

Durham College DATA1202: Data Analysis Tools

Assignment#1 (Data Storage)

Name: Chinedu Onyeka

Due Date: 6 Oct 2021

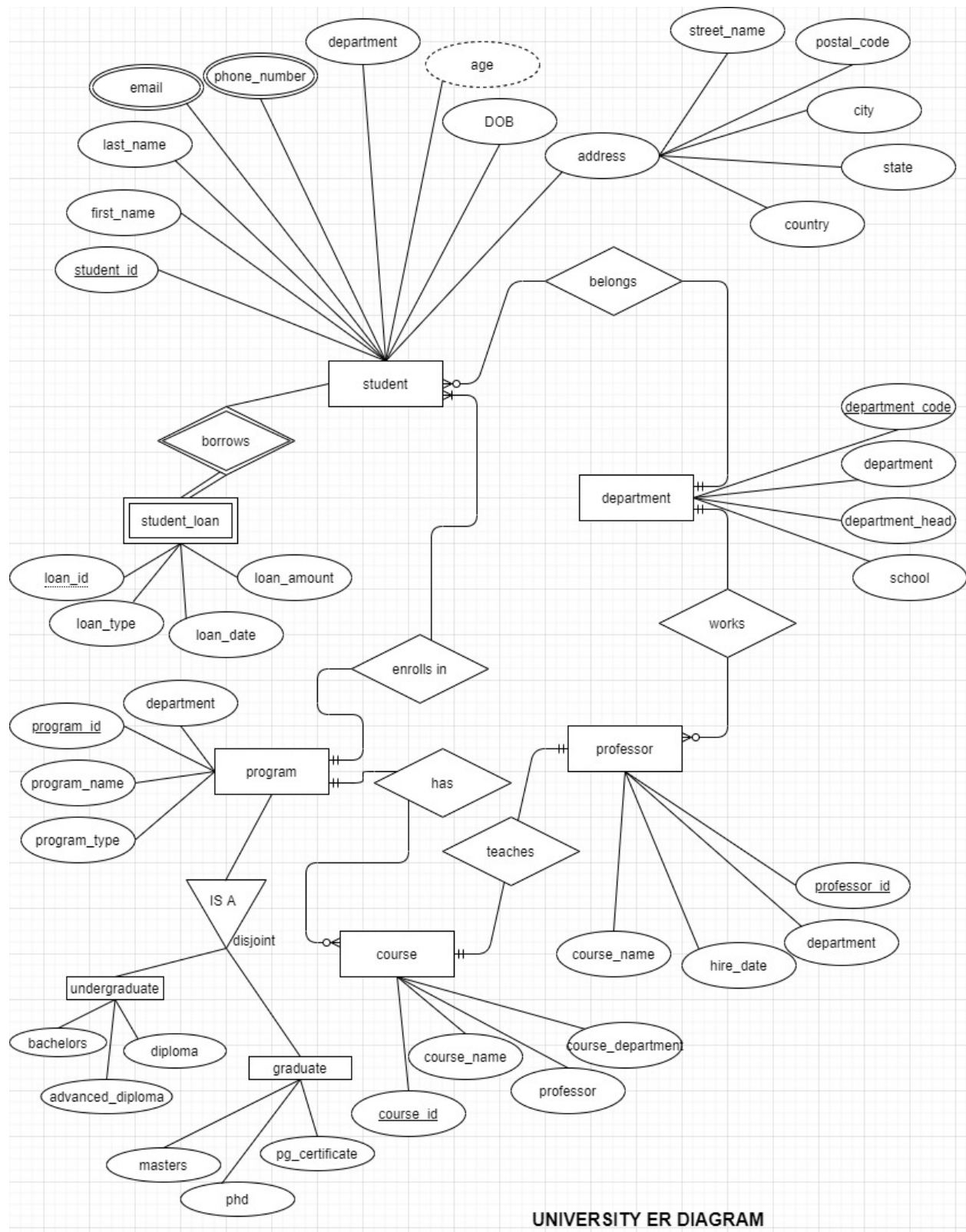
Part-A (10 points)

Create an ER Diagram for an **Airport OR Hospital OR University OR Bank**.

1. Identify atleast 5 entity in the ER Diagram
2. Identify atleast one weak entity.
3. Identify atleast 3 attributes for each entity.
4. Identify atleast one attribute that is:
 - Composite
 - Multivalued attribute
 - Derived attribute
5. Represent all the key(primary) attributes by suitable notations/symbol.
6. Represent the Cardinality and Modality using Crow's foot notation.
7. Identify atleast one specialization/generalization.

Solution:

Part A: ER diagram of a university.



Rubrics for Part-A

1. All the above-mentioned criteria are satisfied. (4 points)
2. Appropriate symbols are used in the ER Diagram. (2 points)
3. ER Diagram is specific and consistent with the Functional Requirements of the chosen domain. (1 Point)
4. ER Diagram presented without ambiguity and gives a complete picture of the selected domain. (2 Points)
5. Overall neatness. (1 Point)

Part-B (10 points)

Convert the following ER Diagram (Figure 1) into a database schema in implement it in SQLite db Browser. Add atleast two records in each of the table.

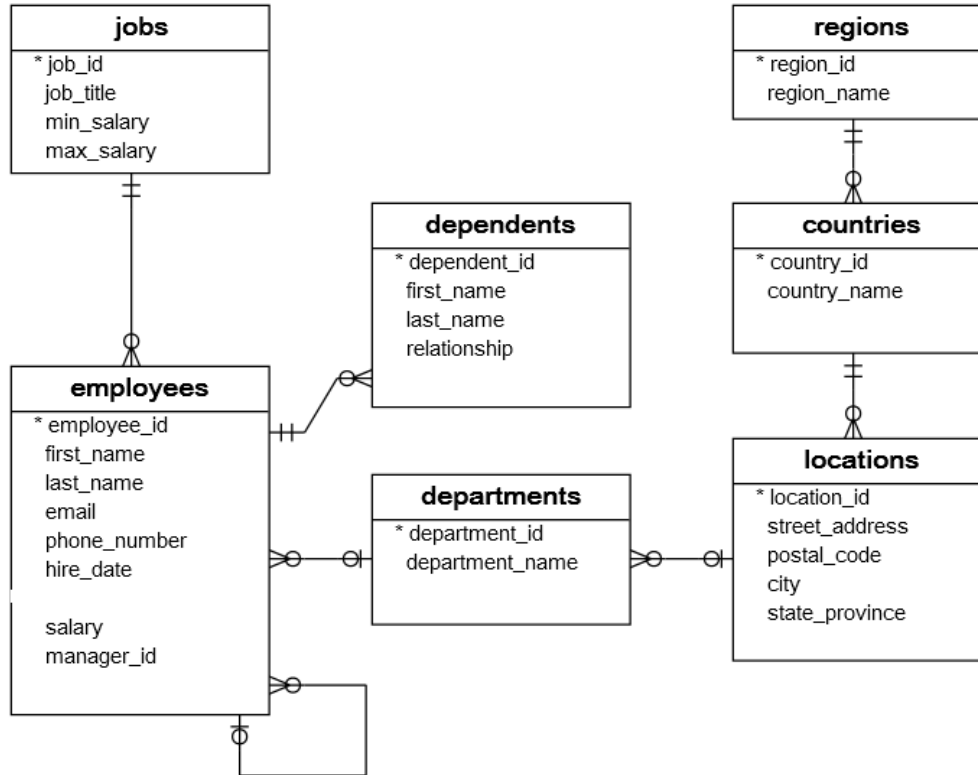


Figure-1

Rubrics for Part-B

1. All the above-mentioned entities and attributes are implemented. Student have submitted the evaluation in the format specified. (2 points)
2. Atleast two records are added into each of the table. (2 points)
3. Overall correctness of the Implementation including specification of Primary key, Foreign Key, Data Type and other constraints added. (4 points)
4. Student research about Recursive relationship and how it is implemented in the database. (2 points)

Note: The sql code for this part (Part B) can be found [here](#)

Screenshot 1 (Create the database and drop all pre-existing tables)

The screenshot shows the SQL Developer interface. The left sidebar contains the 'MANAGEMENT' tree with options like Server Status, Client Connections, Users and Privileges, Status and System Variables, Data Export, and Data Import/Restore. The 'INSTANCE' section shows Startup / Shutdown, Server Logs, and Options File. The 'PERFORMANCE' section shows Dashboard, Performance Reports, and Performance Schema Setup. The 'Administration' tab is selected, showing 'Schemas' and 'Information'. The 'Output' window at the bottom shows the execution of SQL code. The code includes comments and SQL statements to create a database and drop existing tables. The 'Output' window displays the results of the execution, showing the time taken for each statement and the number of rows affected.

```
1  /* Durham College Data Analytics for Business Decision Making
2  DATA1202: Data Analysis Tools; Assignment1 - PartB
3  Name: Chinedu Onyeka
4  */
5
6  # This converts an ER diagram to a schema
7
8  DROP DATABASE IF EXISTS dc_employees;
9  CREATE DATABASE IF NOT EXISTS dc_employees; # creates the database
10 USE dc_employees;
11 DROP TABLE IF EXISTS regions;
12 DROP TABLE IF EXISTS countries;
13 DROP TABLE IF EXISTS locations;
14 DROP TABLE IF EXISTS departments;
15 DROP TABLE IF EXISTS jobs;
16 DROP TABLE IF EXISTS employees;
17 DROP TABLE IF EXISTS dependents;
18
19 # Create the tables
20
21 /* Create the regions table
```

Automatic context help is disabled. Use the toolbar to manually get help for the current caret position or to toggle automatic help.

#	Time	Action	Message	Duration / Fetch
75	22:29:15	CREATE TABLE regions(region_id INT PRIMARY KEY AUTO_INCREMENT, region_name VA...	0 row(s) affected	0.016 sec
76	22:29:15	CREATE TABLE countries(country_id VARCHAR(10) PRIMARY KEY NOT NULL, country VA...	0 row(s) affected	0.031 sec
77	22:29:15	CREATE TABLE locations(location_id INT PRIMARY KEY AUTO_INCREMENT NOT NULL, s...	0 row(s) affected	0.032 sec
78	22:29:15	CREATE TABLE departments(department_id VARCHAR(20) PRIMARY KEY NOT NULL, depa...	0 row(s) affected	0.015 sec
79	22:29:15	CREATE TABLE jobs(job_id INT PRIMARY KEY NOT NULL, job_title VARCHAR(50) NOT NU...	0 row(s) affected	0.031 sec
80	22:29:15	CREATE TABLE employees(employee_id INT PRIMARY KEY NOT NULL, first_name VARCH...	0 row(s) affected	0.016 sec

Screenshot 2 (Create the regions and countries table)

The screenshot shows the SQL Developer interface. The left sidebar contains the 'MANAGEMENT' tree with options like Server Status, Client Connections, Users and Privileges, Status and System Variables, Data Export, and Data Import/Restore. The 'INSTANCE' section shows Startup / Shutdown, Server Logs, and Options File. The 'PERFORMANCE' section shows Dashboard, Performance Reports, and Performance Schema Setup. The 'Administration' tab is selected, showing 'Schemas' and 'Information'. The 'Output' window at the bottom shows the execution of SQL code. The code includes comments and SQL statements to create the regions and countries tables. The 'Output' window displays the results of the execution, showing the time taken for each statement and the number of rows affected.

```
19 # Create the tables
20
21 /* Create the regions table
22 regions(region_id(PK), region_name)
23 */
24 CREATE TABLE regions(
25 region_id INT PRIMARY KEY AUTO_INCREMENT,
26 region_name VARCHAR(50) NOT NULL
27 );
28
29
30 /* Create the countries table
31 countries(country_id(PK), country_name, region_id(FK))
32 */
33 CREATE TABLE countries(
34 country_id VARCHAR(10) PRIMARY KEY NOT NULL,
35 country VARCHAR(50) NOT NULL,
36 region_id INT NOT NULL REFERENCES regions(region_id) ON DELETE CASCADE ON UPDATE CASCADE
37 );
38
39
```

Automatic context help is disabled. Use the toolbar to manually get help for the current caret position or to toggle automatic help.

#	Time	Action	Message	Duration / Fetch
75	22:29:15	CREATE TABLE regions(region_id INT PRIMARY KEY AUTO_INCREMENT, region_name VA...	0 row(s) affected	0.016 sec
76	22:29:15	CREATE TABLE countries(country_id VARCHAR(10) PRIMARY KEY NOT NULL, country VA...	0 row(s) affected	0.031 sec
77	22:29:15	CREATE TABLE locations(location_id INT PRIMARY KEY AUTO_INCREMENT NOT NULL, s...	0 row(s) affected	0.032 sec
78	22:29:15	CREATE TABLE departments(department_id VARCHAR(20) PRIMARY KEY NOT NULL, depa...	0 row(s) affected	0.015 sec
79	22:29:15	CREATE TABLE jobs(job_id INT PRIMARY KEY NOT NULL, job_title VARCHAR(50) NOT NU...	0 row(s) affected	0.031 sec
80	22:29:15	CREATE TABLE employees(employee_id INT PRIMARY KEY NOT NULL, first_name VARCH...	0 row(s) affected	0.016 sec

Screenshot 3 (Create the locations and departments table)

The screenshot shows the SQL Developer interface with the following components:

- Left Panel:** Navigation tree with categories: MANAGEMENT (Server Status, Client Connections, Users and Privileges, Status and System Variables, Data Export, Data Import/Restore), INSTANCE (Startup / Shutdown, Server Logs, Options File), PERFORMANCE (Dashboard, Performance Reports, Performance Schema Setup), Administration (selected), and Schemas.
- Central Editor:** Contains SQL code for creating two tables:

```
40  /* Create the locations table
41  locations(location_id(PK), street_address, postal_code, city, state_province, country_id(FK))
42  */
43  CREATE TABLE locations(
44  location_id INT PRIMARY KEY AUTO_INCREMENT NOT NULL,
45  street_address VARCHAR(200) NOT NULL,
46  postal_code VARCHAR(10) NOT NULL,
47  city VARCHAR(50) NOT NULL,
48  state_province VARCHAR(50) NOT NULL,
49  country_id VARCHAR(10) NOT NULL REFERENCES countries(country_id) ON DELETE CASCADE ON UPDATE CASCADE
50  );
51
52
53  /* Create departments table
54  departments(department_id(PK), department_name, location_id(FK))
55  */
56  CREATE TABLE departments(
57  department_id VARCHAR(20) PRIMARY KEY NOT NULL,
58  department_name VARCHAR(100) NOT NULL,
59  location_id INT NOT NULL REFERENCES locations(location_id) ON DELETE CASCADE ON UPDATE CASCADE
60  );
```
- Right Panel:** Context Help tab showing a message: "Automatic context help is disabled. Use the toolbar to manually get help for the current caret position or to toggle automatic help."
- Bottom Panel:** Output window showing the execution results of the SQL script. The table below represents the data shown in the screenshot.

#	Time	Action	Message	Duration / Fetch
75	22:29:15	CREATE TABLE regions(region_id INT PRIMARY KEY AUTO_INCREMENT, region_name VA...	0 row(s) affected	0.016 sec
76	22:29:15	CREATE TABLE countries(country_id VARCHAR(10) PRIMARY KEY NOT NULL, country VA...	0 row(s) affected	0.031 sec
77	22:29:15	CREATE TABLE locations(location_id INT PRIMARY KEY AUTO_INCREMENT NOT NULL, s...	0 row(s) affected	0.032 sec
78	22:29:15	CREATE TABLE departments(department_id VARCHAR(20) PRIMARY KEY NOT NULL, depa...	0 row(s) affected	0.015 sec
79	22:29:15	CREATE TABLE jobs(job_id INT PRIMARY KEY NOT NULL, job_title VARCHAR(50) NOT NU...	0 row(s) affected	0.031 sec
80	22:29:15	CREATE TABLE employees(employee_id INT PRIMARY KEY NOT NULL, first_name VARCH...	0 row(s) affected	0.016 sec

Screenshot 4 (Create the jobs table)

The screenshot shows the SQL Developer interface with the following components:

- Left Panel:** Same as Screenshot 3, with the 'Administration' tab selected.
- Central Editor:** Contains SQL code for creating the jobs table:

```
62
63  /* Create jobs table
64  jobs(job_id(PK), job_title, min_salary, max_salary)
65  */
66  CREATE TABLE jobs(
67  job_id INT PRIMARY KEY NOT NULL,
68  job_title VARCHAR(50) NOT NULL,
69  min_salary INT NOT NULL,
70  max_salary INT NOT NULL
71  );
72
```
- Right Panel:** Context Help tab showing the same message as in Screenshot 3.
- Bottom Panel:** Output window showing the execution results. The table below represents the data shown in the screenshot.

#	Time	Action	Message	Duration / Fetch
75	22:29:15	CREATE TABLE regions(region_id INT PRIMARY KEY AUTO_INCREMENT, region_name VA...	0 row(s) affected	0.016 sec
76	22:29:15	CREATE TABLE countries(country_id VARCHAR(10) PRIMARY KEY NOT NULL, country VA...	0 row(s) affected	0.031 sec
77	22:29:15	CREATE TABLE locations(location_id INT PRIMARY KEY AUTO_INCREMENT NOT NULL, s...	0 row(s) affected	0.032 sec
78	22:29:15	CREATE TABLE departments(department_id VARCHAR(20) PRIMARY KEY NOT NULL, depa...	0 row(s) affected	0.015 sec
79	22:29:15	CREATE TABLE jobs(job_id INT PRIMARY KEY NOT NULL, job_title VARCHAR(50) NOT NU...	0 row(s) affected	0.031 sec
80	22:29:15	CREATE TABLE employees(employee_id INT PRIMARY KEY NOT NULL, first_name VARCH...	0 row(s) affected	0.016 sec

Screenshot 5 (Create the employees table)

The screenshot shows the SQL Developer interface with the 'employees' table creation script. The script includes comments and SQL code for creating the table with its columns and constraints. The output window shows the execution results for the 'employees' table creation.

```
73
74
75  /* Create employees table
76  employees(employee_id(PK), first_name, last_name, email, phone_number, hire_date, job_id(FK), salary, manager_id(FK))
77  */
78  CREATE TABLE employees(
79    employee_id INT PRIMARY KEY NOT NULL,
80    first_name VARCHAR(100) NOT NULL,
81    last_name VARCHAR(100) NOT NULL,
82    email VARCHAR(100) UNIQUE NOT NULL,
83    phone_number VARCHAR(20) UNIQUE NOT NULL,
84    hire_date VARCHAR(50) NOT NULL,
85    job_id INT NOT NULL REFERENCES jobs(job_id) ON DELETE CASCADE ON UPDATE CASCADE,
86    salary VARCHAR(20) NOT NULL,
87    manager_id INT REFERENCES employees(employee_id) ON DELETE CASCADE ON UPDATE CASCADE
88  );
89
90  /* Create dependents table
91  dependents(dependent_id(PK), first_name, last_name, relationship, employees(FK))
92  */
93  CREATE TABLE dependents(
```

Output:

#	Time	Action	Message	Duration / Fetch
75	22:29:15	CREATE TABLE regions(region_id INT PRIMARY KEY AUTO_INCREMENT, region_name VA...	0 row(s) affected	0.016 sec
76	22:29:15	CREATE TABLE countries(country_id VARCHAR(10) PRIMARY KEY NOT NULL, country VA...	0 row(s) affected	0.031 sec
77	22:29:15	CREATE TABLE locations(location_id INT PRIMARY KEY AUTO_INCREMENT NOT NULL, s...	0 row(s) affected	0.032 sec
78	22:29:15	CREATE TABLE departments(department_id VARCHAR(20) PRIMARY KEY NOT NULL, depa...	0 row(s) affected	0.015 sec
79	22:29:15	CREATE TABLE jobs(job_id INT PRIMARY KEY NOT NULL, job_title VARCHAR(50) NOT NU...	0 row(s) affected	0.031 sec
80	22:29:15	CREATE TABLE employees(employee_id INT PRIMARY KEY NOT NULL, first_name VARCH...	0 row(s) affected	0.016 sec

Screenshot 6 (Create the dependents table)

The screenshot shows the SQL Developer interface with the 'dependents' table creation script. The script includes comments and SQL code for creating the table with its columns and constraints. The output window shows the execution results for the 'dependents' table creation.

```
82    phone_number VARCHAR(20) UNIQUE NOT NULL,
83    hire_date VARCHAR(50) NOT NULL,
84    job_id INT NOT NULL REFERENCES jobs(job_id) ON DELETE CASCADE ON UPDATE CASCADE,
85    salary VARCHAR(20) NOT NULL,
86    manager_id INT REFERENCES employees(employee_id) ON DELETE CASCADE ON UPDATE CASCADE
87  );
88
89
90  /* Create dependents table
91  dependents(dependent_id(PK), first_name, last_name, relationship, employees(FK))
92  */
93  CREATE TABLE dependents(
94    dependent_id INT PRIMARY KEY AUTO_INCREMENT NOT NULL,
95    first_name VARCHAR(100) NOT NULL,
96    last_name VARCHAR(100) NOT NULL,
97    relationship VARCHAR(50) NOT NULL,
98    employee_id INT NOT NULL REFERENCES employees(employee_id) ON DELETE CASCADE ON UPDATE CASCADE
99  );
100
101
102  ### Insert at least two (2) records in each table
```

Output:

#	Time	Action	Message	Duration / Fetch
76	22:29:15	CREATE TABLE countries(country_id VARCHAR(10) PRIMARY KEY NOT NULL, country VA...	0 row(s) affected	0.031 sec
77	22:29:15	CREATE TABLE locations(location_id INT PRIMARY KEY AUTO_INCREMENT NOT NULL, s...	0 row(s) affected	0.032 sec
78	22:29:15	CREATE TABLE departments(department_id VARCHAR(20) PRIMARY KEY NOT NULL, depa...	0 row(s) affected	0.015 sec
79	22:29:15	CREATE TABLE jobs(job_id INT PRIMARY KEY NOT NULL, job_title VARCHAR(50) NOT NU...	0 row(s) affected	0.031 sec
80	22:29:15	CREATE TABLE employees(employee_id INT PRIMARY KEY NOT NULL, first_name VARCH...	0 row(s) affected	0.016 sec
81	22:29:15	CREATE TABLE dependents(dependent_id INT PRIMARY KEY AUTO_INCREMENT NOT N...	0 row(s) affected	0.015 sec

Screenshot 7 (Insert values into the regions and countries table)

The screenshot shows the SQL Developer interface with a script editor containing SQL commands to populate the 'regions' and 'countries' tables. The script includes comments and SQL statements for inserting data. The output window at the bottom shows the execution results for these tables.

Script Editor Content:

```
100  
101  
102 ### Insert at least two (2) records in each table  
103  
104 # Populate the regions table  
105  
106 # Populate the regions table  
107 INSERT INTO regions  
108 VALUES  
109 (1, "North America"),  
110 (2, "Europe"),  
111 (3, "Asia"),  
112 (4, "Africa");  
113  
114 # Populate the countries table  
115 INSERT INTO countries  
116 VALUES  
117 ( "CAN", "Canada", 1),  
118 ( "IND", "India", 3),  
119 ( "NGN", "Nigeria", 4);  
120
```

Output Window:

#	Time	Action	Message	Duration / Fetch
81	22:29:15	CREATE TABLE dependents (dependent_id INT PRIMARY KEY AUTO_INCREMENT NOT N...	0 row(s) affected	0.015 sec
82	22:29:15	INSERT INTO regions VALUES (1, "North America"), (2, "Europe"), (3, "Asia"), (4, "Africa")	4 row(s) affected Records: 4 Duplicates: 0 Warnings: 0	0.000 sec
83	22:29:15	INSERT INTO countries VALUES ("CAN", "Canada", 1), ("IND", "India", 3), ("NGN", "Nigeria"...	3 row(s) affected Records: 3 Duplicates: 0 Warnings: 0	0.000 sec
84	22:29:15	INSERT INTO locations VALUES (1, "1265 King Street East", "L1H 1J2", "Oshawa", "ON", "C...	3 row(s) affected Records: 3 Duplicates: 0 Warnings: 0	0.016 sec
85	22:29:15	INSERT INTO departments VALUES (1, "Chemistry", 1), (2, "Data Analytics", 2), (3, "Biology", 3)	3 row(s) affected Records: 3 Duplicates: 0 Warnings: 0	0.000 sec
86	22:29:15	INSERT INTO jobs VALUES (1, "Associate", 40000, 70000), (2, "Data Scientist", 65000, 1100...	3 row(s) affected Records: 3 Duplicates: 0 Warnings: 0	0.000 sec

Screenshot 8 (Insert values into the locations, departments, and jobs table)

The screenshot shows the SQL Developer interface with a script editor containing SQL commands to populate the 'locations', 'departments', and 'jobs' tables. The script includes comments and SQL statements for inserting data. The output window at the bottom shows the execution results for these tables.

Script Editor Content:

```
121 # Populate the locations table  
122 INSERT INTO locations  
123 VALUES  
124 (1, "1265 King Street East", "L1H 1J2", "Oshawa", "ON", "CAN"),  
125 (2, "10 Nwobodo Street", "410001", "Abuja", "FCI", "NGN"),  
126 (3, "573 Gupta Road", "L58 IND", "Delhi", "Delhi", "IND");  
127  
128 # Populate departments table  
129 INSERT INTO departments  
130 VALUES  
131 (1, "Chemistry", 1),  
132 (2, "Data Analytics", 2),  
133 (3, "Biology", 3);  
134  
135 # Populate the jobs table  
136 INSERT INTO jobs  
137 VALUES  
138 (1, "Associate", 40000, 70000),  
139 (2, "Data Scientist", 65000, 110000),  
140 (3, "Engineer", 45000, 95000);  
141
```

Output Window:

#	Time	Action	Message	Duration / Fetch
83	22:29:15	INSERT INTO countries VALUES ("CAN", "Canada", 1), ("IND", "India", 3), ("NGN", "Nigeria"...	3 row(s) affected Records: 3 Duplicates: 0 Warnings: 0	0.000 sec
84	22:29:15	INSERT INTO locations VALUES (1, "1265 King Street East", "L1H 1J2", "Oshawa", "ON", "C...	3 row(s) affected Records: 3 Duplicates: 0 Warnings: 0	0.016 sec
85	22:29:15	INSERT INTO departments VALUES (1, "Chemistry", 1), (2, "Data Analytics", 2), (3, "Biology", 3)	3 row(s) affected Records: 3 Duplicates: 0 Warnings: 0	0.000 sec
86	22:29:15	INSERT INTO jobs VALUES (1, "Associate", 40000, 70000), (2, "Data Scientist", 65000, 1100...	3 row(s) affected Records: 3 Duplicates: 0 Warnings: 0	0.000 sec
87	22:29:15	INSERT INTO employees VALUES (1, "Noopa", "Gandhi", "noopagandhi@gmail.com", 905278...	4 row(s) affected Records: 4 Duplicates: 0 Warnings: 0	0.000 sec
88	22:29:15	INSERT INTO dependents VALUES (1, "Vicky", "Oryx", "Child", 3), (2, "Jeremie", "Gtair", "B...	3 row(s) affected Records: 3 Duplicates: 0 Warnings: 0	0.016 sec

Screenshot 9 (Insert values into the employees and dependents table)

The screenshot shows the SQL Developer interface with the following SQL commands in the main editor:

```

136 • INSERT INTO jobs
137 VALUES
138 (1, "Associate", 40000, 70000),
139 (2, "Data Scientist", 65000, 110000),
140 (3, "Engineer", 45000, 95000);
141
142 #Populate the employees table
143 • INSERT INTO employees
144 VALUES
145 (1, "Noopa", "Gandhi", "noopagandhi@gmail.com", 9052780124, 9/7/2005, 2, 105000, NULL),
146 (2, "Prati", "Patel", "pratipatel@gmail.gov", 7284012312, 8/17/2019, 3, 50000, 1),
147 (3, "John", "Wickland", "johnwickland@gmail.com", 9812348954, 8/15/2020, 1, 45000, 1),
148 (4, "Joshua", "Vinny", "joshuvinny@mail.ca", 8203459876, 9/23/2021, 1, 41000, 2);
149
150 # Populate the dependents table
151 • INSERT INTO dependents
152 VALUES
153 (1, "Vicky", "Onyx", "Child", 3),
154 (2, "Jeremie", "Gitsim", "Brother", 4),
155 (3, "Pratish", "Pushkar", "Spouse", 1);
156

```

The Output window shows the results of the previous SQL statements:

#	Time	Action	Message	Duration / Fetch
83	22:29:15	INSERT INTO countries VALUES ("CAN", "Canada", 1), ("IND", "India", 3), ("NGN", "Nigeria")...	3 row(s) affected Records: 3 Duplicates: 0 Warnings: 0	0.000 sec
84	22:29:15	INSERT INTO locations VALUES (1, "1265 King Street East", "L1H 1J2", "Oshawa", "ON", "C...")...	3 row(s) affected Records: 3 Duplicates: 0 Warnings: 0	0.016 sec
85	22:29:15	INSERT INTO departments VALUES (1, "Chemistry", 1), (2, "Data Analytics", 2), (3, "Biology", 3)	3 row(s) affected Records: 3 Duplicates: 0 Warnings: 0	0.000 sec
86	22:29:15	INSERT INTO jobs VALUES (1, "Associate", 40000, 70000), (2, "Data Scientist", 65000, 110000)...	3 row(s) affected Records: 3 Duplicates: 0 Warnings: 0	0.000 sec
87	22:29:15	INSERT INTO employees VALUES (1, "Noopa", "Gandhi", "noopagandhi@gmail.com", 9052780124, 9/7/2005, 2, 105000, NULL)...	4 row(s) affected Records: 4 Duplicates: 0 Warnings: 0	0.000 sec
88	22:29:15	INSERT INTO dependents VALUES (1, "Vicky", "Onyx", "Child", 3), (2, "Jeremie", "Gitsim", "Brother", 4), (3, "Pratish", "Pushkar", "Spouse", 1);	3 row(s) affected Records: 3 Duplicates: 0 Warnings: 0	0.016 sec

Screenshot 10 (Select Statement: Select all columns from employees table)

The screenshot shows the SQL Developer interface with the following SQL command in the main editor:

```

154 (2, "Jeremie", "Gitsim", "Brother", 4),
155 (3, "Pratish", "Pushkar", "Spouse", 1);
156
157 # Select the tables
158
159 • SELECT * FROM employees;

```

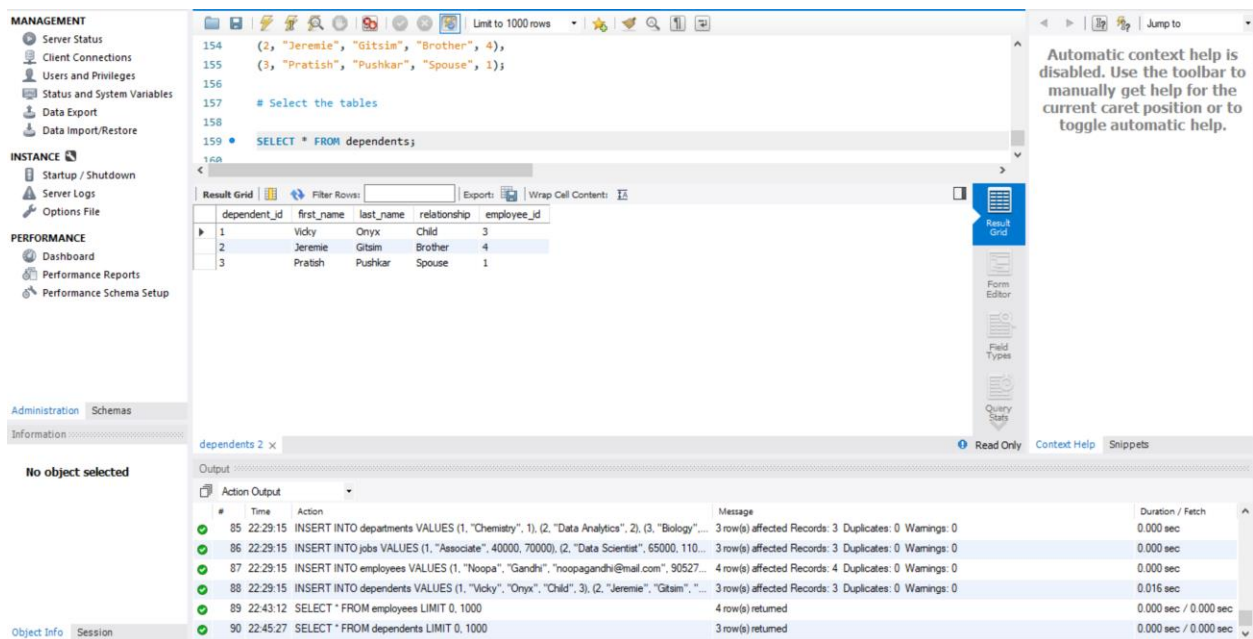
The Result Grid shows the data from the employees table:

employee_id	first_name	last_name	email	phone_number	hire_date	job_id	salary	manager_id
1	Noopa	Gandhi	noopagandhi@gmail.com	9052780124	0.000641254007481296	2	105000	NULL
2	Prati	Patel	pratipatel@gmail.gov	7284012312	0.000233079858841010	3	50000	1
3	John	Wickland	johnwickland@gmail.com	9812348954	0.000264026402475247	1	45000	1
4	Joshua	Vinny	joshuvinny@mail.ca	8203459876	0.000193619172191984	1	41000	2

The Output window shows the results of the previous SQL statements:

#	Time	Action	Message	Duration / Fetch
84	22:29:15	INSERT INTO locations VALUES (1, "1265 King Street East", "L1H 1J2", "Oshawa", "ON", "C...")...	3 row(s) affected Records: 3 Duplicates: 0 Warnings: 0	0.016 sec
85	22:29:15	INSERT INTO departments VALUES (1, "Chemistry", 1), (2, "Data Analytics", 2), (3, "Biology", 3)	3 row(s) affected Records: 3 Duplicates: 0 Warnings: 0	0.000 sec
86	22:29:15	INSERT INTO jobs VALUES (1, "Associate", 40000, 70000), (2, "Data Scientist", 65000, 110000)...	3 row(s) affected Records: 3 Duplicates: 0 Warnings: 0	0.000 sec
87	22:29:15	INSERT INTO employees VALUES (1, "Noopa", "Gandhi", "noopagandhi@gmail.com", 9052780124, 9/7/2005, 2, 105000, NULL)...	4 row(s) affected Records: 4 Duplicates: 0 Warnings: 0	0.000 sec
88	22:29:15	INSERT INTO dependents VALUES (1, "Vicky", "Onyx", "Child", 3), (2, "Jeremie", "Gitsim", "Brother", 4), (3, "Pratish", "Pushkar", "Spouse", 1);	3 row(s) affected Records: 3 Duplicates: 0 Warnings: 0	0.016 sec
89	22:43:12	SELECT * FROM employees LIMIT 0, 1000	4 row(s) returned	0.000 sec / 0.000 sec

Screenshot 11 (Select Statement: Select all columns from dependents table)



MANAGEMENT

- Server Status
- Client Connections
- Users and Privileges
- Status and System Variables
- Data Export
- Data Import/Restore

INSTANCE

- Startup / Shutdown
- Server Logs
- Options File

PERFORMANCE

- Dashboard
- Performance Reports
- Performance Schema Setup

Administration Schemas

Information

No object selected

Object Info Session

SQL Editor:

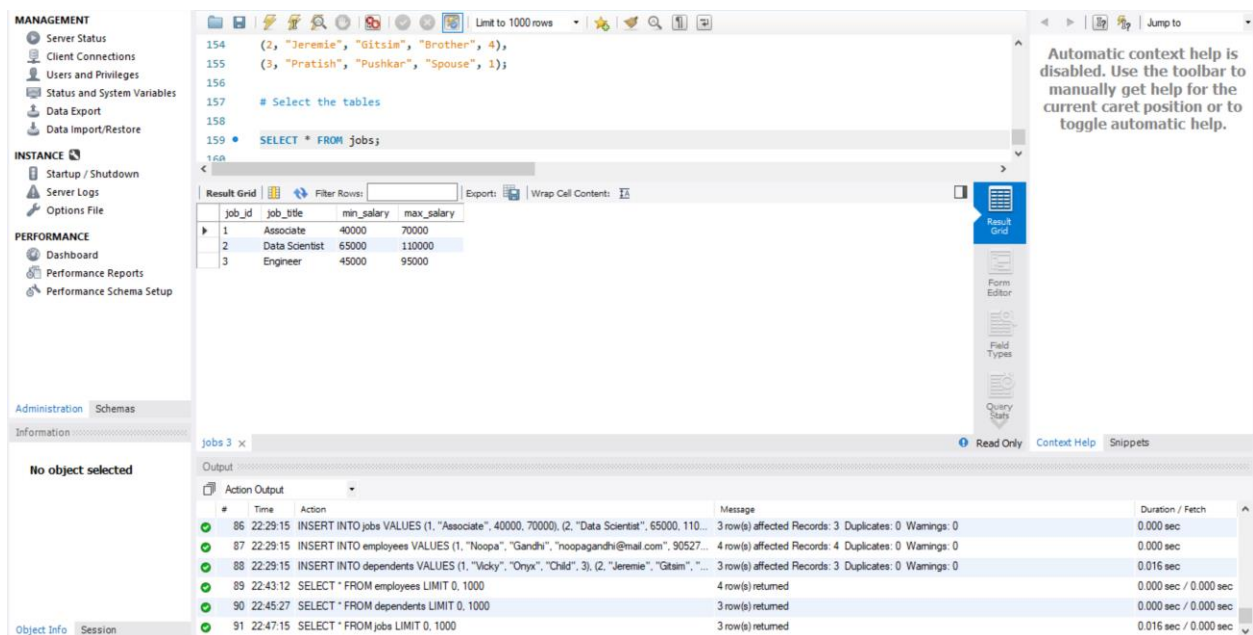
```
154 (2, "Jeremie", "Gitsim", "Brother", 4),
155 (3, "Pratish", "Pushkar", "Spouse", 1);
156
157 # Select the tables
158
159 SELECT * FROM dependents;
```

Results:

dependent_id	first_name	last_name	relationship	employee_id
1	Vicky	Onyx	Child	3
2	Jeremie	Gitsim	Brother	4
3	Pratish	Pushkar	Spouse	1

Automatic context help is disabled. Use the toolbar to manually get help for the current caret position or to toggle automatic help.

Screenshot 12 (Select Statement: Select all columns from jobs table)



MANAGEMENT

- Server Status
- Client Connections
- Users and Privileges
- Status and System Variables
- Data Export
- Data Import/Restore

INSTANCE

- Startup / Shutdown
- Server Logs
- Options File

PERFORMANCE

- Dashboard
- Performance Reports
- Performance Schema Setup

Administration Schemas

Information

No object selected

Object Info Session

SQL Editor:

```
154 (2, "Jeremie", "Gitsim", "Brother", 4),
155 (3, "Pratish", "Pushkar", "Spouse", 1);
156
157 # Select the tables
158
159 SELECT * FROM jobs;
```

Results:

job_id	job_title	min_salary	max_salary
1	Associate	40000	70000
2	Data Scientist	65000	110000
3	Engineer	45000	95000

Automatic context help is disabled. Use the toolbar to manually get help for the current caret position or to toggle automatic help.

Screenshot 13 (Select Statement: Select all columns from departments table)

Automatic context help is disabled. Use the toolbar to manually get help for the current caret position or to toggle automatic help.

departments 4 x

Output

#	Time	Action	Message	Duration / Fetch
87	22:29:15	INSERT INTO employees VALUES (1, "Noopa", "Gandhi", "noopagandhi@mail.com", 90527...	4 row(s) affected Records: 4 Duplicates: 0 Warnings: 0	0.000 sec
88	22:29:15	INSERT INTO dependents VALUES (1, "Vicky", "Onyx", "Child", 3), (2, "Jeremie", "Gitam", "...)	3 row(s) affected Records: 3 Duplicates: 0 Warnings: 0	0.016 sec
89	22:43:12	SELECT * FROM employees LIMIT 0, 1000	4 row(s) returned	0.000 sec / 0.000 sec
90	22:45:27	SELECT * FROM dependents LIMIT 0, 1000	3 row(s) returned	0.000 sec / 0.000 sec
91	22:47:15	SELECT * FROM jobs LIMIT 0, 1000	3 row(s) returned	0.016 sec / 0.000 sec
92	22:48:20	SELECT * FROM departments LIMIT 0, 1000	3 row(s) returned	0.000 sec / 0.000 sec

Screenshot 14 (Select Statement: Select all columns from locations table)

Automatic context help is disabled. Use the toolbar to manually get help for the current caret position or to toggle automatic help.

locations 5 x

Output

#	Time	Action	Message	Duration / Fetch
88	22:29:15	INSERT INTO dependents VALUES (1, "Vicky", "Onyx", "Child", 3), (2, "Jeremie", "Gitam", "...)	3 row(s) affected Records: 3 Duplicates: 0 Warnings: 0	0.016 sec
89	22:43:12	SELECT * FROM employees LIMIT 0, 1000	4 row(s) returned	0.000 sec / 0.000 sec
90	22:45:27	SELECT * FROM dependents LIMIT 0, 1000	3 row(s) returned	0.000 sec / 0.000 sec
91	22:47:15	SELECT * FROM jobs LIMIT 0, 1000	3 row(s) returned	0.016 sec / 0.000 sec
92	22:48:20	SELECT * FROM departments LIMIT 0, 1000	3 row(s) returned	0.000 sec / 0.000 sec
93	22:49:50	SELECT * FROM locations LIMIT 0, 1000	3 row(s) returned	0.000 sec / 0.000 sec

Screenshot 15 (Select Statement: Select all columns from countries table)

Automatic context help is disabled. Use the toolbar to manually get help for the current caret position or to toggle automatic help.

```

154 (2, "Jeremie", "Gitsin", "Brother", 4),
155 (3, "Pratish", "Pushkar", "Spouse", 1);
156
157 # Select the tables
158
159 * SELECT * FROM countries;

```

country_id	country	region_id
CAN	Canada	1
IND	India	3
NGN	Nigeria	4

countries 6 x

Output

#	Time	Action	Message	Duration / Fetch
89	22:43:12	SELECT * FROM employees LIMIT 0, 1000	4 row(s) returned	0.000 sec / 0.000 sec
90	22:45:27	SELECT * FROM dependents LIMIT 0, 1000	3 row(s) returned	0.000 sec / 0.000 sec
91	22:47:15	SELECT * FROM jobs LIMIT 0, 1000	3 row(s) returned	0.016 sec / 0.000 sec
92	22:48:20	SELECT * FROM departments LIMIT 0, 1000	3 row(s) returned	0.000 sec / 0.000 sec
93	22:49:50	SELECT * FROM locations LIMIT 0, 1000	3 row(s) returned	0.000 sec / 0.000 sec
94	22:51:14	SELECT * FROM countries LIMIT 0, 1000	3 row(s) returned	0.000 sec / 0.000 sec

Screenshot 16 (Select Statement: Select all columns from regions table)

Automatic context help is disabled. Use the toolbar to manually get help for the current caret position or to toggle automatic help.

```

154 (2, "Jeremie", "Gitsin", "Brother", 4),
155 (3, "Pratish", "Pushkar", "Spouse", 1);
156
157 # Select the tables
158
159 * SELECT * FROM regions;

```

region_id	region_name
1	North America
2	Europe
3	Asia
4	Africa

regions 7 x

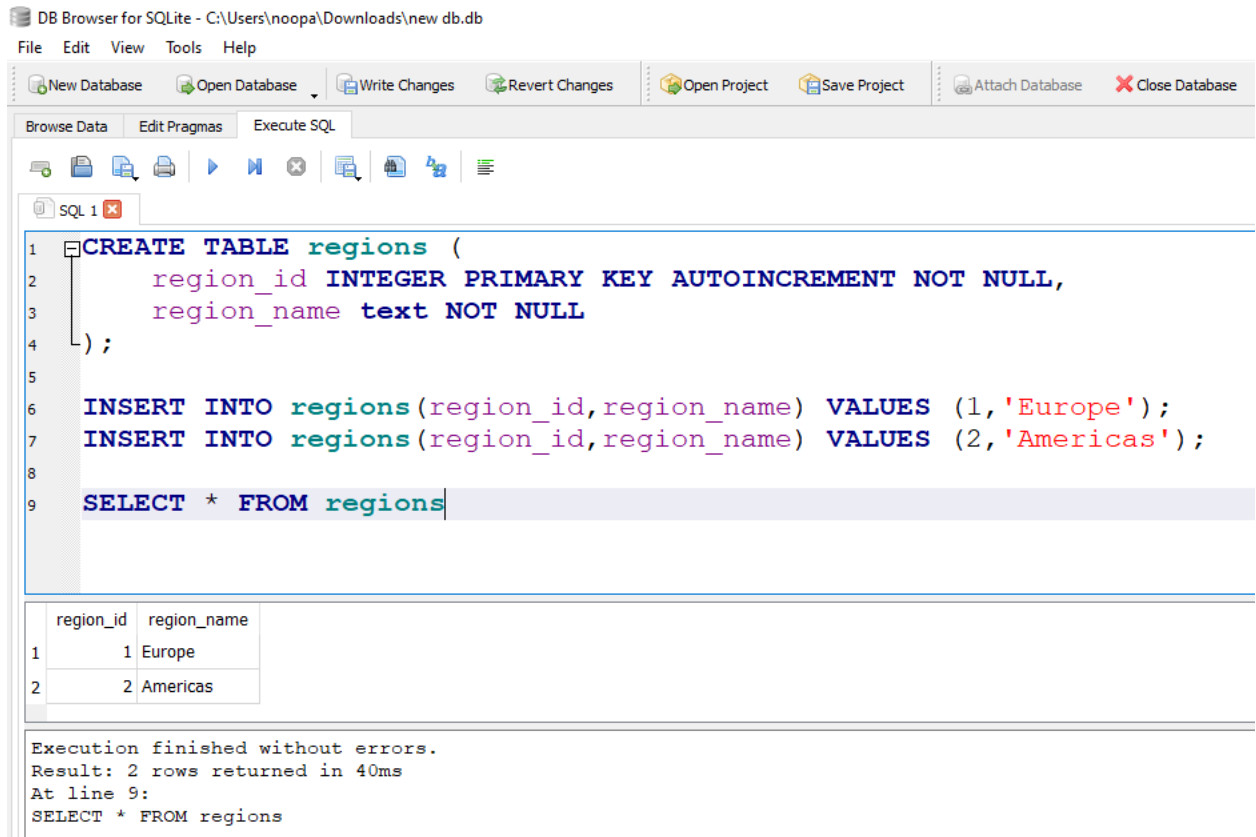
Output

#	Time	Action	Message	Duration / Fetch
90	22:45:27	SELECT * FROM dependents LIMIT 0, 1000	3 row(s) returned	0.000 sec / 0.000 sec
91	22:47:15	SELECT * FROM jobs LIMIT 0, 1000	3 row(s) returned	0.016 sec / 0.000 sec
92	22:48:20	SELECT * FROM departments LIMIT 0, 1000	3 row(s) returned	0.000 sec / 0.000 sec
93	22:49:50	SELECT * FROM locations LIMIT 0, 1000	3 row(s) returned	0.000 sec / 0.000 sec
94	22:51:14	SELECT * FROM countries LIMIT 0, 1000	3 row(s) returned	0.000 sec / 0.000 sec
95	22:52:43	SELECT * FROM regions LIMIT 0, 1000	4 row(s) returned	0.000 sec / 0.000 sec

Submission Format:

A single word document including Part-A and Part-B. For **Part A** students can draw the ER diagram on a paper with pen or use the tool <https://app.diagrams.net/>

For **Part-B** the screen shots of all the constructed table have to be provided as given below. Every table screenshot has to be provided as below



The screenshot shows the DB Browser for SQLite interface. The title bar indicates the file path: C:\Users\nooa\Downloads\new db.db. The menu bar includes File, Edit, View, Tools, and Help. The toolbar contains buttons for New Database, Open Database, Write Changes, Revert Changes, Open Project, Save Project, Attach Database, and Close Database. The main window has tabs for Browse Data, Edit Pragmas, and Execute SQL. The Execute SQL tab is active, showing a SQL script in a text area. Below the text area, a table displays the results of the SELECT statement. At the bottom, a status bar shows the execution message: 'Execution finished without errors. Result: 2 rows returned in 40ms. At line 9: SELECT * FROM regions'.

```
1 CREATE TABLE regions (  
2     region_id INTEGER PRIMARY KEY AUTOINCREMENT NOT NULL,  
3     region_name text NOT NULL  
4 );  
5  
6 INSERT INTO regions(region_id,region_name) VALUES (1, 'Europe');  
7 INSERT INTO regions(region_id,region_name) VALUES (2, 'Americas');  
8  
9 SELECT * FROM regions
```

region_id	region_name
1	Europe
2	Americas

Execution finished without errors.
Result: 2 rows returned in 40ms
At line 9:
SELECT * FROM regions

The database created ie the .db file for Part-B should also be submitted.

Note:

Any kind of violation of academic integrity including copying other students work and copying answers directly from Internet will automatically lead to the assignment to be graded as zero.

The facility for the submission of late assignments is up to a maximum of 72 hours after the assignment due date. All allowed late submissions will be assessed a penalty of 25% of the total possible grade for the assignment.