# ASSESSMENT TASK: 3

|  |  |
| --- | --- |
| **Task:** | **3** |
| **Task Title:** | **Project** |
| **Task Code:** | **ICTPRG432 AT3 PROJECT** |

|  |  |  |  |
| --- | --- | --- | --- |
| Assessment type (❌): | | | |
| ☐ | Questioning (Oral/Written) | ☒ | Portfolio |
| ☐ | Practical Demonstration | ☐ | Project |
| ☐ | 3rd Party Report | ☐ | Other – Please Specify in space below |
|  |  |  | … |

|  |
| --- |
| Requirements for Assessment The base requirements this assessment task include:   * Web server, database server, DBMS management application. (We recommend Laragon with phpMyAdmin, and MariaDB.Others are also valid) * Access to Office 365 & Microsoft Word * Access to a plain text editor. (We recommend VS Code or similar)   Use of some of these items may not occur in this part of the assessment task. |

|  |
| --- |
| Assessment Due This assessment is to be completed and submitted before:   * Session 18 23:59 (11:59PM) on the day of the scheduled lecture/collaborate session.   Refer to Blackboard for most accurate dates, which may alter due to unforeseen circumstances.  We will also endeavour to update these document(s) at the same time. |
| Table of contents for this assessment task is shown on the following page. |

|  |
| --- |
| Table of Contents  [ASSESSMENT TASK: 3 1](#_Toc135294403)  [Assessment type (❌): 1](#_Toc135294404)  [Requirements for Assessment 1](#_Toc135294405)  [Assessment Due 1](#_Toc135294406)  [Instructions 5](#_Toc135294407)  [Important 5](#_Toc135294408)  [Scenario 5](#_Toc135294409)  [General Instructions 5](#_Toc135294410)  [Word Answer Document 5](#_Toc135294411)  [Answering Questions 6](#_Toc135294412)  [Word Limits 6](#_Toc135294413)  [SQL Answer File 7](#_Toc135294414)  [Writing SQL Code 8](#_Toc135294415)  [Sources of Information 8](#_Toc135294416)  [Code Storage 9](#_Toc135294417)  [Assessment Task Procedure 10](#_Toc135294418)  [Getting Prepared 10](#_Toc135294419)  [C#WPF Application where Sql is at backend and Xaml at front end 11](#_Toc135294420)  [An old company wants to upgrade their system and need to create application for their employees. These two tables are related to a Company data base that you created. 11](#_Toc135294421)  [ Employee 11](#_Toc135294422)  [ Branches 11](#_Toc135294423)  [ Working with 11](#_Toc135294424)  [ Clients Table 11](#_Toc135294425)  [ Branch Supplier Table 11](#_Toc135294426)  [Table Name 11](#_Toc135294427)  [Operations 11](#_Toc135294428)  [Priority 11](#_Toc135294429)  [Completion 11](#_Toc135294430)  [Create a WPF application for the company that would do the following tasks 13](#_Toc135294431)  [Create name, summary, and logo for your company 14](#_Toc135294432)  [Screenshot of the output/logo. 14](#_Toc135294433)  [Screenshot of the output (Executed code). 15](#_Toc135294434)  [Screenshot of the output (Executed code). 16](#_Toc135294435)  [Screenshot of the output (Executed code). 17](#_Toc135294436)  [Screenshot of the output (Executed code). 18](#_Toc135294437)  [e. Add a new employee 19](#_Toc135294438)  [Screenshot of the output (Executed code). 19](#_Toc135294439)  [f. Delete an employee record. 20](#_Toc135294440)  [Screenshot of the output (Executed code). 21](#_Toc135294441)  [Screenshot of the output (Executed code). 22](#_Toc135294442)  [02h 23](#_Toc135294443)  [Screenshot of the output (Executed code). 24](#_Toc135294444)  [Screenshot of the Main Window and other windows if created. 25](#_Toc135294445)  [Screenshot of the zipped file. 26](#_Toc135294446)  [Submission of Portfolio Work 29](#_Toc135294447)  [Appendix A: Scenario 31](#_Toc135294448)  [General Information 31](#_Toc135294449)  [Appendix B: Database Standards 32](#_Toc135294450)  [Database Naming 32](#_Toc135294451)  [Database User Naming 32](#_Toc135294452)  [Database Host 32](#_Toc135294453)  [Database Passwords 32](#_Toc135294454)  [Appendix C: Employees 33](#_Toc135294455)  [Sample Data 33](#_Toc135294456)  [Sample Table Design 33](#_Toc135294457)  [Sample SQL Create Command 34](#_Toc135294458)  [Sample SQL Seeding Commands 34](#_Toc135294459)  [Appendix D: Branches 35](#_Toc135294460)  [Sample Data 35](#_Toc135294461)  [Sample Table Design 35](#_Toc135294462)  [Sample SQL Create Command 35](#_Toc135294463)  [Sample SQL Insert Commands 35](#_Toc135294464)  [Appendix E: Clients Table 36](#_Toc135294465)  [Sample Data 36](#_Toc135294466)  [Sample Table Design 36](#_Toc135294467)  [Sample SQL Create Command 37](#_Toc135294468)  [Sample SQL Insert Commands 37](#_Toc135294469)  [Appendix F: Working With Table 38](#_Toc135294470)  [Sample Data 38](#_Toc135294471)  [Sample Table Design 38](#_Toc135294472)  [Sample SQL Create Command 39](#_Toc135294473)  [Sample SQL Insert Commands 39](#_Toc135294474)  [Appendix G: Branch Suppliers Table 40](#_Toc135294475)  [Sample Data 40](#_Toc135294476)  [Appendix Z: Example SQL Commands 41](#_Toc135294477) |

|  |
| --- |
| Instructions Follow the steps listed in this assessment item.  Submission of the documentation, code, and associated items is at the end of each part of the portfolio.  It is advantageous to you to attempt to meet the deadline provided. |

|  |
| --- |
| Important If you are using a different configuration of tools and equipment for this assessment item, then assistance in this and subsequent parts of the portfolio to ensure the systems work correctly will be limited. |

|  |
| --- |
| Scenario You are currently an intern with a small company based in Perth, Western Australia, called **Incredibly Obvious Technology** (IOT).  Please go to Appendix XXX to Appendix YYY and read the full scenario and data that is provided. |

|  |
| --- |
| General Instructions You will be completing TWO documents for submission:   * This MS Word file where you will add copies of Screen Shots in the provided spaces * A zipped file containing your C# WPF codes (Template is provided)   We require TWO files so that the lecturers and assessors can verify your work more effectively. |

|  |
| --- |
| Word Answer Document You have downloaded this document, next rename this document to:  XXX-ICTPRG432-AT3-PROJECT.docx  Replacing the XXX with your initials.  For example, Adrian Gould would use AJG-ICTPRG432-AT2-PROJECT.docx for his copy of this document’s filename. |

|  |
| --- |
| Answering Questions When a step includes a question, you must attempt to answer it.  All answers must be in complete sentences unless indicated.  The use of a sentence that leads into a list is appropriate if the question asks for a “list” or similar. |

|  |
| --- |
| SQL Answer File Rename the XXX-ICTPRG432-AT3-PROJECT.docx file by replacing the XXX with your initials. Adrian Gould would name his: AG-ICTPRG402-AT3-PROJECT.docx.  To edit the file, use a text editor or IDE such as HeidiSQL, Notepad++, Sublime Text, VS Code or similar. You may also use PyCharm to edit this file. We recommend using HeidiSQL for this task.  Make sure you have the following at the top of the SQL file with the correct details (Names, ID, Initials) filled out.  -- --------------------------------------------------------------------  -- Filename: XXX-ICTPRG432-AT2-POR-PtN.sql  -- Author: GIVEN & FAMIlY\_NAME  -- Email: ID@tafe.wa.edu.au  -- --------------------------------------------------------------------  -- Purpose:  -- This file contains the SQL used to create and execute  -- the solutions for the assessment ICTPRG402 Portfolio  -- --------------------------------------------------------------------  -- Declaration:  -- By completing and submitting this assessment  -- via the Blackboard LMS or other methods, to my  -- lecturer, I am stating that:  -- \* The attached submission is completely own work  -- \* I have correctly indicated all sources of information  -- used in this work (if required)  -- \* I have kept a copy of this assessment (where practicable)  -- \* I understand a copy of my assessment will be kept by  -- NMTAFE for their records  -- \* I understand my assessment may be selected for use in  -- NMTAFE’s validation and audit process to ensure student  -- assessment is valid and meets requirements of the unit  -- of competency  -- -------------------------------------------------------------------- |

|  |
| --- |
| Writing SQL Code When writing SQL code, you should NOT use any built-in features of phpMyAdmin or a similar SQL IDE, that let you create the code using a Graphical interface other than an SQL editor ‘page’.  For example, do not use the phpMyAdmin, MySQL Workbench, HeidiSQL or similar’s “Users” tab to create the user and database (see the picture below):  Graphical user interface  Description automatically generated  Instead, use the SQL or Query tab and enter the SQL to perform the action(s).  Below we show an example of using a SELECT query:  Graphical user interface, text, application  Description automatically generated  This is “Writing Code by Hand”.  You should expect to be asked questions that will be used to verify your ability to write and execute SQL correctly. These may or may not be questions in this assessment task. |

|  |
| --- |
| Sources of Information In industry, it is good practice to keep track of where information was obtained. This is especially true if it is a written document, or even code.  If you answer any questions using information from web sites, please include the site name and URL (Web site address) after the answer.  Likewise, include the title and author for books and magazine articles. For example:   * RS Electronics Ltd: <https://au.rs-online.com/> * Slack API Documentation, Users List Method: <https://api.slack.com/methods/users.list> |

|  |
| --- |
| Code Storage You MUST back-up the answer documents to your OneDrive storage provided by yourTAFE Office365 account, and it is advised to also back up to a Thumb drive, Flash drive, or External storage device. |

|  |
| --- |
| *This space left intentionally blank.* |

# Assessment Task Procedure

The following pages give you instructions and space to place answers, screen capture and other required responses.

Ensure screenshots are cropped and resized once placed in the space provided to enable assessor to review your work. This includes, when possible, making the content legible in the screen capture.

| STEP | | Task to perform |
| --- | --- | --- |
| 00 | Getting Prepared Ensure you have completed these steps before attempting ***ANY*** questions in this document:   * Read the ***whole*** document from start to end at least ***ONCE*** without doing any work on the assessment. * Downloaded and renamed this document as required. * Make sure you have followed the instructions on creating the answer document, as given in the General Instructions. * Make sure that you complete the title page of this document. * You have created and populated the following tables in the portfolio1 of the Assessment.  1. Employee Table 2. Branches Table 3. Clients Table 4. Working With Table 5. Branch Supplier Table   Get Ready to perform C # WPF operations with GUI to extract data from these tables. | |
| 00 | C#WPF Application where Sql is at backend and Xaml at front endAn old company wants to upgrade their system and need to create application for their employees. These tables are related to a company data base that you created.EmployeeBranchesWorking withClients TableBranch Supplier Table Your WPF application should show the following information, exercise your programming skills to create a user friendly and interactive program. (Using list box, text box or button etc... Make sure that you follow given table and perform according to priority   |  |  |  |  | | --- | --- | --- | --- | | Table Name | Operations | Priority | Completion | | Employee | List all the records  Search  Insert  Delete  Edit/Save  Join operation | Must be done |  | | Branches | List all the records  Search  Insert  Delete  Edit/Save  Join operation. | Optional |  | | Works With | List all the records  Search  Insert  Delete  Edit/Save  Join operation. | Optional |  | | Client | List all the records  Search  Insert  Delete  Edit/Save  Join operation. | Optional |  | | Branch Supplier | List all the records  Search  Insert  Delete  Edit/Save  Join operation. | Optional |  |  Create a WPF application for the company that would do the following tasks | |
| 01 | Create name, summary, and logo for your company | |
| 01 | Screenshot of the output/logo. | |
| 02a | 1. Show all employees information. | |
| 02a | Screenshot of the output (Executed code). Application searching for everything in employees on application loading: | |
| 02b | b. Search specific employee information by his/her name. For example: show the information/Salary for Kelly Kapoor. | |
| 02b | Screenshot of the output (Executed code). When clicking Search By Name With Kelly on the First name Text Box    Searches in DB for a first name in DB:    Output:    Can filter specific employee info by clicking filter by:      Output: | |
| 02c | c. Show all employees who work in the same branch. For example, user wants to see all employees in the branch number 2 | |
| 02c | Screenshot of the output (Executed code). Give a branch number:    **Grabs BranchId by 2:**   Output: | |
| 02d | d. Show all employees information who have salary more than $70000 | |
| 02d | Screenshot of the output (Executed code). Put the min and max Salaries on textboxes:    Searches a salary within range instead = more flexible:   Output: | |
| 02e | e. Add a new employee | |
| 02e | Screenshot of the output (Executed code). Fill out the form and press the add button:    Adds to Database and Employee table:    Output: | |
| 02f | f. Delete an employee record. Images on next page. | |
| 02f | Screenshot of the output (Executed code). Select an Employee and press the Delete Employee button:    Deletes Employee by unique employee ID:    Employee Disappears, Output: | |
| 02g | g. Show the employee sales using Works Table. For example, user want to see total sales for Michael Scott.  If he sales to different companies show the separate total sales(optional)  . | |
| 02g | Screenshot of the output (Executed code). Click Show Employees with sales will bring a new table:    Checks database for new info:    Output:    To show only Michael scott, put his id and click Show Sales By Employee ID:    Checks employee id = to 102    Output: | |
| 02h | h. Update/Save the salary of any of the employee. | |
| 02h | Screenshot of the output (Executed code). To update and save employee, select an employee, change any info you want(in this case, salary) and press the update button:      Updates database:    Outputs: | |

|  |  |
| --- | --- |
| 03 | Make your application user friendly with appropriate graphical things and colours. |
| 03 | Screenshot of the Main Window and other windows if created. |

|  |  |  |
| --- | --- | --- |
| 04 | 4.Send the app version and your C# code and Xaml code in the Zip format via Blackboard. |  |
| 04 | Screenshot of the zipped file.   Finished prj derived From v3 |  |

|  |  |  |
| --- | --- | --- |
|  |  | |
|  |  | |
|  |  | |
|  |  | |
| END | Submission of Portfolio Work To submit the portfolio, do the following:   * Save the document with your answers as a MS Word file (.docx). * Open Blackboard, and locate the ICTPRG432 Project assessment * Open the assessment: * Upload:   + Your word-processed answer document,   + Zipped program file * Click submit.   Whilst there is no need to use any other word processing software as you have access to Office 365 for free as a student, if you use Apple Pages, or Open Office, we will then require you to upload the original file **AND** a PDF version. | |
|  | | End of Assessment Task  **Following pages contain Appendices  with Scenario and Data for Portfolio** |

# Appendix A: Scenario

You are currently an intern with a small company based in Perth, Western Australia, called **Incredibly Obvious Technologies** (IOT).

As part of your internship with the company, they have set a task to help them ascertain your skills and capabilities.

They have a scenario that requires database and querying skills, and this is outlined in the remainder of this document.

## General Information

Incredibly Obvious Technologies are a multifaceted company who provide software development, 3D designs, and printing, and other skills to a wide variety of customers.

The scenario you are to work upon is a corporate database with details of employees, clients, branches, suppliers, and which clients work with which employees.

# Appendix B: Database Standards

IoT have strict naming conventions for their databases and tables.

These conventions MUST be adhered to.

## Database Naming

All databases are named in the form of database\_name.

For this scenario, the database will be named xxx\_company, or xxx\_ictprg432, where xxx are the intern’s initials.

## Database User Naming

All database users are named in the form of database\_name.

For this scenario, the database management system’s database specific username will be named xxx\_company, or xxx\_ictprg432, where xxx are the intern’s initials.

## Database Host

The database’s host computer will be the PC the intern is using and thus will be accessed via the “localhost” Server name during the period of development.

## Database Passwords

All database passwords are documented so that supervisors and the full-time developers may verify work and collaborate when appropriate.

The use of passwords is also to protect against other interns from plagiarising your work if they are completing the same task, now or during future internships.

During development, all database user passwords must conform with the following criteria:

|  |  |
| --- | --- |
| Minimum Length | 6 Characters |
| Maximum Length | 16 Characters |
| Characters must be from the following | ABCDEFGHIZJKLMNOPQRSTUVWXYZ abcdefghizjklmnopqrstuvwxyz 0123456789 !@#$%^&()\_+-={}[]<>,. |

# Appendix C: Employees

Sample data, table design and SQL to create the table and insert the data.

## Sample Data

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **Table** | Employee | | | | | | |
|  |  |  |  |  |  |  |  |
| **Employee ID** | **Given Name** | **Family Name** | **Date of Birth** | **Gender Identity** | **Gross Salary** | **Supervisor ID** | **Branch ID** |
| 100 | David | Wallace | 17/11/1967 | M | 25000 | Null | 1 |
| 101 | Jan | Levinson | 11/05/1967 | F | 110000 | 100 | 1 |
| 102 | Michael | Scott | 15/03/1964 | M | 75000 | 100 | 2 |
| 103 | Angela | Martin | 25/06/1971 | F | 63000 | 102 | 2 |
| 104 | Kelly | Kapoor | 05/02/1980 | F | 55000 | 102 | 2 |
| 105 | Stanley | Hudson | 19/02/1958 | M | 69000 | 102 | 2 |
| 106 | Josh | Porter | 05/09/1969 | M | 78000 | 100 | 3 |
| 107 | Andy | Bernard | 22/07/1973 | M | 65000 | 106 | 3 |
| 108 | Jen | Halpert | 01/10/1978 | F | 71000 | 106 | 3 |

## Sample Table Design

|  |  |
| --- | --- |
| Table: | Employees |
| Table Name | employees |

| Field | Type | Size | Options | Notes |
| --- | --- | --- | --- | --- |
| Id | Big Integer |  | Auto Increment  Unsigned  Primary Key |  |
| Given Name | Var Char | 64 | Nullable | May be empty for people with ONE name only |
| Family Name | Var Char | 64 | Not Null | Required |
| Date of Birth | Date |  | Not Null | Date format: YYYY-MM—DD  Default 1900-01-01 |
| Gender Identity | Character | 1 | Nullable |  |
| Gross Salary | Big Integer |  | Default 0 | Default to 0 |
| Supervisor ID | Big integer |  | Nullable | Has a default of 0 |
| Branch ID | Big Integer |  | Nullable | Default 0 |
| Created At | Timestamp |  | Not Null | Default 2022-07-01 |
| Updated At | Timestamp |  |  | Nullable  On update currenttimestamp |

## Sample SQL Create Command

CREATE TABLE `employees` (

`id` bigintUNSIGNED NOT NULL AUTO\_INCREMENT,

`given\_name` varchar(64),

`family\_name` varchar(64) NOT NULL,

`date\_of\_birth` date DEFAULT '1970-01-01',

`gender\_identity` char(1),

`gross\_salary` int DEFAULT '0',

`supervisor\_id` bigint DEFAULT '0',

`branch\_id` bigint DEFAULT '0',

`created\_at` timestamp DEFAULT '2022-07-01',

`updated\_at` timestampON UPDATE CURRENT\_TIMESTAMP,

PRIMARY KEY (id)

);

## Sample SQL Seeding Commands

INSERT INTO `employees`

(`date\_of\_birth`, `id`, `family\_name`, `branch\_id`,`supervisor\_id`, `given\_name`, `gross\_salary`, `gender\_identity`)

VALUES

('1967-05-11', 101, 'Levinson', 1, 100, 'Jan', 110000, 'F');

INSERT INTO `employees`

(`date\_of\_birth`, `id`, `family\_name`, `branch\_id`, `supervisor\_id`, `given\_name`, `gross\_salary`, `gender\_identity`)

VALUES

('1964-03-15', 102, 'Scott', 2, 100, 'Michael', 75000, 'O'),

('1971-06-25', 103, 'Martin', 2, 102, 'Angela', 63000, 'F'),

('1980-02-05', 104, 'Kapoor', 2, 102, 'Kelly', 55000, 'O'),

('1958-02-19', 105, 'Hudson', 2, 102, 'Stanley', 69000, 'M'),

('1969-09-05', 106, 'Porter', 3, 100, 'Josh', 78000, 'M'),

('1973-07-22', 107, 'Bernard', 3, 106, 'Andy', 65000, 'M'),

('1978-10-01', 108, 'Halpert', 3, 106, 'Jen', 71000, 'F');

# Appendix D: Branches

Sample data, table design and SQL to create the table and insert the data.

## Sample Data

|  |  |  |  |
| --- | --- | --- | --- |
| **Table** | Branches | | |
|  |  |  |  |
| **Branch ID** | **Branch Name** | **Manager ID** | **Manager Start Date** |
| 1 | Corporate | 100 | 09/02/2006 |
| 2 | Scranton | 102 | 06/04/1992 |
| 3 | Stamford | 106 | 13/02/1998 |

## Sample Table Design

|  |  |
| --- | --- |
| Table: | Branches |
| Table Name | branches |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Field | Type | Size | Options | Notes |
| Id | Big Integer |  | Auto Increment  Unsigned  Primary Key |  |
| Branch Name | Var Char | 64 | Not Null  Default ‘ERROR’ | Unique |
| Manager ID | Big integer |  | Default 0  Unsigned |  |
| Manager Start Date | Date |  | Not Null  Default 1900-01-01 | Date format: YYYY-MM-DD |
| Created At | Timestamp |  | Not Null  Now |  |
| Updated At | Timestamp |  | Nullable  On update currenttimestamp |  |

## Sample SQL Create Command

## Sample SQL Insert Commands

# Appendix E: Clients Table

Sample data, table design and SQL to create the table and insert the data.

## Sample Data

|  |  |  |
| --- | --- | --- |
| Table | Clients | |
|  |  |  |
| **Client ID** | **Client Name** | **Branch ID** |
| 400 | Dunmore Hoghschool | 2 |
| 401 | Lackawana Country | 2 |
| 402 | FedEx | 3 |
| 403 | John Daly Law, LLC | 3 |
| 404 | Scranton Whitepages | 2 |
| 405 | Times Newspaper | 3 |
| 406 | FedEx | 2 |

## Sample Table Design

|  |  |
| --- | --- |
| Table: | Staff Members |
| Table Name | staff\_members |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Field | Type | Size | Options | Notes |
| Id | Big Integer |  | AI, PK | Primary key |
| Name | Varchar | 32 | NN | Not null |
| Pay Rate | Decimal | 6,2 | Default 0.00 | Has a default of 0.00 |
|  |  |  |  |  |

## Sample SQL Create Command

CREATE TABLE `employees` (

`id` bigint UNSIGNED NOT NULL AUTO\_INCREMENT,

`given\_name` varchar(64),

`family\_name` varchar(64) NOT NULL,

`date\_of\_birth` date DEFAULT '1970-01-01',

`gender\_identity` char(1),

`gross\_salary` int DEFAULT '0',

`supervisor\_id` bigint DEFAULT '0',

`branch\_id` bigint DEFAULT '0',

`created\_at` timestamp DEFAULT '2022-07-01',

`updated\_at` timestampON UPDATE CURRENT\_TIMESTAMP,

PRIMARY KEY (id)

);

## Sample SQL Insert Commands

INSERT INTO `employees`

( `date\_of\_birth`, `id`, `family\_name`, `branch\_id`, `supervisor\_id`, `given\_name`, `gross\_salary`, `gender\_identity` )

VALUES

( '1967-05-11', 101, 'Levinson', 1, 100, 'Jan', 110000, 'F' );

INSERT INTO `employees`

(`date\_of\_birth`, `id`, `family\_name`, `branch\_id`, `supervisor\_id`, `given\_name`, `gross\_salary`, `gender\_identity`)

VALUES

('1964-03-15', 102, 'Scott', 2, 100, 'Michael', 75000, 'O'),

('1971-06-25', 103, 'Martin', 2, 102, 'Angela', 63000, 'F'),

('1980-02-05', 104, 'Kapoor', 2, 102, 'Kelly', 55000, 'O'),

('1958-02-19', 105, 'Hudson', 2, 102, 'Stanley', 69000, 'M'),

('1969-09-05', 106, 'Porter', 3, 100, 'Josh', 78000, 'M'),

('1973-07-22', 107, 'Bernard', 3, 106, 'Andy', 65000, 'M'),

('1978-10-01', 108, 'Halpert', 3, 106, 'Jen', 71000, 'F');

# Appendix F: Working With Table

Sample data, table design and SQL to create the table and insert the data.

## Sample Data

|  |  |  |
| --- | --- | --- |
| Table | Working With | |
|  |  |  |
| **Employee ID** | **Client ID** | **Total Sales** |
| 105 | 400 | 55000 |
| 102 | 401 | 267000 |
| 108 | 402 | 22500 |
| 107 | 403 | 5000 |
| 108 | 403 | 12000 |
| 105 | 404 | 33000 |
| 107 | 405 | 26000 |
| 102 | 406 | 15000 |
| 105 | 406 | 130000 |

## Sample Table Design

|  |  |
| --- | --- |
| Table: | Staff Members |
| Table Name | staff\_members |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Field | Type | Size | Options | Notes |
| Id | Big Integer |  | AI, PK | Primary key |
| Name | Varchar | 32 | NN | Not null |
| Pay Rate | Decimal | 6,2 | Default 0.00 | Has a default of 0.00 |
|  |  |  |  |  |

## Sample SQL Create Command

CREATE TABLE `employees` (

`id` bigint UNSIGNED NOT NULL AUTO\_INCREMENT,

`given\_name` varchar(64),

`family\_name` varchar(64) NOT NULL,

`date\_of\_birth` date DEFAULT '1970-01-01',

`gender\_identity` char(1),

`gross\_salary` int DEFAULT '0',

`supervisor\_id` bigint DEFAULT '0',

`branch\_id` bigint DEFAULT '0',

`created\_at` timestamp DEFAULT '2022-07-01',

`updated\_at` timestampON UPDATE CURRENT\_TIMESTAMP,

PRIMARY KEY (id)

);

## Sample SQL Insert Commands

INSERT INTO `employees`

( `date\_of\_birth`, `id`, `family\_name`, `branch\_id`, `supervisor\_id`, `given\_name`, `gross\_salary`, `gender\_identity` )

VALUES

( '1967-05-11', 101, 'Levinson', 1, 100, 'Jan', 110000, 'F' );

INSERT INTO `employees`

(`date\_of\_birth`, `id`, `family\_name`, `branch\_id`, `supervisor\_id`, `given\_name`, `gross\_salary`, `gender\_identity`)

VALUES

('1964-03-15', 102, 'Scott', 2, 100, 'Michael', 75000, 'O'),

('1971-06-25', 103, 'Martin', 2, 102, 'Angela', 63000, 'F'),

('1980-02-05', 104, 'Kapoor', 2, 102, 'Kelly', 55000, 'O'),

('1958-02-19', 105, 'Hudson', 2, 102, 'Stanley', 69000, 'M'),

('1969-09-05', 106, 'Porter', 3, 100, 'Josh', 78000, 'M'),

('1973-07-22', 107, 'Bernard', 3, 106, 'Andy', 65000, 'M'),

('1978-10-01', 108, 'Halpert', 3, 106, 'Jen', 71000, 'F');

# Appendix G: Branch Suppliers Table

Sample data only.

## Sample Data

|  |  |  |
| --- | --- | --- |
| **Table** | Branch Supplier | |
|  |  |  |
| **Branch ID** | **Supplier Name** | **Product Supplied** |
| 2 | Hammer Mill | Paper |
| 2 | Uni-Ball | Writing Instruments |
| 3 | Patriot Paper | Paper |
| 2 | J. T. Forms & Labels | Custom Forms |
| 3 | Uni-Ball | Writing Instruments |
| 3 | Hammer Mill | Paper |
| 3 | Stamford Labels | Custom Forms |

# Appendix Z: Example SQL Commands

The following are some sample SQL commands that may or may not be of use in your assessment task.

These commands are not guaranteed to provide any solutions, but they may provide hints to assist you.

CREATE DATABASE test\_dummy;

USE test\_dummy;

SHOW DATABASES;

CREATE USER 'test\_dummy\_user'@'localhost' IDENTIFIED BY 'Password1';

GRANT USAGE ON \*.\* TO 'test\_dummy\_user'@'localhost';

GRANT EXECUTE, SELECT, SHOW VIEW, ALTER, ALTER ROUTINE, CREATE, CREATE ROUTINE, CREATE TEMPORARY TABLES, CREATE VIEW, DELETE, DROP, EVENT, INDEX, INSERT, REFERENCES, TRIGGER, UPDATE, LOCK TABLES ON `test\\_dummy`.\* TO 'test\_dummy\_user'@'localhost' WITH GRANT OPTION;

GRANT ALL ON `test\\_dummy`.\* TO 'test\_dummy\_user'@'localhost' WITH GRANT OPTION;

FLUSH PRIVILEGES;

DROP DATABASE test\_dummy;

SELECT \* FROM products WHERE product\_name like “%paper%”;

CREATE TABLE dummy\_products(id BIGINT UNSIGNED AUTO\_INCREMENT, name STRING(192) NOT NULL DEFAULT “ERROR”, PRIMARY KEY(id));