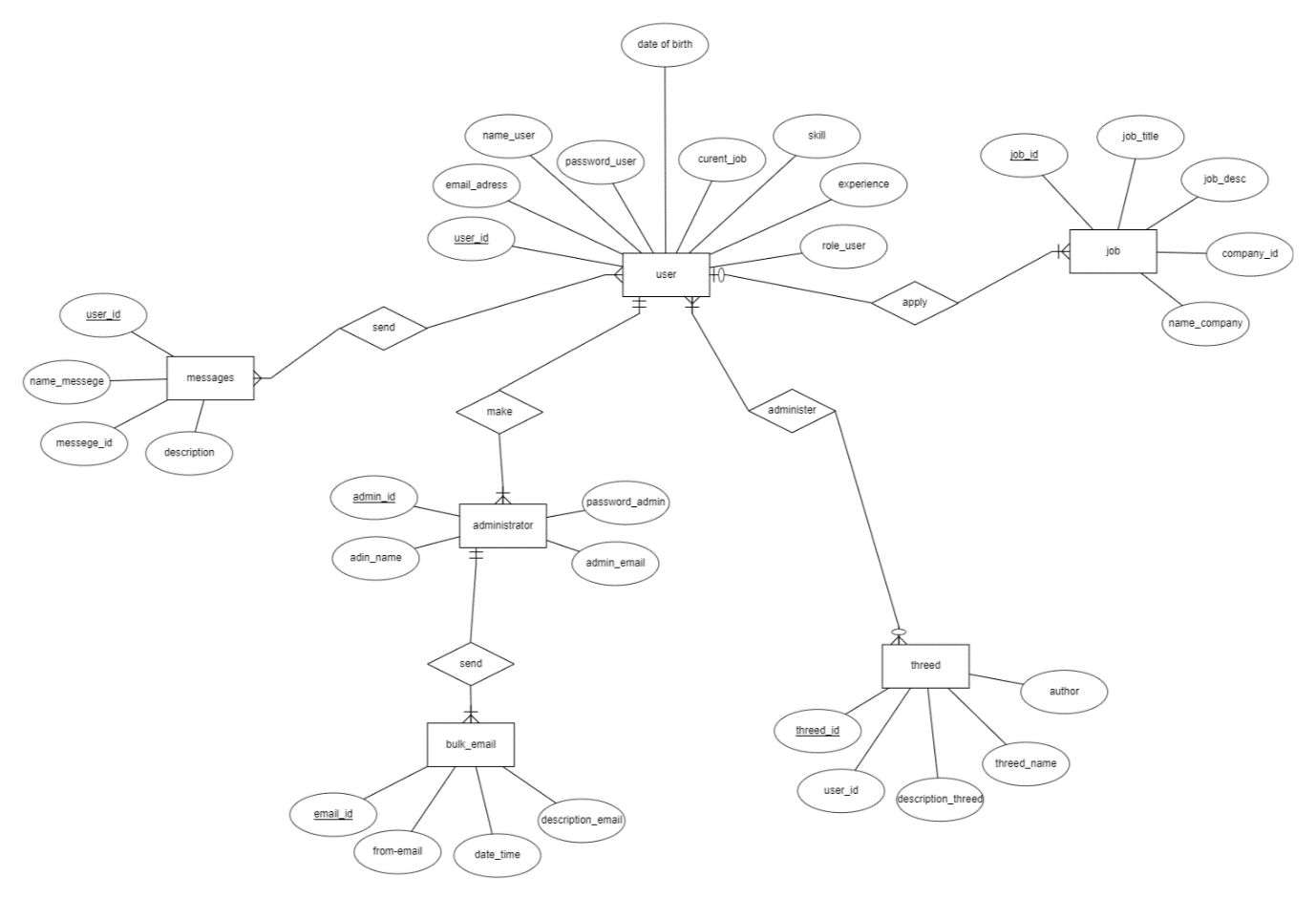
# 3. Create Logical Design for the proposed Database.

# 4. Create the ER Diagram for the project scenario.

# 5. Include the above as part of Project Presentation. (A2)

Solution

Describe what A software program called RDBMS, or Relational Database Management System, is used to store and administer databases. The database can be defined, constructed, and modified. A database management system (DBMS) is software that enables the creation, definition, and modification of databases, making it simple for users to store, handle, and analyze data.



Conceptual planning

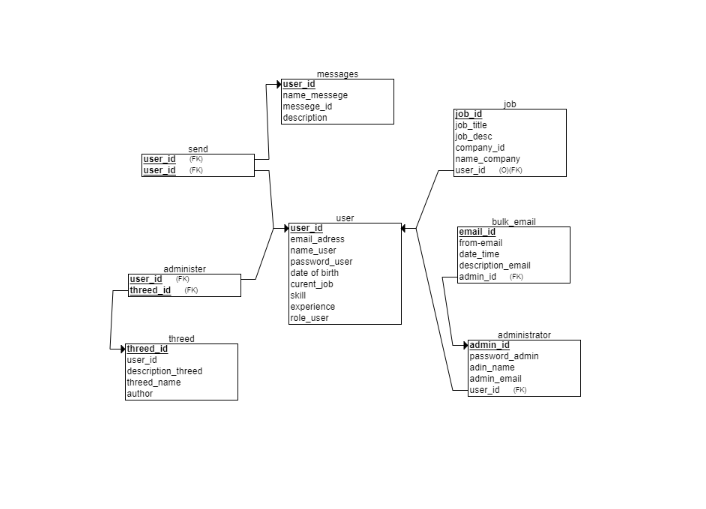
Constructing a conceptual model based on the developer's predetermined criteria is the aim of the conceptual design phase. a conceptual data model that explains entity data, the relationship between entities, and the relationship between attributes is the end result of conceptual design.

Diagram

Description automatically generated

Logical design

Logical database design is the process of creating an information model based on a specific data model but not exactly depend on database management.



Normalization

# What is 1NF?

1NF or the first normal form does not have repeated or similar data with a primary key unique to the entity. Each table cell should be unique and only have a single value.

# What is 2NF?

Table needs to be in 1NF and must have a relation of every non-primary key attribute to be fully dependent on the primary key. A table is in 2NF if it has no partial dependency and involve the removal of partial dependencies and placing them in a new relation

# What is 3NF?

Table must be in 2NF and has no transitive dependencies. Transitive dependency is when non-primary key attribute is dependent on another non-primary key attribute. 3NF involve the removal of transitive dependencies and placing then in a new relation.

# Normalization of user table

**1NF**

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Id | Email Id | Password | Name | Current Job | Location | Experience | Skill | Role |
| 1 | user1@email | \*\*\*\*\*\*\* | John anjay | Developer | New York | 5 years | Java | User |
| 2 | user2@email | \*\*\*\*\*\*\* | Jak danil | Engineer | London | 3 years | Python | Admin |
| 3 | user3@email | \*\*\*\*\*\*\* | Anjay mabar | Manager | Paris | 8 years | Css | User |

**2NF**

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| user\_id | email\_id | password | name | current\_job\_id | location\_id | experience | role |
| 1 | [user1@email.com](mailto:user1@email.com) | password123 | John Doe | 1 | 1 | 5 | admin |
| 2 | [user2@email.com](mailto:user2@email.com) | password456 | Jane Smith | 2 | 2 | 2 | user |
| 3 | [user3@email.com](mailto:user3@email.com) | password789 | Michael Johnson | 3 | 3 | 3 | moderato |

|  |  |
| --- | --- |
| current\_job\_id | current\_job |
| 1 | developer |
| 2 | designer |
| 3 | content writer |

|  |  |
| --- | --- |
| location\_id | location |
| 1 | New York |
| 2 | Los Angeles |
| 3 | San Francisco |

|  |  |
| --- | --- |
| user\_id | skill |
| 1 | programming |
| 1 | Python |
| 2 | design |
| 2 | Adobe Photoshop |
| 3 | writing |
| 3 | SEO |

Entities and attributes for community Portal

|  |  |
| --- | --- |
| Entities | Attributes |
| User | 1. user\_id (PK) 2. Email\_adress 3. Password\_user 4. Date of birth 5. Curet\_job 6. skill 7. experience 8. role\_user |
| Administrator | 1. Admin\_id 2. Password\_admin 3. Admin\_name 4. Admin\_email 5. User\_id (FK) |
| Bulk\_email | 1. Email\_id (PK) 2. From\_email 3. Date\_time 4. description 5. Admin\_id (FK) |
| Thread | 1. Thread\_id (PK) 2. User\_id 3. description 4. threed\_name 5. author |
| Messages | 1. user\_id (PK) 2. name\_messege 3. messege\_id 4. description |
| Job | 1. Job\_id (PK) 2. Job\_title 3. Job\_desc 4. Company\_id 5. Company\_name 6. User\_id |

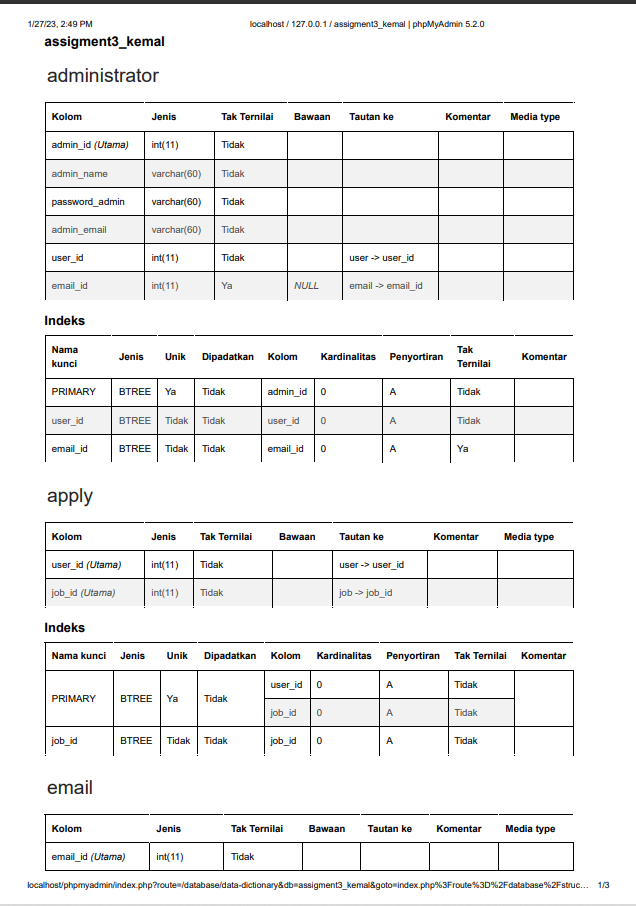
# Task 3

1. Create the Database dictionary with tables, fields & datatypes

2. Include the above as part of Project Report (A3)

Task Statement:

Create the database dictionary



Graphical user interface, application, table

Description automatically generated

Graphical user interface, application, table

Description automatically generated

# Task 4

1. Create a MySQL database & Implement the database design in that database.

2. Implement Primary Key, Foreign Key & Constraints.

3. Use SQL Scripts & Mysql or phpMyAdmin to create the database.

4. Produce the Screen capture of created tables in phpMyAdmin or MySQL Command prompt.

5. Include it as part of Project Presentation.

# Solution

Implement the database in MySQL using SQL scripts

**USER**

CREATE TABLE User

(

user\_id INT NOT NULL,

name\_user VARCHAR(60) NOT NULL,

email\_adress VARCHAR(60) NOT NULL,

password\_user VARCHAR(60) NOT NULL,

skill VARCHAR(60) NOT NULL,

date\_of\_birth DATE NOT NULL,

role\_user VARCHAR(60) NOT NULL,

experience VARCHAR(60) NOT NULL,

PRIMARY KEY (user\_id)

);

**Graphical user interface, application

Description automatically generated**

**ADMINISTRATORS**

CREATE TABLE administrator (

admin\_id INT AUTO\_INCREMENT PRIMARY KEY,

admin\_name VARCHAR(60) NOT NULL,

password\_admin VARCHAR(60) NOT NULL,

admin\_email VARCHAR(60) NOT NULL

);

Chart

Description automatically generated with medium confidence

**JOB**

CREATE TABLE Job

(

job\_id INT NOT NULL,

job\_title VARCHAR(60) NOT NULL,

job\_desc VARCHAR(60) NOT NULL,

company\_id INT NOT NULL,

nama\_company VARCHAR(60) NOT NULL,

location VARCHAR(60) NOT NULL,

PRIMARY KEY (job\_id),

UNIQUE (company\_id)

);

**Graphical user interface, application

Description automatically generated**

**EMAIL**

CREATE TABLE email

(

email\_id INT NOT NULL,

date\_time VARCHAR(60) NOT NULL,

from\_email VARCHAR(60) NOT NULL,

description\_email VARCHAR(60) NOT NULL,

PRIMARY KEY (email\_id)

);

**Graphical user interface

Description automatically generated**

**MESSAGE**

CREATE TABLE message

(

user\_id INT NOT NULL,

name\_message VARCHAR(60) NOT NULL,

description VARCHAR(60) NOT NULL,

messege\_id INT NOT NULL,

email\_id INT NOT NULL,

PRIMARY KEY (user\_id),

FOREIGN KEY (email\_id) REFERENCES User(email\_id),

UNIQUE (messege\_id)

);

**Graphical user interface, application, website

Description automatically generated**

**APPLY JOB**

CREATE TABLE apply

(

email\_id INT NOT NULL,

job\_id INT NOT NULL,

PRIMARY KEY (email\_id, job\_id),

FOREIGN KEY (email\_id) REFERENCES User(email\_id),

FOREIGN KEY (job\_id) REFERENCES Job(job\_id)

);

**Graphical user interface, application

Description automatically generated with medium confidence**

**THREAD**

CREATE TABLE thread (

thread\_id INT AUTO\_INCREMENT PRIMARY KEY,

author VARCHAR(255) NOT NULL,

thread\_name VARCHAR(255) NOT NULL,

description\_thread TEXT NOT NULL,

user\_id INT NOT NULL,

FOREIGN KEY (user\_id) REFERENCES user(user\_id)

);

**Graphical user interface, text, application

Description automatically generated**

EERD Diagram

Diagram

Description automatically generated

# Task 5

1. Create indexes in table, provide script for creating the index.

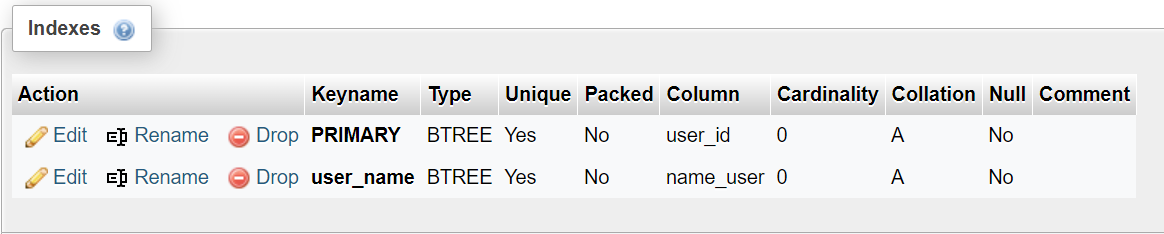
2. Create a Backup script to back up the database every 6 hours.

3. Include the mechanism and sources as part of Project Report

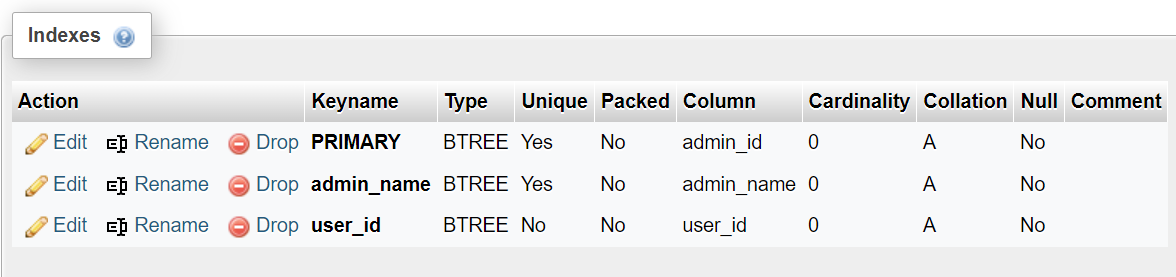
Task Statement:

Create indexes and backup script for daily backup

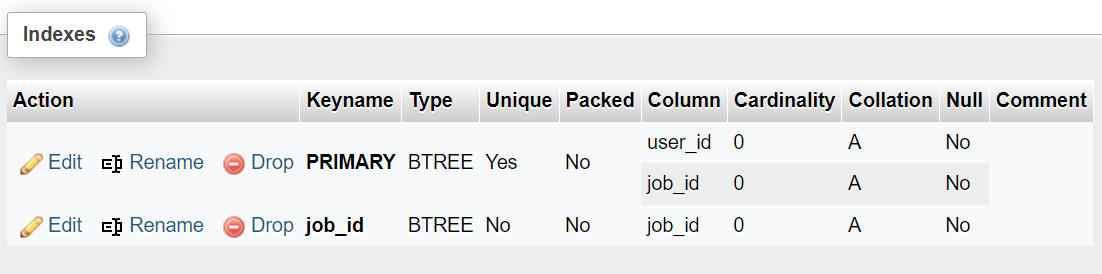
"ALTER TABLE `userr` ADD INDEX(`user\_name`);"



"ALTER TABLE `administrator` ADD INDEX(`admin\_name`);"



"ALTER TABLE `job` ADD INDEX(`job\_id`);"



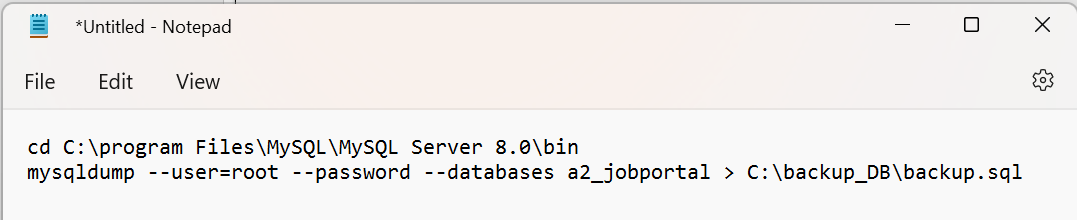
Backup your database

Choose the database.

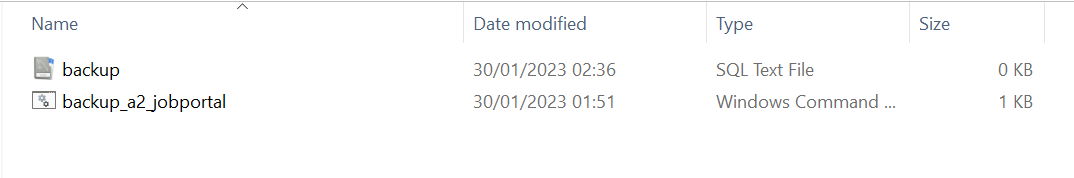


Create a new folder to store the backup data and than crate a script to backup data and save it as

.batch extensions in the folder.



This is the output



Now use windows task scheduler. Click on Create Task

Graphical user interface, text, application, email

Description automatically generated

Set the time to 07:00 every 7 days

Graphical user interface, application, Word

Description automatically generated

.Create a basic task and fill all the inputs. Once it is finished try to run the task and check the result

Graphical user interface, text, application, email

Description automatically generated

# Task 6

1. Create 8 SQL queries which will be used by the Community portal.

2. Import Sample data from CSV file in to the database.

3. Provide evidence of import as part of Project Presentation.

Task Statement:

Create query for the Community Portal and import sample data from CSV file

Solution

2 useful queries to develop the application

|  |  |  |  |
| --- | --- | --- | --- |
| No | Note | Query | Evidence |
| 1 | All users contact info.  To fetch users, contact info information | SELECT user\_id, email\_adress, skill, name\_user FROM user ORDER BY name\_user; | Report User Details |
| 2 | All jobs type info, To fetch job type information | SELECT job\_id, company\_name AS Company, job\_desc AS Description, job\_title AS job\_title  FROM job  ORDER BY job\_title; | Report jobs type log |

Report User Details



Report jobs type log

Graphical user interface, text, application, chat or text message

Description automatically generated

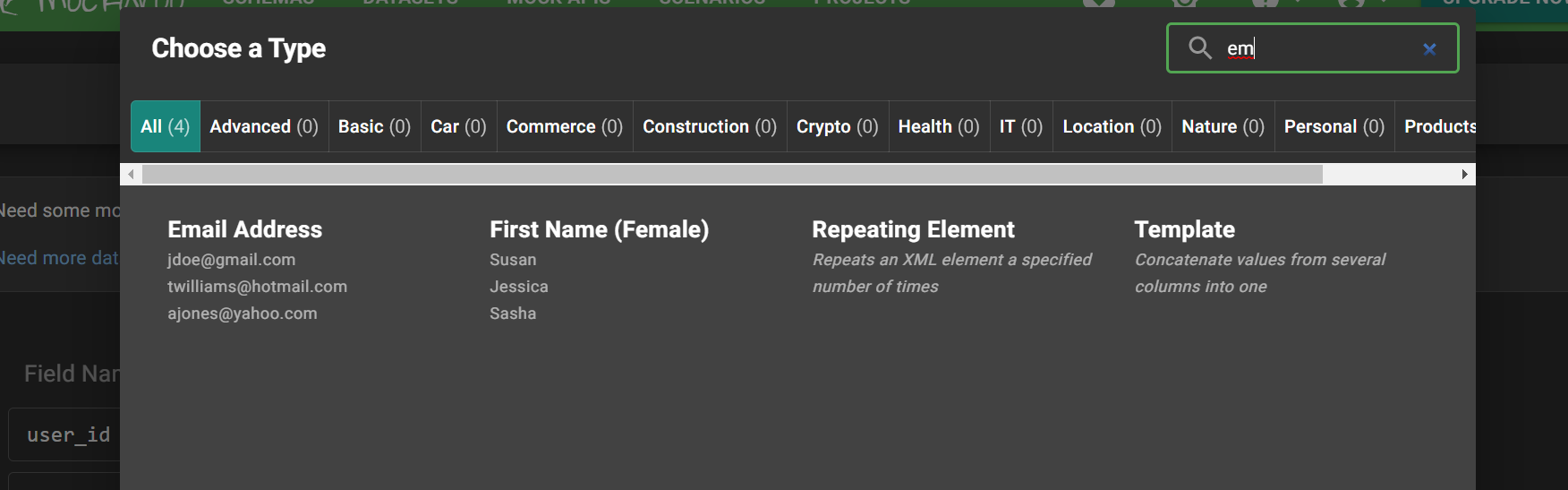
Steps to import CSV files

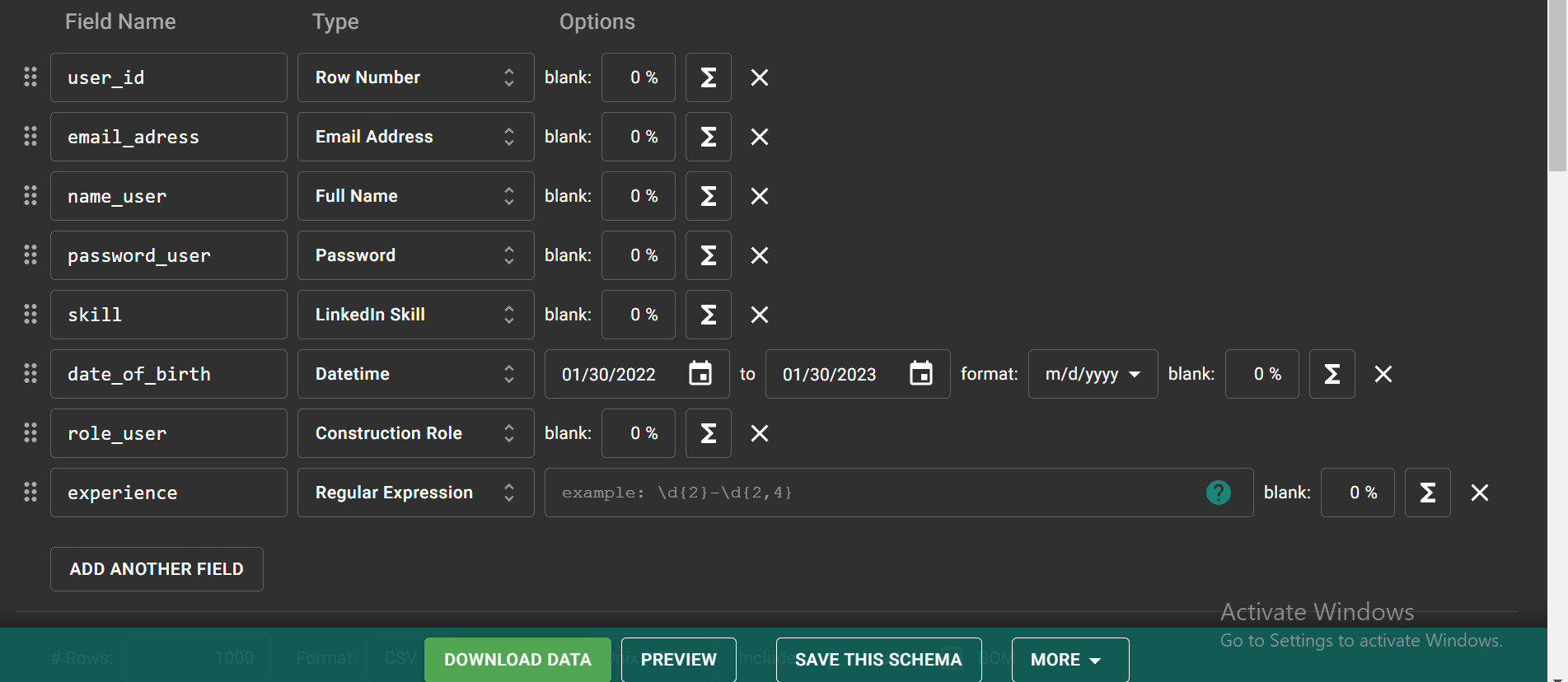
Open mockaroo.com

A screenshot of a computer

Description automatically generated

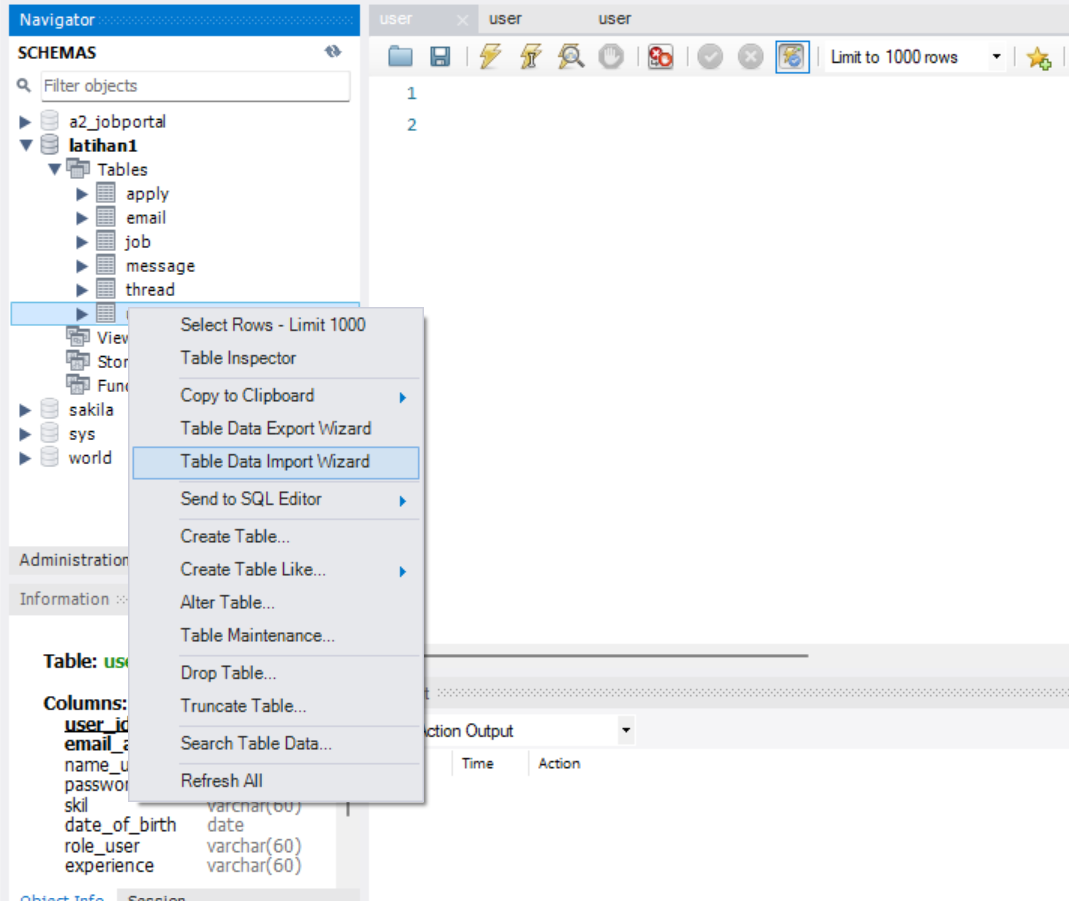
By clicking the X to delete the field or "Add another Filed" to add a new field, you can edit the field to meet your needs. Change the field's name and make it identical to your column table. Then select the field's type. You can search for and select a data type that is sufficiently similar to the sample data when you click the "Choose Data Type" button in the data type section.

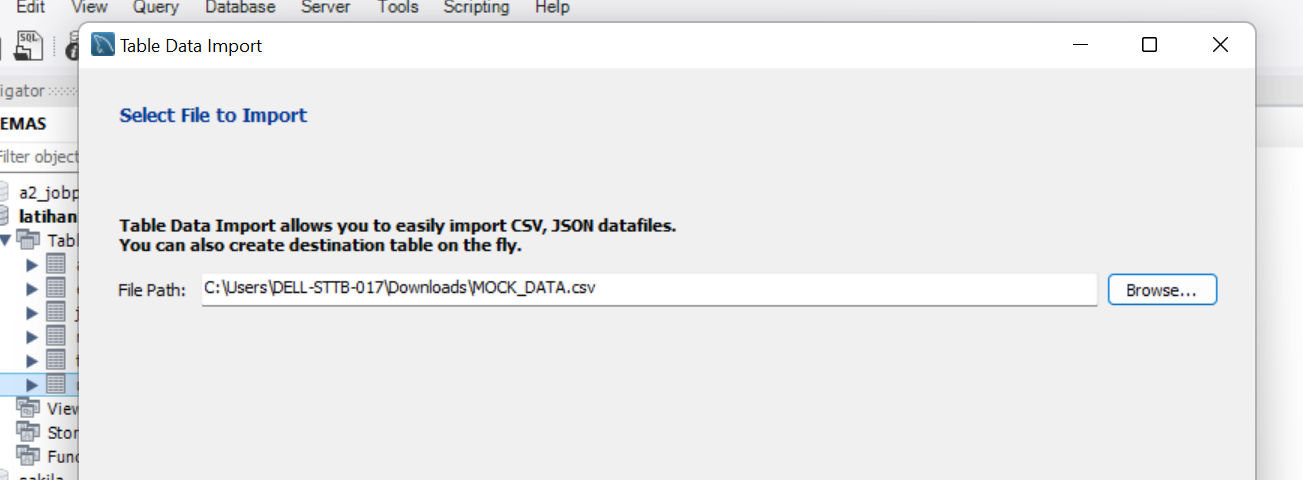




Set the number of random sample data you wish to make after that, select CSV as the file type, and then click "Download Data."

Open workbench and choose the database, and click in the table that you want to import





Click next

Graphical user interface, application, Word

Description automatically generated

Click next

Graphical user interface

Description automatically generated

Click next

Graphical user interface, text, application

Description automatically generated

Finished