



**Department of Electrical  
& Computer Engineering**  
Faculty of Engineering & Architectural Science

<b>Course Title</b>	Database Systems 1
<b>Course Number</b>	CPS 510
<b>Semester/Year</b>	F2020
<b>Instructor</b>	Dr. Abdolreza Abhari

<b>Assignment Number</b>	10
<b>Assignment Title</b>	Final Project - Database Management System

<b>Submission Date</b>	Dec 01, 2020
<b>Due Date</b>	Dec 01, 2020

<b>Full Name</b>	<b>Student Number</b>	<b>Section Number</b>	<b>Initial</b>
Edmond Chu	500894595	07	E.C.
Brian Lee	500900778		B.L
Ashwin Thomas	500877120		A.T.

## **Table of Contents**

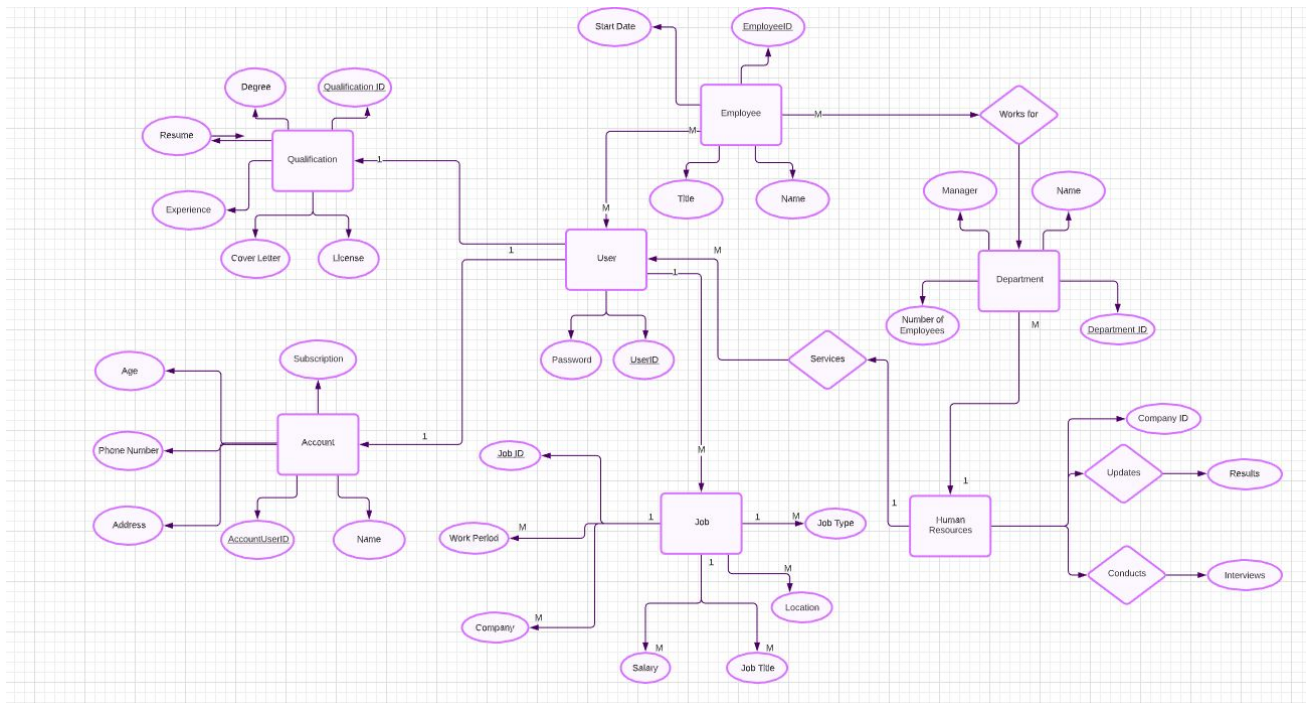
Introduction	2
Entity-Relationship Diagram	3
UNIX Shell Implementation	4
Functional Dependencies and Database Normalization	5
Simple and Advanced Database Queries (with Relational Algebra)	23
SQL Code	26
Java GUI Code	32
Conclusion	71

## **Introduction:**

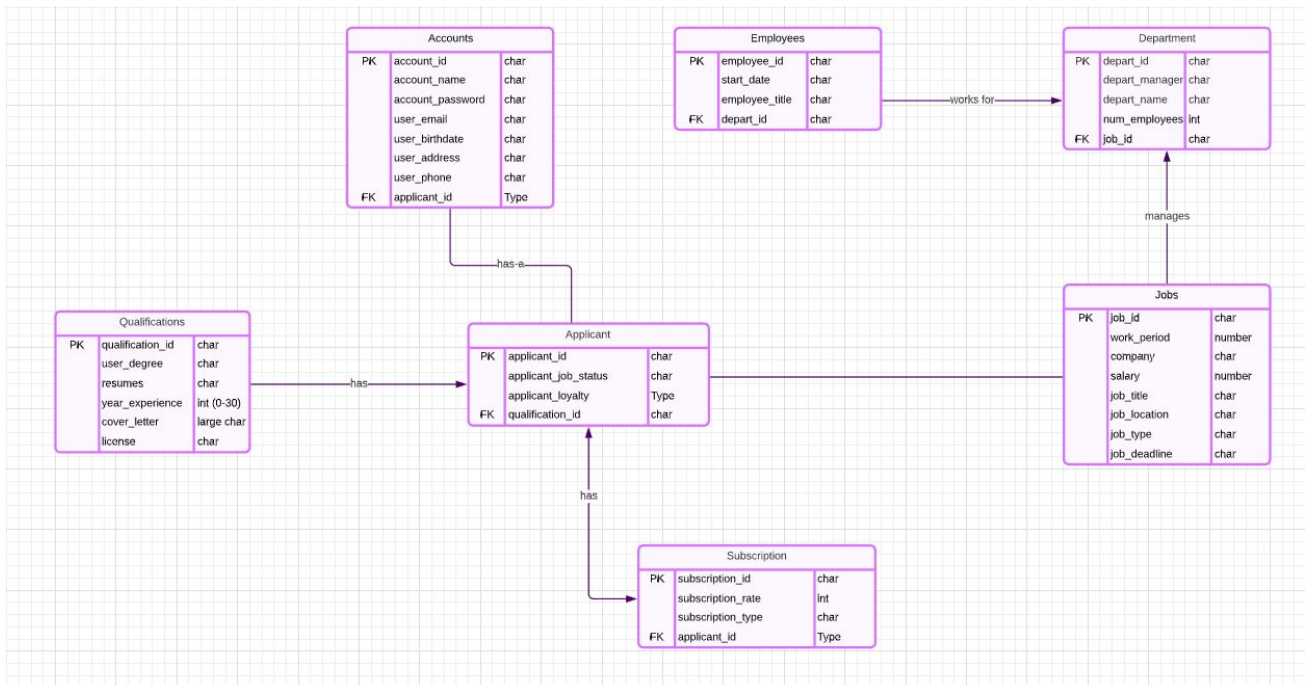
For this project, our objective was to create a platform where users gain access to a job search database. In this technology centric world and competitive state of today's job market, online job banks such as LinkedIn and Indeed have been soaring in popularity and are crucial as they connect people all around the world. They provide users with various job opportunities that can be accessed with just a click of a button. Taking inspiration from these established services, we try to incorporate networking components from LinkedIn and job filtering components from Indeed. This would facilitate interactions between potential job candidates and hiring managers and provide an easy-to-use interface where users can easily identify and apply for jobs that tailor to their needs.

In our DBMS, users can also opt for a subscription based premium service in which they can get further detailed information about jobs and a direct connection to interviewers. Large and small companies also reap the benefits of these subscription services as they could pool these applicants in a different bracket and hire them based on company needs. This application will have the ability to hold specific types of information about jobs making it easier for the user. Users would be able to search and sort for jobs based on their professions, salaries, companies, job types, locations, and many more. With the subscription, they are provided with a more premium service including the ability to communicate with industry professionals, precisely track profile interest levels, along with a plethora of features which would be available based on their subscription type.

## Entity-Relationship Diagram:



Initial diagram used a starting point for DBMS Code



Finalized diagram after BCNF Normalization

## Basic UNIX Shell Implementation:

```
| Oracle All Inclusive Tool
| Main Menu - Select Desired Operation(s):
| <CTRL-Z Anytime to Enter Interactive CMD Prompt>
|-----|
M) View Manual
1) Drop Tables
2) Create Tables
3) Populate Tables
4) Query Tables

X) Force/Stop/Kill Oracle DB

E) End/Exit
Choose:
1

SQL*Plus: Release 12.1.0.2.0 Production on Tue Oct 27 03:16:28 2020
Copyright (c) 1982, 2014, Oracle. All rights reserved.

Connected to:
Oracle Database 11g Enterprise Edition Release 11.2.0.1.0 - 64bit Production
With the Partitioning, OLAP, Data Mining and Real Application Testing options

SQL>
Table dropped.

SQL>
Table dropped.

SQL>
Table dropped.

SQL>
Table dropped.

SQL>
Table dropped.

SQL>
Table dropped.

SQL>
Table dropped.
```

### Dropping tables

```
| Oracle All Inclusive Tool
| Main Menu - Select Desired Operation(s):
| <CTRL-Z Anytime to Enter Interactive CMD Prompt>
|-----|
M) View Manual
1) Drop Tables
2) Create Tables
3) Populate Tables
4) Query Tables

X) Force/Stop/Kill Oracle DB

E) End/Exit
Choose:
2

SQL*Plus: Release 12.1.0.2.0 Production on Tue Oct 27 03:17:09 2020
Copyright (c) 1982, 2014, Oracle. All rights reserved.

Connected to:
Oracle Database 11g Enterprise Edition Release 11.2.0.1.0 - 64bit Production
With the Partitioning, OLAP, Data Mining and Real Application Testing options

SQL> 2 3 4 5 6 7
Table created.

SQL> 2 3 4 5
Table created.

SQL> 2 3 4 5 6 7 8 9 10 11
Table created.

SQL> 2 3 4 5 6 7 8 9 10
Table created.

SQL> 2 3 4 5 6 7 8 9 10
Table created.

SQL> 2 3 4 5 6 7
Table created.

SQL> 2 3 4 5 6 7 8
Table created.
```

### Creating tables

```
SQL> SQL> 2
1 row created.

SQL> SQL> 2
1 row created.

SQL> SQL> 2
1 row created.

SQL> SQL> 2
1 row created.

SQL> SQL> 2
1 row created.

SQL> SQL> 2
1 row created.

SQL> SQL> 2
1 row created.

SQL> SQL> 2
1 row created.

SQL> SQL> 2
1 row created.

SQL> SQL> 2
1 row created.

SQL> SQL> 2
1 row created.

SQL> SQL> 2
1 row created.

SQL> SQL> 2
1 row created.

SQL> SQL> 2
1 row created.

SQL> SQL> 2
1 row created.

SQL> SQL> 2
1 row created.

SQL> SQL> 2
1 row created.
```

### Table population

```
SQL>
ACCOUNT_USERID      SUBSCRIPTION      AGE
-----
ADDRESS             PHONE ACCOUNT_NAME
-----
ID#2                Free              45
18 Houseview       6478724302 User2

ID#3                Student           20
6 FreeMarket       6474520201 User3

ID#1                Premium           16
7 Eglinton         6478720202 User1

SQL>
COMPANY             JOB_LOCATION
-----
Samsung             Cupertino
Facebook            Toronto
Apple               San Jose

SQL>
WORKER_TYPE
-----
Apprentice
Full-time
Full-time

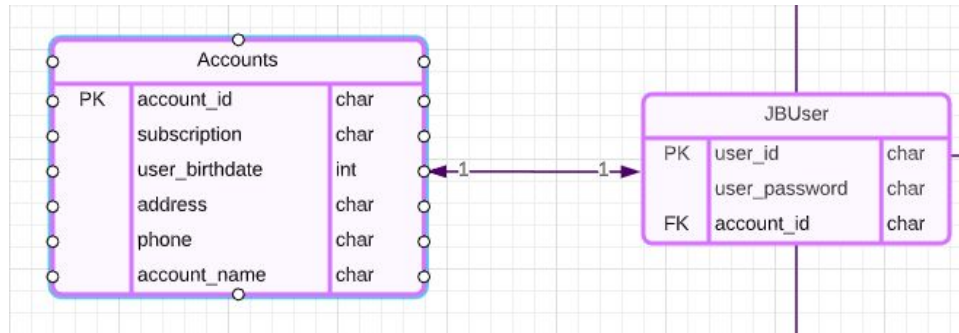
SQL>
HR_ID
-----
INTERVIEW
-----
HR#1
Complete

HR#3
Complete
```

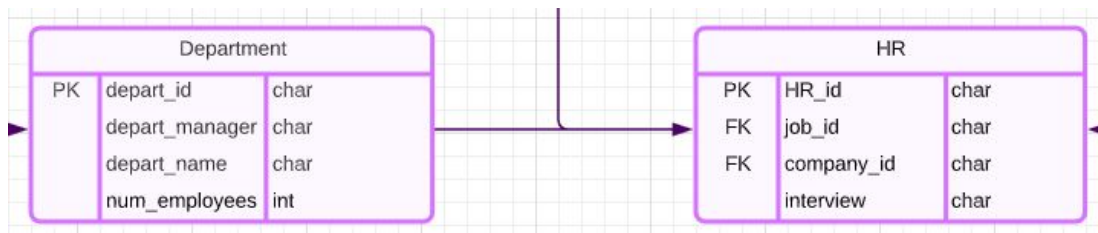
### Sample Queries

## Functional Dependencies and Database Normalization:

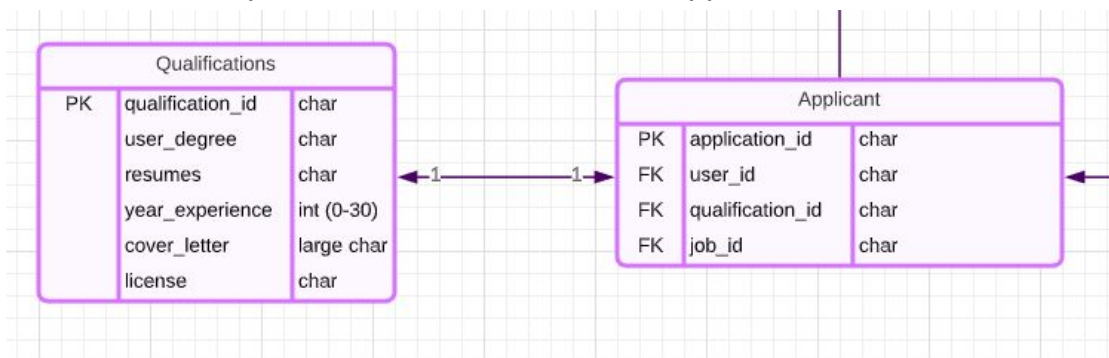
After the completion of our design for our DBMS system, we arrived at the step of normalizing our design. In order to normalize our design, we first outlined the functional dependencies in the system. For example, our one-to-one relationship between JBUUser and Accounts.



A user can have one account, and each account may only have one user. Therefore, the functional dependency that would express this relationship would be,  $\text{account\_id} \rightarrow \text{user\_id}$  and  $\text{user\_id} \rightarrow \text{account\_id}$ . Another example is our one-to-one relationship between Department and HR.

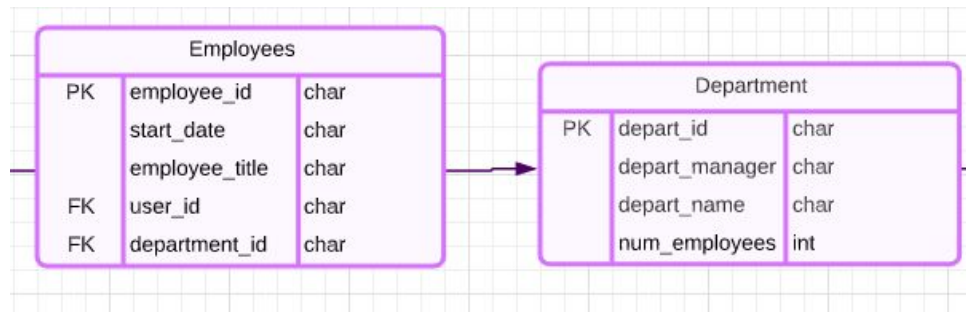


A Department can have one HR, and each HR may only have one Department. Therefore, the functional dependency that would express this relationship would be,  $\text{depart\_id} \rightarrow \text{HR\_id}$  and  $\text{HR\_id} \rightarrow \text{depart\_id}$ . One more Example would be our one-to-one relationship between Qualifications and Applicant.

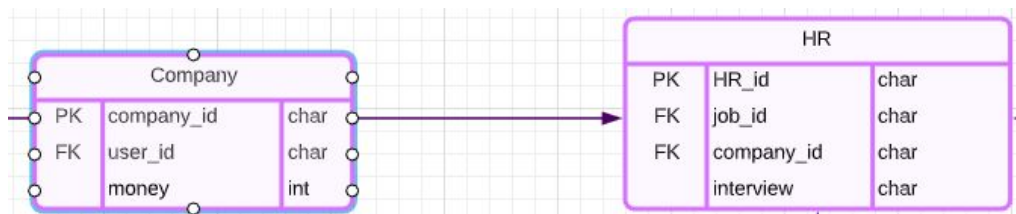


An Applicant can have one set of qualifications, and each set of qualifications may only have one Applicant. Therefore, the functional dependency that expresses this relationship would be,  $\text{application\_id} \rightarrow \text{qualification\_id}$  and  $\text{qualification\_id} \rightarrow \text{application\_id}$

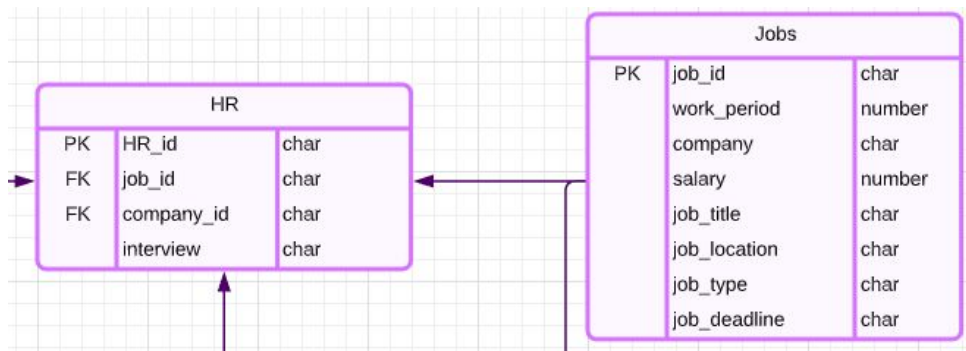
→application\_id. We can also look at an example between the one-to-many relationship between our Department and Employee entity.



This functional dependency shows how the department can have several employees, but an employee can only have one department. Due to this fact, the functional dependency that expresses this relationship would be,  $employee\_id \rightarrow depart\_id$ . More examples of the functional dependency of one-to-many relationship are shown below.

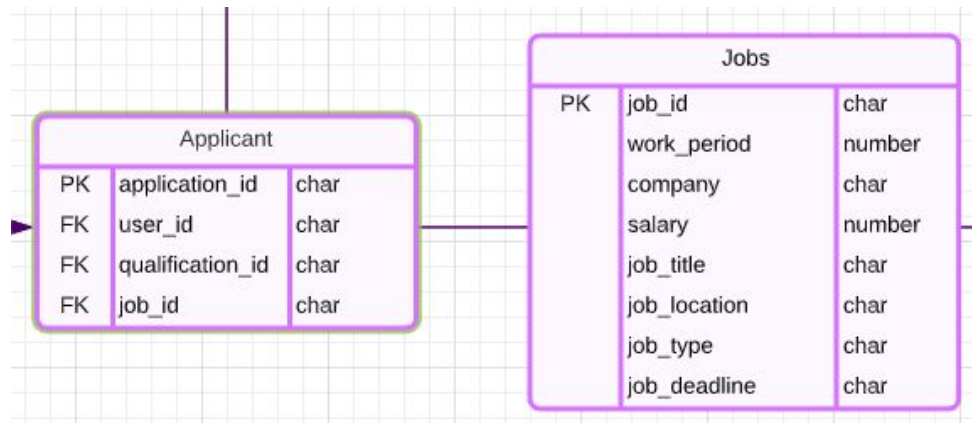


This functional dependency shows how HR can have service to several Companies, but a Company can only communicate with one HR department. Due to this fact, the functional dependency that expresses this relationship would be,  $company\_id \rightarrow HR\_id$ .

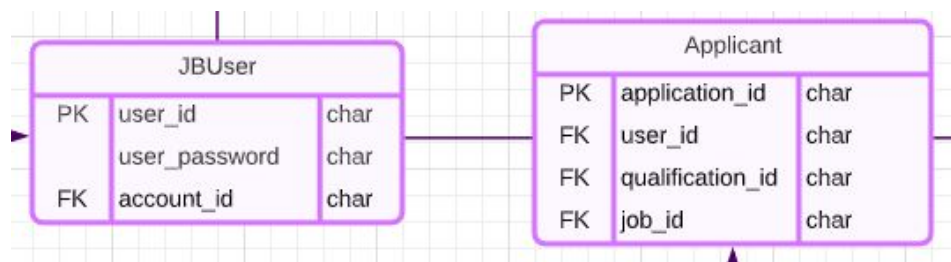


This functional dependency shows how the HR can manage several jobs, but a job can only be managed by one HR. Due to this fact, the functional dependency that expresses this relationship would be,  $job\_id \rightarrow HR\_id$ . In the case that there is a many-to-many relationship, there is no functional dependency. For example, the relationship between Applicant and Jobs.

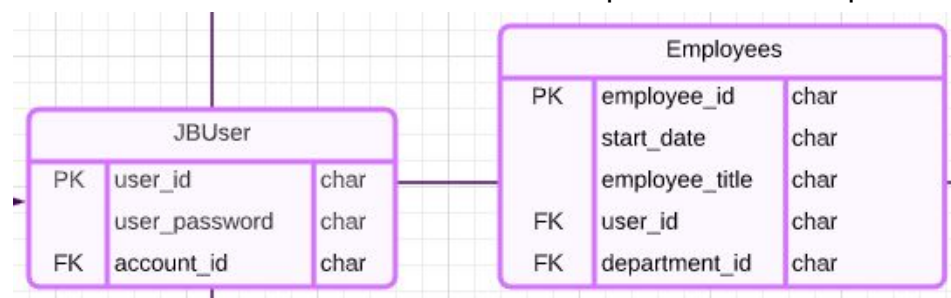




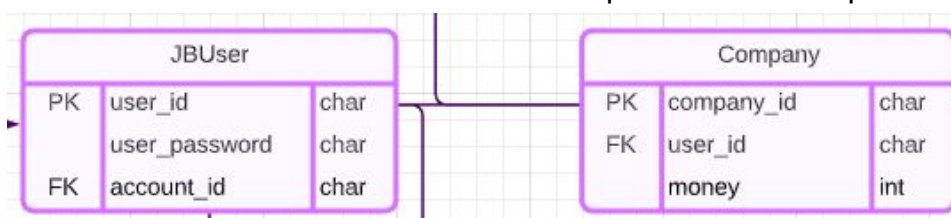
As we can see, many applicants can apply to one job, however one applicant can apply to many jobs. Therefore we can see that there are no functional dependencies to capture here.



As we can see, many users can be an applicant, however all applicants are a user. Therefore we can see that there are no functional dependencies to capture here.



As we can see, many users can be an employee, however all employees are a user. Therefore we can see that there are no functional dependencies to capture here.

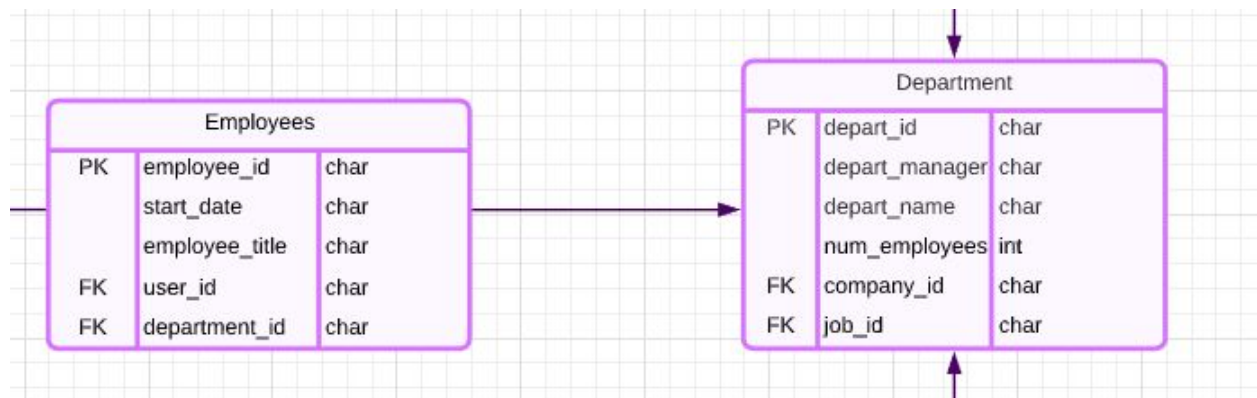


As we can see, many users can be a company, however all companies are users. Therefore we can see that there are no functional dependencies to capture here. By identifying all the functional dependencies, we were able to optimize our DBMS system by reducing the redundancies in our system.



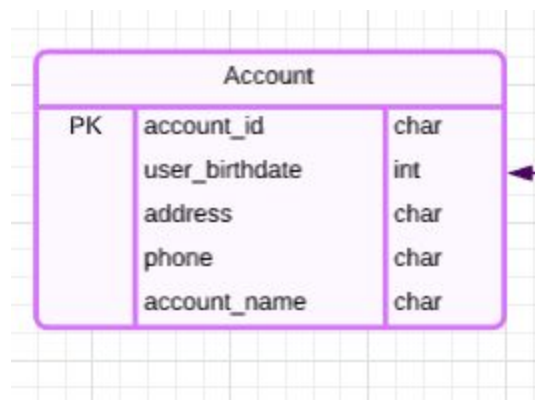
### 3NF between Department and Employee

- Employees are part of a specific department. This employee has a manager, but the manager attribute is reliant on the department id which creates a transitive dependency. The department that the employee is part of also has a name, however the department name is reliant on the department id creating a transitive dependency. Therefore, following the 3rd Normal Form rules, we move department name and department manager into the department table, and instead reference these columns using department id.



### Account Table

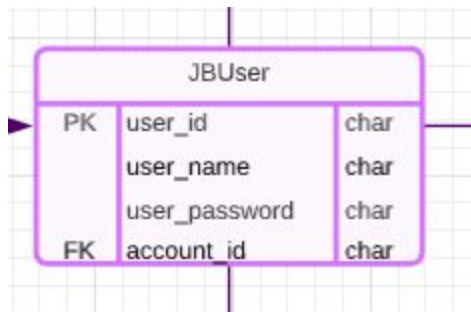
- The attributes in the account table are all dependent on the account id and are not dependent on any other attributes from other tables. By observation, the user birthdate, address, phone, account name are all unique attributes that are dependent on the account id.



### JBUser Table

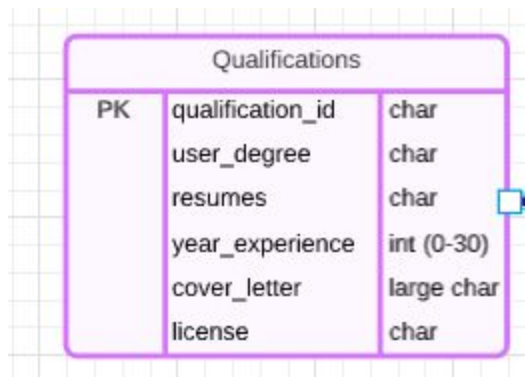
- The attributes in the JBUser table are all dependent on the user id and are not dependent on any other attributes from other tables. By observation, the

username and password are all unique attributes that are dependent on the user id.



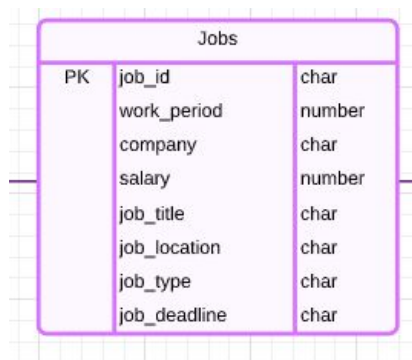
## Qualifications

- The attributes in the qualifications table are all dependent on the qualification id and are not dependent on any other attributes from other tables. By observation, the degree, resume, years of experience, cover letter, license are all unique attributes that are dependent on the qualification id.



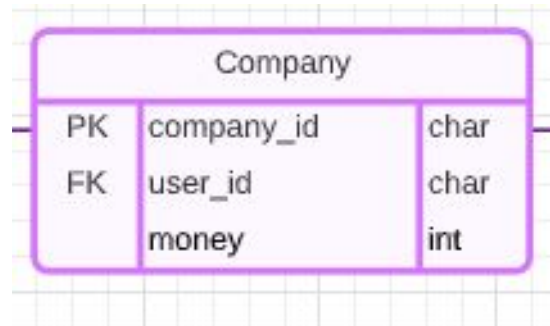
## Jobs

- The attributes in the Jobs table are all dependent on the Job id and are not dependent on any other attributes from other tables. By observation, the job work period, company, salary, job title, job location, job type, job deadline are all unique attributes that are dependent on the job id.



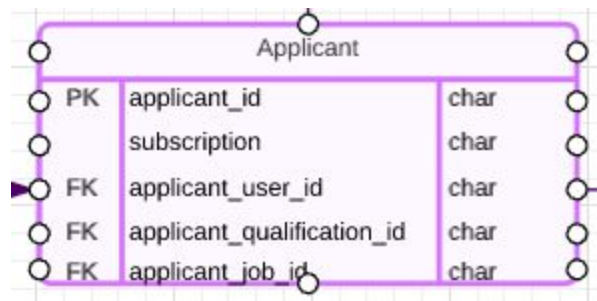
## Company

- The attributes in the Company table are all dependent on the company id and are not dependent on any other attributes from other tables. By observation, the company money is a unique attribute that is dependent on the company id.



### Applicant

- The attributes in the Applicant table are all dependent on the applicant id and are not dependent on any other attributes from other tables. By observation, the applicant subscription unique attributes that are dependent on the applicant id. The applicant subscription was not included in accounts because accounts can be used by companies and employees that do not require a subscription.



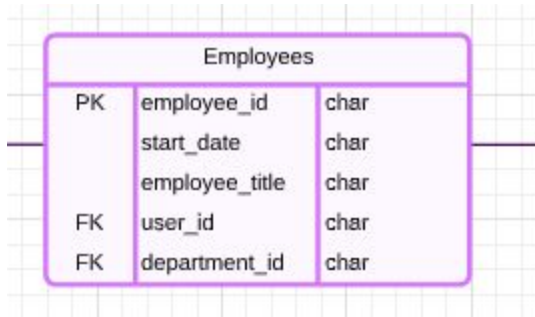
### Final Step: Conversion from 3NF to BCNF

#### Bernstein's Algorithm - Broken down into 4 steps:

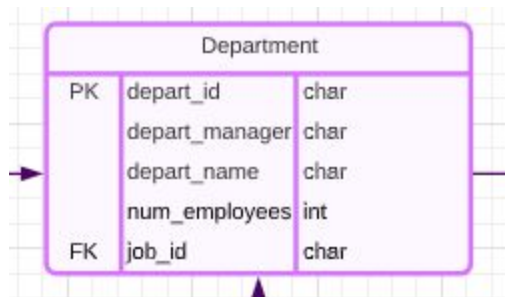
- 1) Determine all the functional dependencies
- 2) a) Find and remove redundancies  
b) Find and remove partial dependencies
- 3) Find keys
- 4) Create tables

#### Step 1 (finding all functional dependencies)

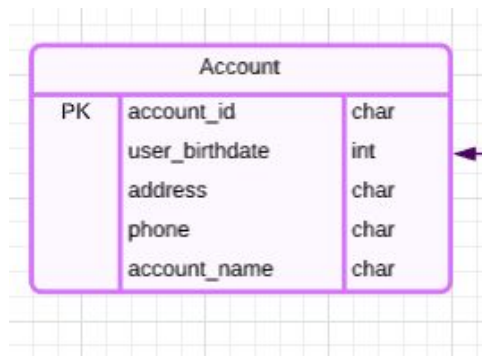
- **employee\_id** -> {start\_date, employee\_title}



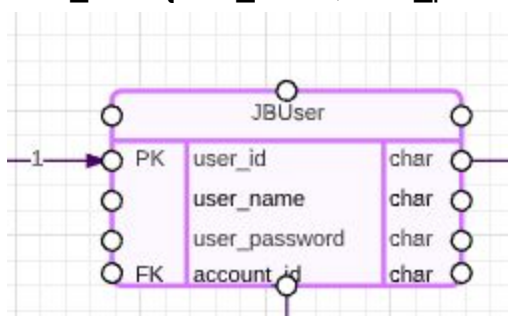
- **depart\_id** -> {depart\_manager, depart\_name, num\_employees}



- **account\_id** -> {user\_birthdate, address, phone, account\_name}



- **user\_id** -> {user\_name, user\_password}



- **qualification\_id** -> {user\_degree, resume, years\_experience, cover\_letter, license}

Qualifications		
PK	qualification_id	char
	user_degree	char
	resumes	char
	year_experience	int (0-30)
	cover_letter	large char
	license	char

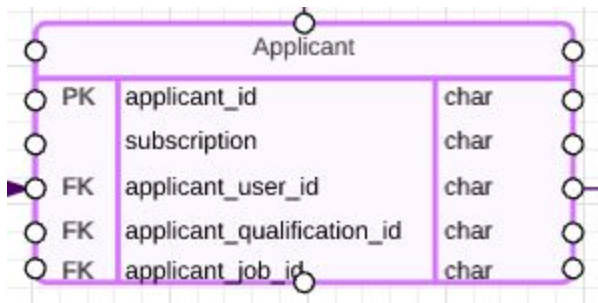
- **job\_id** -> {work\_period, company, salary, job\_title, job\_location, job\_type, job\_deadline}

Jobs		
PK	job_id	char
	work_period	number
	company	char
	salary	number
	job_title	char
	job_location	char
	job_type	char
	job_deadline	char
FK	company_id	char

- **company\_id** -> {money}

Company		
PK	company_id	char
FK	user_id	char
	money	int

- **applicant\_id** -> {subscription}



## Step 2a (Break RHS and find redundancies)

### Redundancies

- **employee\_id** -> {start\_date, employee\_title, user\_id}
  - Reduced list of FD's:
  - **employee\_id** -> {start\_date}
  - **employee\_id** -> {employee\_title}
  - **no redundancies**
  
- **depart\_id** -> {depart\_manager, depart\_name, num\_employees}
  - Reduced list of FD's:
  - **depart\_id** -> {depart\_manager}
  - **depart\_id** -> {depart\_name}
  - **depart\_id** -> {num\_employees}
  - **no redundancies**
  
- **account\_id** -> {user\_birthdate, address, phone, account\_name}
  - Reduced list of FD's:
  - **Account\_id** -> {user\_birthdate}
  - **Account\_id** -> {address, phone}
  - **Account\_id** -> {phone}
  - **Account\_id** -> {account\_name}
  - **no redundancies**
  
- **user\_id** -> {user\_name, user\_password}
  - Reduced list of FD's:
  - **user\_id** -> {user\_name}
  - **user\_id** -> {user\_password}
  - **no redundancies**
  
- **qualification\_id** -> {user\_degree, resume, years\_experience, cover\_letter, license}



- Reduced list of FD's:
- **qualification\_id** -> {user\_degree}
- **qualification\_id** -> {resume}
- **qualification\_id** -> {years\_experience}
- **qualification\_id** -> {cover\_letter}
- **qualification\_id** -> {license}
- **no redundancies**
  
- **job\_id** -> {work\_period, company, salary, job\_title, job\_location, job\_type, job\_deadline}
  - Reduced list of FD's:
  - **job\_id** -> {work\_period}
  - **job\_id** -> {company}
  - **job\_id** -> {salary}
  - **job\_id** -> {job\_title}
  - **job\_id** -> {job\_location}
  - **job\_id** -> {job\_type}
  - **job\_id** -> {job\_deadline}
  - **no redundancies**
  
- **company\_id** -> {money}
  - Reduced list of FD's:
  - **company\_id** -> {money}
  - **no redundancies**
  
- **applicant\_id** -> {subscription}
  - Reduced list of FD's:
  - **applicant\_id** -> {subscription}
  - **no redundancies**

### Step 2b (Minimize LHS, find and remove partial dependencies)

- LHS is already minimized, therefore there are no partial dependencies

### Step 3(Find keys) (relational schema)

- **employee\_id** -> {start\_date, employee\_title}
  - Attributes on RHS but not on LHS (cannot be keys)
    - start\_date
    - employee\_title
  - Possible Keys

- employee\_id
- **depart\_id** -> {depart\_manager, depart\_name, num\_employees}
  - Attributes on RHS but not on LHS (cannot be keys)
    - depart\_manager
    - depart\_name
    - num\_employees
  - Possible Keys
    - depart\_id
- **account\_id** -> {user\_birthdate, address, phone, account\_name}
  - Attributes on RHS but not on LHS (cannot be keys)
    - user\_birthdate
    - address
    - phone
    - account\_name
  - Possible Keys
    - account\_id
- **user\_id** -> {user\_name, user\_password}
  - Attributes on RHS but not on LHS (cannot be keys)
    - user\_name
    - user\_password
  - Possible Keys
    - user\_id
- **qualification\_id** -> {user\_degree, resume, years\_experience, cover\_letter, license}
  - Attributes on RHS but not on LHS (cannot be keys)
    - user\_degree
    - resume
    - years\_experience
    - cover\_letter
    - license
  - Possible Keys
    - qualification\_id
- **job\_id** -> {work\_period, company, salary, job\_title, job\_location, job\_type, job\_deadline}
  - Attributes on RHS but not on LHS (cannot be keys)
    - work\_period
    - company
    - salary
    - job\_title
    - job\_location

- job\_type
- job\_deadline
- Possible Keys
  - job\_id
- **company\_id** -> {money}
  - Attributes on RHS but not on LHS (cannot be keys)
    - money
  - Possible Keys
    - company\_id
- **applicant\_id** -> {subscription}
  - Attributes on RHS but not on LHS (cannot be keys)
    - subscription
  - Possible Keys
    - applicant\_id

#### Step 4(Make tables)

R1(employee\_id, start\_date, employee\_title) with FD: **employee\_id** -> {start\_date, employee\_title}

R2(depart\_id, depart\_manager, depart\_name, num\_employees) with FD: **depart\_id** -> {depart\_manager, depart\_name, num\_employees}

R3(account\_id, user\_birthdate, address, phone, account\_name) with FD: **account\_id** -> {user\_birthdate, address, phone, account\_name}

R4(user\_id, user\_name, user\_password) with FD: **user\_id** -> {user\_name, user\_password}

R5(qualification\_id, user\_degree, resume, years\_experience, cover\_letter, license) with FD: **qualification\_id** -> {user\_degree, resume, years\_experience, cover\_letter, license}

R6(job\_id, work\_period, company, salary, job\_title, job\_location, job\_type, job\_deadline) with FD: **job\_id** -> {work\_period, company, salary, job\_title, job\_location, job\_type, job\_deadline}

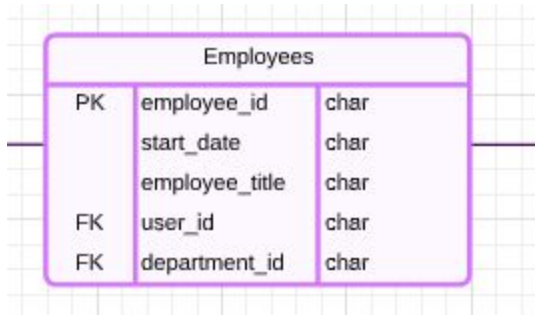
R7(company\_id, money) with FD: **company\_id** -> {money}

R8(application\_id, subscription) with FD: **applicant\_id** -> {subscription}

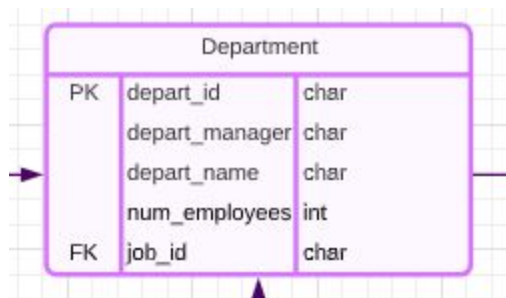
BCNF (Boyce Codd Normal Form)

### Step 1 (finding all functional dependencies)

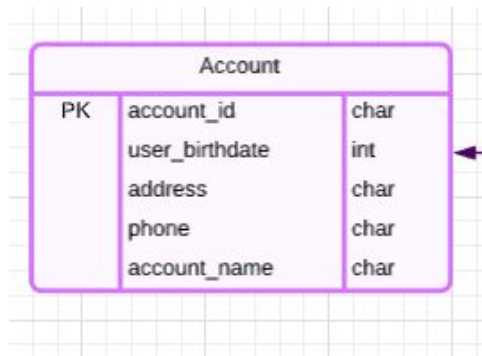
- **employee\_id** -> {start\_date, employee\_title, user\_id, department\_id}



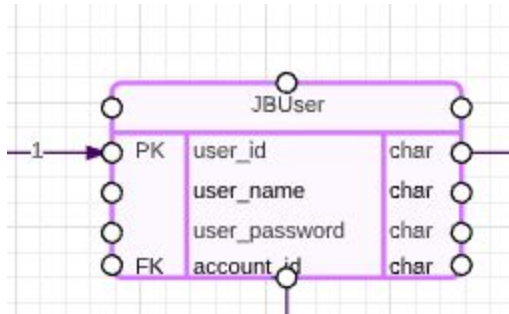
- **depart\_id** -> {depart\_manager, depart\_name, num\_employees, job\_id}



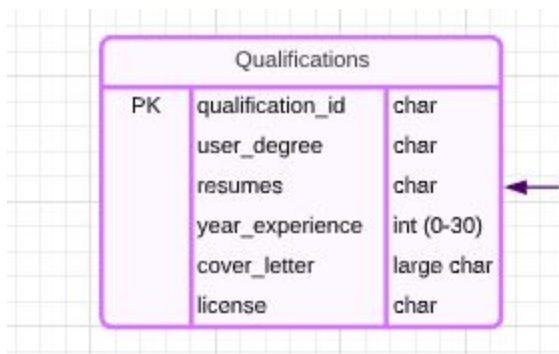
- **account\_id** -> {user\_birthdate, address, phone, account\_name}



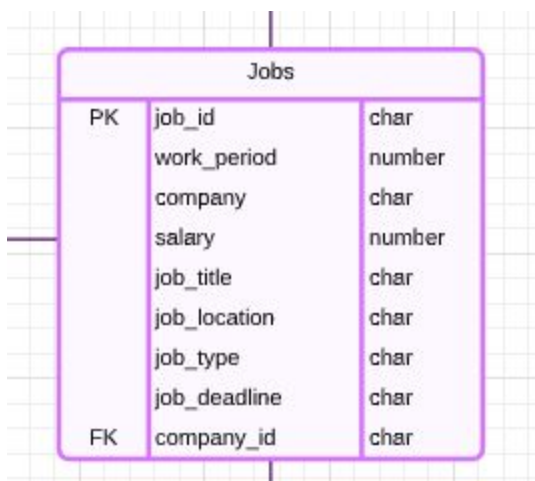
- **user\_id** -> {user\_name, user\_password, account\_id}



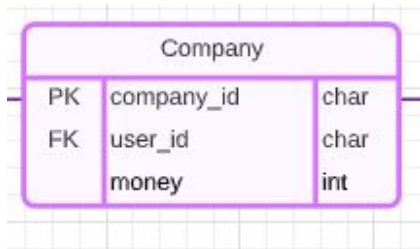
- **qualification\_id** -> {user\_degree, resume, years\_experience, cover\_letter, license}



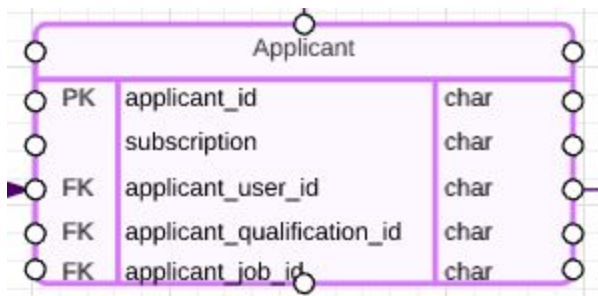
- **job\_id** -> {work\_period, company, salary, job\_title, job\_location, job\_type, job\_deadline, company\_id}



- **company\_id** -> {money, user\_id}



- **applicant\_id** -> {subscription, applicant\_user\_id, applicant\_qualifications\_id, applicant\_job\_id}



## Step 2 (make sure that the left hand side are keys if not decompose)

Consider the relation schema

- R1(employee\_id, start\_date, employee\_title)

with FD

- **employee\_id** -> {start\_date, employee\_title}

This schema has one candidate key

- employee\_id

Therefore this schema is in BCNF

Consider the relation schema

- R2(depart\_id, depart\_manager, depart\_name, num\_employees)

with FD

- **depart\_id** -> {depart\_manager, depart\_name, num\_employees}

This schema has one candidate key

- depart\_id

Therefore this schema is in BCNF

Consider the relation schema

- R3(account\_id, user\_birthdate, address, phone, account\_name)

with FD

- **account\_id** -> {user\_birthdate, address, phone, account\_name}

This schema has one candidate key

- account\_id



Therefore this schema is in BCNF

Consider the relation schema

- R4(user\_id, user\_name, user\_password)

with FD

- **user\_id** -> {user\_name, user\_password}

This schema has one candidate key

- user\_id

Therefore this schema is in BCNF

Consider the relation schema

- R5(qualification\_id, user\_degree, resume, years\_experience, cover\_letter, license)

with FD

- **qualification\_id** -> {user\_degree, resume, years\_experience, cover\_letter, license}

This schema has one candidate key

- qualification\_id

Therefore this schema is in BCNF

Consider the relation schema

- R6(job\_id, work\_period, company, salary, job\_title, job\_location, job\_type, job\_deadline)

with FD

- **job\_id** -> {work\_period, company, salary, job\_title, job\_location, job\_type, job\_deadline}

This schema has one candidate key

- job\_id

Therefore this schema is in BCNF

Consider the relation schema

- R7(company\_id, money)

with FD

- **company\_id** -> {money}

This schema has one candidate key

- company\_id

Therefore this schema is in BCNF

Consider the relation schema

- R8(application\_id, subscription)

with FD

- **applicant\_id** -> {subscription}

This schema has one candidate key

- **applicant\_id**

Therefore this schema is in BCNF

### Step 3 (final BCNF schema for R)

R1(employee\_id, start\_date, employee\_title)

R2(depart\_id, depart\_manager, depart\_name, num\_employees)

R3(account\_id, user\_birthdate, address, phone, account\_name)

R4(user\_id, user\_name, user\_password)

R5(qualification\_id, user\_degree, resume, years\_experience, cover\_letter, license)

R6(job\_id, work\_period, company, salary, job\_title, job\_location, job\_type, job\_deadline)

R7(company\_id, money)

R8(application\_id, subscription)

### Creating BCNF with Functional Dependencies for at least one table:

#### Relationship Used: Employees and Department

#### Combined Table (with FD):

emp_id	emp_nationality	emp_name	emp_department	emp_departNum
1000	Chinese	Edmond	Purchasing	200
1000	Chinese	Edmond	HR	15
1001	Korean	Brian	R&D	45
1002	Indian	Ashwin	Technical Support	100

#### Functional Dependencies:

1) emp\_id -> {emp\_nationality, emp\_name}

2) emp\_department -> emp\_departNum

#### Candidate key:

{emp\_id, emp\_department}

BCNF Conversion as Follows:

#### Table 1 (Employee Table):

emp_id	emp_nationality	emp_name
1000	Chinese	Edmond
1000	Chinese	Edmond
1001	Korean	Brian
1002	Indian	Ashwin

**Table 2 (Department Table):**

emp_dept	emp_dept_num
Purchasing	200
HR	15
R&D	45
Technical Support	100

**Emp\_dept\_combined\_mapping table:**

emp_id	emp_deptNum
1000	Purchasing
1000	HR
1001	R&D
1002	Technical Support

**Functional Dependencies:**

- 1) emp\_id -> {emp\_nationality, emp\_name}
- 2) emp\_department -> dept\_num

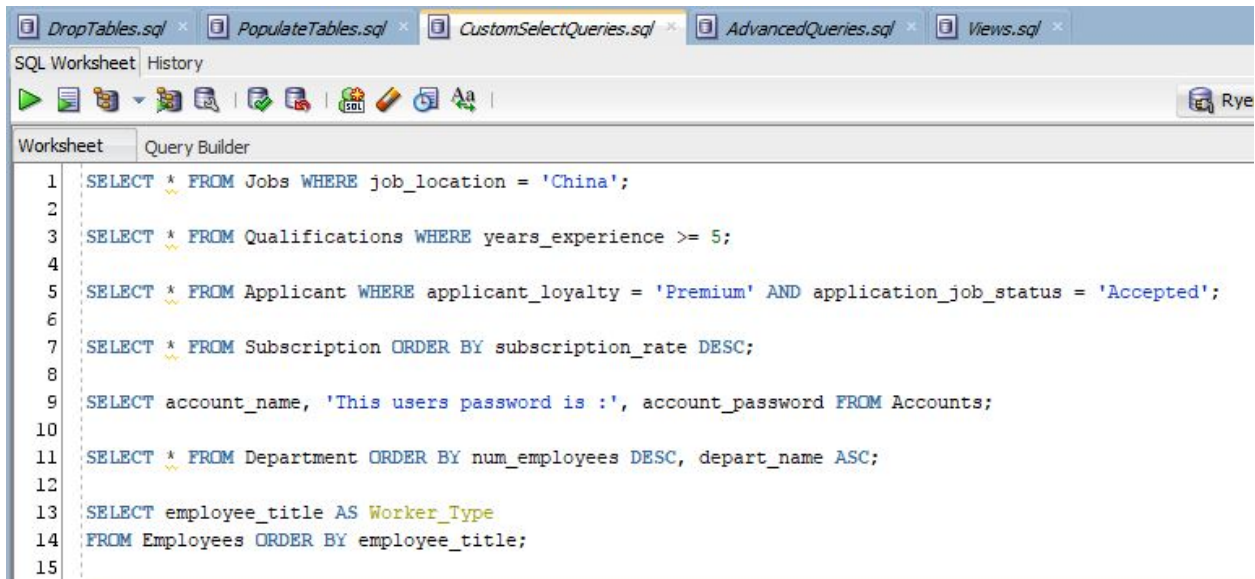
**Candidate Keys:**

**Table 1: emp\_id**

**Table 2: emp\_dept**

**Table 3: {emp\_id, emp\_dept}**

## Simple and Advanced Database Queries (with Relational Algebra):



The screenshot shows an SQL Worksheet application with a toolbar and a list of queries. The queries are as follows:

```
1 SELECT * FROM Jobs WHERE job_location = 'China';
2
3 SELECT * FROM Qualifications WHERE years_experience >= 5;
4
5 SELECT * FROM Applicant WHERE applicant_loyalty = 'Premium' AND application_job_status = 'Accepted';
6
7 SELECT * FROM Subscription ORDER BY subscription_rate DESC;
8
9 SELECT account_name, 'This users password is :', account_password FROM Accounts;
10
11 SELECT * FROM Department ORDER BY num_employees DESC, depart_name ASC;
12
13 SELECT employee_title AS Worker_Type
14 FROM Employees ORDER BY employee_title;
15
```

### SQL

```
SELECT *
    FROM Jobs
    WHERE job_location = 'China';
```

### Relational Algebra

$\sigma_{\text{Location} = \text{'China'}} (\text{Jobs})$

### SQL

```
SELECT *
    FROM Qualifications
    WHERE years_experience >= 5;
```

### Relational Algebra

$\sigma_{\text{year\_experience} \geq 5} (\text{Qualifications})$

### SQL

```
SELECT *
    FROM Applicant
    WHERE applicant_loyalty = 'Premium' AND application_job_status = 'Accepted';
```

### Relational Algebra

$\sigma_{(\text{applicant\_loyalty} = \text{'Premium'}) \text{ AND } (\text{application\_job\_status} = \text{'Accepted'})} (\text{Applicant})$

## SQL

```
SELECT *  
    FROM Department  
    ORDER BY num_employees DESC, depart_name ASC;
```

## Relational Algebra

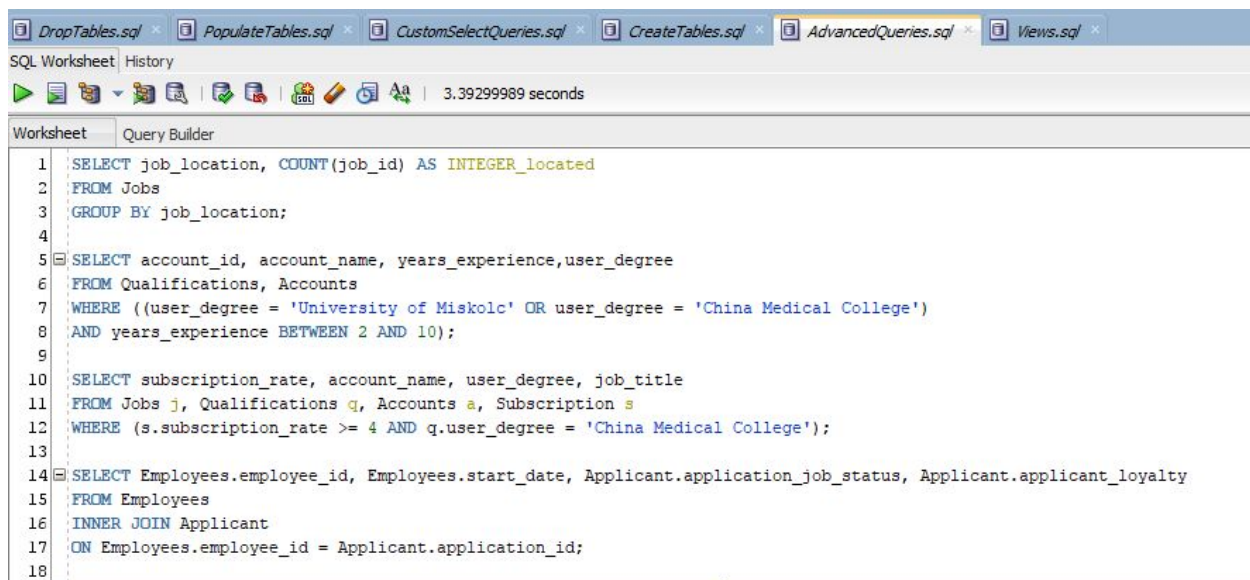
$\tau_{(\text{num\_employees DESC}) \text{ AND } (\text{depart\_name ASC})}(\text{Department})$

## SQL

```
SELECT employee_title AS Worker_Type  
    FROM Employees  
    ORDER BY employee_title;
```

## Relational Algebra

$\tau_{\text{num\_employees ASC}} \Pi_{\text{employee\_title}}(\text{Employees})$



## SQL

```
SELECT job_location, COUNT(job_id) AS INTEGER_located  
    FROM Jobs  
    GROUP BY job_location;
```

## Relational Algebra

$\Pi_{\text{job\_location, job\_id}}(\mathcal{G}_{\text{job\_location}})(\text{Jobs})$

## SQL

```
SELECT account_id, account_name, years_experience, user_degree  
    FROM Qualifications, Accounts
```

WHERE ((user\_degree = 'University of Miskolc' OR user\_degree = 'China Medical College')  
AND years\_experience BETWEEN 2 AND 10);

### Relational Algebra

$\Pi_{\text{account\_id, account\_name, years\_experience, user\_degree}}(\sigma_{(\text{user\_degree} = \text{'University of Miskolc'} \text{ OR } \text{user\_degree} = \text{'China Medical College'}) \text{ AND } (\text{year\_experience} \geq 2) \text{ AND } (\text{year\_experience} \leq 10)})(\text{Qualifications} \bowtie \text{Accounts})$

### SQL

SELECT subscription\_rate, account\_name, user\_degree, job\_title  
FROM Jobs j, Qualifications q, Accounts a, Subscription s  
WHERE (s.subscription\_rate >= 4 AND q.user\_degree = 'China Medical College');

### Relational Algebra

$\Pi_{\text{subscription\_rate, account\_name, user\_degree, job\_title}}(\sigma_{(\text{subscription\_rate} \geq 4) \text{ AND } (\text{user\_degree} = \text{'China Medical College'})})(\text{Jobs} \bowtie \text{Qualifications} \bowtie \text{Accounts} \bowtie \text{Subscription})$

### SQL

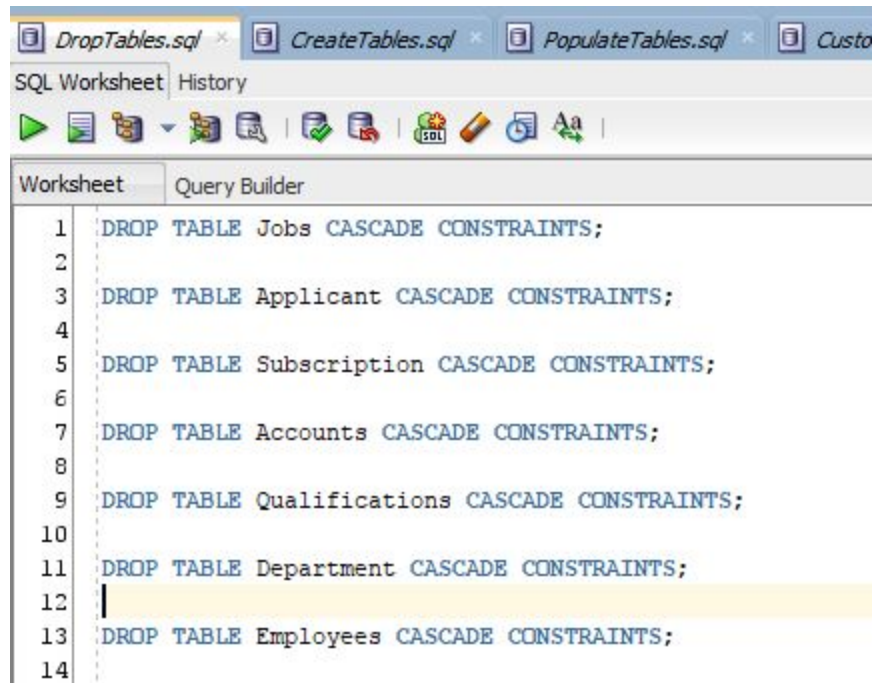
SELECT Employees.employee\_id, Employees.start\_date, Applicant.application\_job\_status,  
Applicant.applicant\_loyalty  
FROM Employees  
INNER JOIN Applicant  
ON Employees.employee\_id = Applicant.application\_id;

### Relational Algebra

$\Pi_{\text{employee\_id, start\_date, application\_job\_status, applicant\_loyalty}}(\sigma_{\text{employee\_id} = \text{applicant\_id}})(\text{Applicant} \bowtie \text{Employee})$

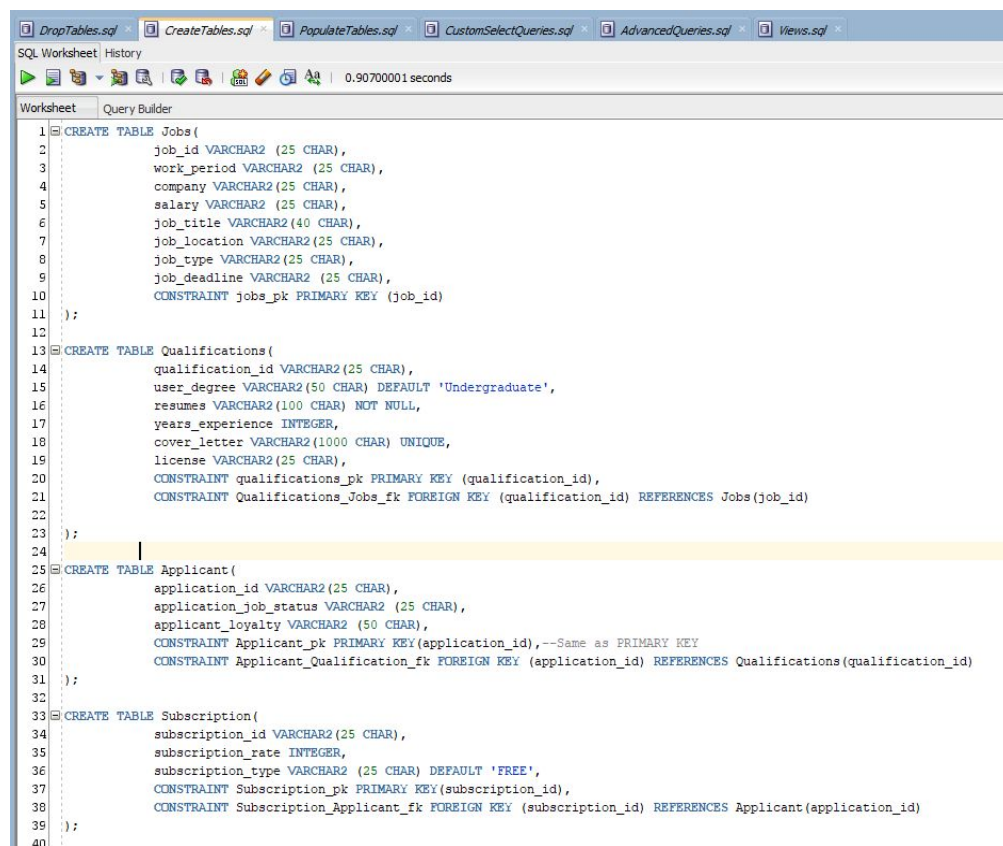


## SQL Code:



```
DropTables.sql * CreateTables.sql * PopulateTables.sql * Custo
SQL Worksheet History
Worksheet Query Builder
1 DROP TABLE Jobs CASCADE CONSTRAINTS;
2
3 DROP TABLE Applicant CASCADE CONSTRAINTS;
4
5 DROP TABLE Subscription CASCADE CONSTRAINTS;
6
7 DROP TABLE Accounts CASCADE CONSTRAINTS;
8
9 DROP TABLE Qualifications CASCADE CONSTRAINTS;
10
11 DROP TABLE Department CASCADE CONSTRAINTS;
12
13 DROP TABLE Employees CASCADE CONSTRAINTS;
14
```

## SQL Drop Tables



```
DropTables.sql * CreateTables.sql * PopulateTables.sql * CustomSelectQueries.sql * AdvancedQueries.sql * Views.sql *
SQL Worksheet History
Worksheet Query Builder
1 CREATE TABLE Jobs(
2     job_id VARCHAR2 (25 CHAR),
3     work_period VARCHAR2 (25 CHAR),
4     company VARCHAR2(25 CHAR),
5     salary VARCHAR2 (25 CHAR),
6     job_title VARCHAR2(40 CHAR),
7     job_location VARCHAR2(25 CHAR),
8     job_type VARCHAR2(25 CHAR),
9     job_deadline VARCHAR2 (25 CHAR),
10    CONSTRAINT jobs_pk PRIMARY KEY (job_id)
11 );
12
13 CREATE TABLE Qualifications(
14     qualification_id VARCHAR2(25 CHAR),
15     user_degree VARCHAR2(50 CHAR) DEFAULT 'Undergraduate',
16     resumes VARCHAR2(100 CHAR) NOT NULL,
17     years_experience INTEGER,
18     cover_letter VARCHAR2(1000 CHAR) UNIQUE,
19     license VARCHAR2(25 CHAR),
20     CONSTRAINT qualifications_pk PRIMARY KEY (qualification_id),
21     CONSTRAINT Qualifications_Jobs_fk FOREIGN KEY (qualification_id) REFERENCES Jobs(job_id)
22 );
23
24
25 CREATE TABLE Applicant(
26     application_id VARCHAR2(25 CHAR),
27     application_job_status VARCHAR2 (25 CHAR),
28     applicant_loyalty VARCHAR2 (50 CHAR),
29     CONSTRAINT Applicant_pk PRIMARY KEY(application_id),--Same as PRIMARY KEY
30     CONSTRAINT Applicant_Qualification_fk FOREIGN KEY (application_id) REFERENCES Qualifications(qualification_id)
31 );
32
33 CREATE TABLE Subscription(
34     subscription_id VARCHAR2(25 CHAR),
35     subscription_rate INTEGER,
36     subscription_type VARCHAR2 (25 CHAR) DEFAULT 'FREE',
37     CONSTRAINT Subscription_pk PRIMARY KEY(subscription_id),
38     CONSTRAINT Subscription_Applicant_fk FOREIGN KEY (subscription_id) REFERENCES Applicant(application_id)
39 );
40
```

```

40
41 CREATE TABLE Accounts(
42     account_id VARCHAR2 (25 CHAR),
43     account_name VARCHAR2 (25 CHAR) NOT NULL,
44     account_password VARCHAR2 (25 CHAR) UNIQUE,
45     user_email VARCHAR2 (100 CHAR),
46     user_birthdate VARCHAR2 (40 CHAR),
47     user_address VARCHAR2(150 CHAR),
48     user_phone VARCHAR2 (25 CHAR) UNIQUE,
49     CONSTRAINT accounts_pk PRIMARY KEY (account_id),
50     CONSTRAINT Accounts_Applicant_fk FOREIGN KEY (account_id) REFERENCES Applicant(application_id)
51 );
52
53
54 CREATE TABLE Department(
55     depart_id VARCHAR2(25 CHAR),
56     depart_manager VARCHAR2(50 CHAR) NOT NULL,
57     depart_name VARCHAR2(50 CHAR),
58     num_employees INTEGER DEFAULT 0,
59     CONSTRAINT depart_pk PRIMARY KEY(depart_id),
60     CONSTRAINT Department_Jobs_fk FOREIGN KEY (depart_id) REFERENCES Jobs(job_id)
61 );
62
63
64 CREATE TABLE Employees(
65     employee_id VARCHAR2 (25 CHAR),
66     start_date VARCHAR2(25 CHAR),
67     employee_title VARCHAR2(50 CHAR),
68     CONSTRAINT Employees_pk PRIMARY KEY (employee_id),
69     CONSTRAINT Employees_Department_fk FOREIGN KEY (employee_id) REFERENCES Department(depart_id)
70 );
71
72

```

## SQL Create Tables

DropTables.sql	CreateTables.sql	PopulateTables.sql	CustomSelectQueries.sql	AdvancedQueries.sql	Views.sql										
SQL Worksheet: History															
0.35100001 seconds															
Worksheet Query Builder															
1	,	company,	salary,	job_title,	job_location,	job_type,	job_deadline	values ('Job#1',	'June 2023',	'Eabon',	'594960.91',	'Accountant I',	'Poland',	'Contract',	'2021-02-28 03:02:47');
2	,	company,	salary,	job_title,	job_location,	job_type,	job_deadline	values ('Job#2',	'Aug 2021',	'Photobug',	'597442.65',	'Recruiter',	'China',	'Part-time',	'2021-09-18 23:47:28');
3	,	company,	salary,	job_title,	job_location,	job_type,	job_deadline	values ('Job#3',	'Dec 2020',	'Npath',	'594123.85',	'Occupational Therapist',	'Indonesia',	'Internship',	'2021-09-15 06:40:08');
4	,	company,	salary,	job_title,	job_location,	job_type,	job_deadline	values ('Job#4',	'Apr 2020',	'Meevee',	'593863.70',	'Senior Cost Accountant',	'China',	'Hourly',	'2020-12-29 05:08:51');
5	,	company,	salary,	job_title,	job_location,	job_type,	job_deadline	values ('Job#5',	'May 2021',	'Twitterwire',	'567890.17',	'Compensation Analyst',	'Indonesia',	'Temporary',	'2021-07-26 02:59:29');
6	,	company,	salary,	job_title,	job_location,	job_type,	job_deadline	values ('Job#6',	'June 2023',	'Skinix',	'592008.53',	'Software Test Engineer IV',	'Argentina',	'Temporary',	'2021-11-21 02:06:08');
7	,	company,	salary,	job_title,	job_location,	job_type,	job_deadline	values ('Job#7',	'Dec 2020',	'Realpoint',	'558857.32',	'Project Manager',	'China',	'Full-time',	'2021-03-25 22:24:32');
8	,	company,	salary,	job_title,	job_location,	job_type,	job_deadline	values ('Job#8',	'Apr 2020',	'Zoomcast',	'560664.76',	'Financial Advisor',	'China',	'Temporary',	'2021-09-09 21:06:08');
9	,	company,	salary,	job_title,	job_location,	job_type,	job_deadline	values ('Job#9',	'Dec 2020',	'Shuffledrive',	'593700.02',	'Professor',	'Russia',	'Temporary',	'2021-08-04 12:27:29');
10	,	company,	salary,	job_title,	job_location,	job_type,	job_deadline	values ('Job#10',	'May 2021',	'Kymbo',	'593894.08',	'Quality Control Specialist',	'Japan',	'Part-time',	'2021-03-20 08:25:46');
11	,	company,	salary,	job_title,	job_location,	job_type,	job_deadline	values ('Job#11',	'Aug 2021',	'Thoughtblab',	'594525.39',	'Legal Assistant',	'Canada',	'Internship',	'2021-10-24 12:37:07');
12	,	company,	salary,	job_title,	job_location,	job_type,	job_deadline	values ('Job#12',	'June 2023',	'Skipfire',	'592415.02',	'Sales Representative',	'Czech Republic',	'Internship',	'2021-08-28 10:09:12');
13	,	company,	salary,	job_title,	job_location,	job_type,	job_deadline	values ('Job#13',	'May 2021',	'Quimba',	'590239.96',	'Account Executive',	'Indonesia',	'Contract',	'2021-08-04 06:25:37');
14	,	company,	salary,	job_title,	job_location,	job_type,	job_deadline	values ('Job#14',	'June 2023',	'Ooba',	'565394.41',	'Project Manager',	'China',	'Full-time',	'2021-05-04 02:42:33');
15	,	company,	salary,	job_title,	job_location,	job_type,	job_deadline	values ('Job#15',	'May 2021',	'Wikizz',	'565038.73',	'Staff Accountant I',	'Philippines',	'Part-time',	'2021-07-16 08:51:56');
16	,	company,	salary,	job_title,	job_location,	job_type,	job_deadline	values ('Job#16',	'May 2021',	'Skynoodle',	'563098.10',	'Executive Secretary',	'Indonesia',	'Hourly',	'2021-08-02 19:02:33');
17	,	company,	salary,	job_title,	job_location,	job_type,	job_deadline	values ('Job#17',	'May 2021',	'Skimia',	'550615.84',	'Professor',	'Botswana',	'Temporary',	'2021-02-08 03:20:09');
18	,	company,	salary,	job_title,	job_location,	job_type,	job_deadline	values ('Job#18',	'Aug 2022',	'Twitterworks',	'569451.85',	'Geological Engineer',	'Vietnam',	'Hourly',	'2021-04-24 04:12:50');
19	,	company,	salary,	job_title,	job_location,	job_type,	job_deadline	values ('Job#19',	'May 2021',	'Cogidoo',	'593237.87',	'Senior Sales Associate',	'Peru',	'Temporary',	'2021-01-08 11:25:57');
20	,	company,	salary,	job_title,	job_location,	job_type,	job_deadline	values ('Job#20',	'Apr 2020',	'Tazz',	'565555.67',	'Environmental Specialist',	'Brazil',	'Full-time',	'2021-04-07 06:58:38');
21	,	company,	salary,	job_title,	job_location,	job_type,	job_deadline	values ('Job#21',	'Aug 2022',	'Yata',	'563330.29',	'Accounting Assistant I',	'Indonesia',	'Contract',	'2021-05-22 13:08:08');
22	,	company,	salary,	job_title,	job_location,	job_type,	job_deadline	values ('Job#22',	'Aug 2022',	'Tagopia',	'556425.05',	'Research Assistant III',	'Tajikistan',	'Internship',	'2021-11-27 04:27:04');
23	,	company,	salary,	job_title,	job_location,	job_type,	job_deadline	values ('Job#23',	'Aug 2022',	'Zoonder',	'597542.25',	'Project Manager',	'Armenia',	'Internship',	'2021-03-26 00:06:00');
24	,	company,	salary,	job_title,	job_location,	job_type,	job_deadline	values ('Job#24',	'Aug 2022',	'Dynazys',	'585232.79',	'Nurse',	'Macedonia',	'Contract',	'2021-10-18 04:28:41');
25	,	company,	salary,	job_title,	job_location,	job_type,	job_deadline	values ('Job#25',	'Aug 2021',	'Meedoo',	'551918.94',	'Geologist IV',	'Brazil',	'Hourly',	'2021-07-30 00:56:08');
26	,	company,	salary,	job_title,	job_location,	job_type,	job_deadline	values ('Job#26',	'Dec 2020',	'Oyonder',	'572584.19',	null, null, 'Part-time',	'2020-12-02 22:14:55');		
27	,	company,	salary,	job_title,	job_location,	job_type,	job_deadline	values ('Job#27',	'Apr 2020',	'Fanoodle',	'574612.53',	'Occupational Therapist',	'Philippines',	'Internship',	'2021-05-26 15:52:25');
28	,	company,	salary,	job_title,	job_location,	job_type,	job_deadline	values ('Job#28',	'June 2023',	'Dabjam',	'579339.96',	'Budget/Accounting Analyst II',	'Brazil',	'Temporary',	'2021-08-20 07:20:10');
29	,	company,	salary,	job_title,	job_location,	job_type,	job_deadline	values ('Job#29',	'Aug 2022',	'Rhybox',	'569074.58',	'Recruiter',	'Uzbekistan',	'Temporary',	'2021-04-05 10:43:58');
30	,	company,	salary,	job_title,	job_location,	job_type,	job_deadline	values ('Job#30',	'Dec 2020',	'Jayo',	'598277.19',	'Electrical Engineer',	'Peru',	'Contract',	'2021-01-06 16:03:30');
31	,	company,	salary,	job_title,	job_location,	job_type,	job_deadline	values ('Job#31',	'Apr 2020',	'Mgworks',	'596990.84',	'Office Assistant I',	'Russia',	'Hourly',	'2021-08-12 01:38:09');
32	,	company,	salary,	job_title,	job_location,	job_type,	job_deadline	values ('Job#32',	'Apr 2020',	'Topdrive',	'565516.36',	'Technical Writer',	'Pakistan',	'Contract',	'2021-09-15 18:46:14');
33	,	company,	salary,	job_title,	job_location,	job_type,	job_deadline	values ('Job#33',	'Aug 2021',	'Topiclounge',	'573840.21',	'Assistant Manager',	'Argentina',	'Hourly',	'2021-01-11 18:59:30');
34	,	company,	salary,	job_title,	job_location,	job_type,	job_deadline	values ('Job#34',	'Aug 2022',	'Skyndu',	'589244.08',	'Teacher',	'Russia',	'Internship',	'2021-08-09 02:40:57');
35	,	company,	salary,	job_title,	job_location,	job_type,	job_deadline	values ('Job#35',	'Aug 2022',	'Miboo',	'553422.40',	'Geologist IV',	'Morocco',	'Hourly',	'2021-03-01 08:07:32');
36	,	company,	salary,	job_title,	job_location,	job_type,	job_deadline	values ('Job#36',	'Apr 2020',	'Blotagz',	'560480.10',	'VP Product Management',	'Poland',	'Part-time',	'2021-10-31 02:09:38');
37	,	company,	salary,	job_title,	job_location,	job_type,	job_deadline	values ('Job#37',	'Apr 2020',	'Wikivu',	'554362.62',	'Staff Accountant II',	'Germany',	'Internship',	'2021-04-19 22:53:10');
38	,	company,	salary,	job_title,	job_location,	job_type,	job_deadline	values ('Job#38',	'Aug 2021',	'Room',	'550626.66',	'Community Outreach Specialist',	'Nigeria',	'Hourly',	'2021-08-16 18:10:13');
39	,	company,	salary,	job_title,	job_location,	job_type,	job_deadline	values ('Job#39',	'June 2023',	'Thoughtmix',	'561553.70',	null, null, 'Hourly',	'2021-02-23 09:52:12');		
40	,	company,	salary,	job_title,	job_location,	job_type,	job_deadline	values ('Job#40',	'Apr 2020',	'Dabfeed',	'576867.97',	'Dental Hygienist',	'Mauritius',	'Hourly',	'2021-01-19 09:58:39');
41	,	company,	salary,	job_title,	job_location,	job_type,	job_deadline	values ('Job#41',	'May 2021',	'Fanoodle',	'571657.22',	null, null, 'Hourly',	'2021-02-28 15:13:28');		
42	,	company,	salary,	job_title,	job_location,	job_type,	job_deadline	values ('Job#42',	'Aug 2022',	'Topicstorm',	'554698.04',	'Statistician III',	'United States',	'Hourly',	'2021-09-10 22:21:55');



```

53 insert into Qualifications (qualification_id, user_degree, resumes, years_experience, cover_letter, license) values ('Job#1', 'Instituci  n Universitaria Iberoamericana', 'Fundamental impactful ability',
54 insert into Qualifications (qualification_id, user_degree, resumes, years_experience, cover_letter, license) values ('Job#2', 'University of Wichita', 'Virtual attitude-oriented workforce', 6, 'Vivamus ma
55 insert into Qualifications (qualification_id, user_degree, resumes, years_experience, cover_letter, license) values ('Job#3', 'Universidad de Buenos Aires', 'Multi-layered object-oriented internet solution
56 insert into Qualifications (qualification_id, user_degree, resumes, years_experience, cover_letter, license) values ('Job#4', 'Fundacion Escuela Colombiana de Rehabilitaci  n', 'Synchronised bandwidth-monit
57 insert into Qualifications (qualification_id, user_degree, resumes, years_experience, cover_letter, license) values ('Job#5', 'University of Wisconsin - Madison', 'Cloned bandwidth-monitored archive', 12,
58 insert into Qualifications (qualification_id, user_degree, resumes, years_experience, cover_letter, license) values ('Job#6', 'Joetsu University of Education', 'Phased coherent Graphical User Interface',
59 insert into Qualifications (qualification_id, user_degree, resumes, years_experience, cover_letter, license) values ('Job#7', 'Universidad Estatal a Distancia', 'Integrated zero tolerance task-force', 19,
60 insert into Qualifications (qualification_id, user_degree, resumes, years_experience, cover_letter, license) values ('Job#8', 'Shazic Pile College', 'Seamless dynamic database', 21, 'Morbi odio odio, ele
61 insert into Qualifications (qualification_id, user_degree, resumes, years_experience, cover_letter, license) values ('Job#9', 'University of Montana', 'Intuitive mobile application', 17, 'Nunc purus. Phas
62 insert into Qualifications (qualification_id, user_degree, resumes, years_experience, cover_letter, license) values ('Job#10', 'Hochschule Harze Wissenschaften (FH)', 'Implemented fresh-thinking moderator
63 insert into Qualifications (qualification_id, user_degree, resumes, years_experience, cover_letter, license) values ('Job#11', 'Bastyr University', 'Robust web-enabled algorithm', 15, 'Integer aliquet, ma
64 insert into Qualifications (qualification_id, user_degree, resumes, years_experience, cover_letter, license) values ('Job#12', 'Universidad Central de Bayam  n', 'Customer-focused 4th generation open arch
65 insert into Qualifications (qualification_id, user_degree, resumes, years_experience, cover_letter, license) values ('Job#13', 'Mountain View College', 'Cloned holistic function', 12, 'Quisque erat eros,
66 insert into Qualifications (qualification_id, user_degree, resumes, years_experience, cover_letter, license) values ('Job#14', null, 'Versatile methodical success', null, null, 'Consequat Non-Durables');
67 insert into Qualifications (qualification_id, user_degree, resumes, years_experience, cover_letter, license) values ('Job#15', 'Goshen College', 'Multi-tiered dedicated methodology', 11, 'Fusce congue, di
68 insert into Qualifications (qualification_id, user_degree, resumes, years_experience, cover_letter, license) values ('Job#16', 'Mercy College', 'Synergetic 3rd generation contingency', 19, 'Nulla ut erat
69 insert into Qualifications (qualification_id, user_degree, resumes, years_experience, cover_letter, license) values ('Job#17', 'Ferris State University', 'Decentralized human-resource protocol', 5, 'Pellentesque
70 insert into Qualifications (qualification_id, user_degree, resumes, years_experience, cover_letter, license) values ('Job#18', 'Tashkent School of Finance', 'Seamless heuristic projection', 15, 'Nulla justo
71 insert into Qualifications (qualification_id, user_degree, resumes, years_experience, cover_letter, license) values ('Job#19', 'Islamic Azad University, Tabriz', 'Extended executive project', 13, 'Etiam v
72 insert into Qualifications (qualification_id, user_degree, resumes, years_experience, cover_letter, license) values ('Job#20', null, 'Open-source contextually-based synergy', null, null, 'Public Utilities
73 insert into Qualifications (qualification_id, user_degree, resumes, years_experience, cover_letter, license) values ('Job#21', '  cole Nationale Sup  rieure des Mines de Nancy', 'Self-enabling secondary an
74 insert into Qualifications (qualification_id, user_degree, resumes, years_experience, cover_letter, license) values ('Job#22', null, 'Networked scalable capability', null, null, 'Search Core');
75 insert into Qualifications (qualification_id, user_degree, resumes, years_experience, cover_letter, license) values ('Job#23', 'Augsburg College', 'De-engineered holistic synergy', 22, 'Nulla facilisi. Cu
76 insert into Qualifications (qualification_id, user_degree, resumes, years_experience, cover_letter, license) values ('Job#24', null, 'Cloned solution-oriented archive', null, null, 'Capital Goods');
77 insert into Qualifications (qualification_id, user_degree, resumes, years_experience, cover_letter, license) values ('Job#25', null, 'Optional tangible installation', null, null, 'Finance');
78 insert into Qualifications (qualification_id, user_degree, resumes, years_experience, cover_letter, license) values ('Job#26', null, 'Programmable system-worthy focus group', null, null, 'Finance');
79 insert into Qualifications (qualification_id, user_degree, resumes, years_experience, cover_letter, license) values ('Job#27', 'Lucerne University of Applied Sciences and Arts', 'Enhanced impactful functi
80 insert into Qualifications (qualification_id, user_degree, resumes, years_experience, cover_letter, license) values ('Job#28', 'Dionysius University of Thrace', 'Grass-roots global neural-net', 2, 'Morbi
81 insert into Qualifications (qualification_id, user_degree, resumes, years_experience, cover_letter, license) values ('Job#29', 'Maarawa State University Keffi', 'Advanced real-time moderator', 23, 'Maecenas
82 insert into Qualifications (qualification_id, user_degree, resumes, years_experience, cover_letter, license) values ('Job#30', 'Institute of Teachers Education, Tnu Russein Om', 'Fully-configurable didac
83 insert into Qualifications (qualification_id, user_degree, resumes, years_experience, cover_letter, license) values ('Job#31', 'Virginia State University', 'Extended analyzing superstructure', 9, 'Maecenas
84 insert into Qualifications (qualification_id, user_degree, resumes, years_experience, cover_letter, license) values ('Job#32', 'St. Joseph's College New York, Suffolk Campus', 'Managed maximized emulatio
85 insert into Qualifications (qualification_id, user_degree, resumes, years_experience, cover_letter, license) values ('Job#33', 'China Medical College', 'Focused zero administration concept', 7, 'Proin at
86 insert into Qualifications (qualification_id, user_degree, resumes, years_experience, cover_letter, license) values ('Job#34', 'Spalding University', 'Reengineered bandwidth-monitored analyzer', 18, 'Donec u
87 insert into Qualifications (qualification_id, user_degree, resumes, years_experience, cover_letter, license) values ('Job#35', 'University of North Carolina at Charlotte', 'Fully-configurable object-orient
88 insert into Qualifications (qualification_id, user_degree, resumes, years_experience, cover_letter, license) values ('Job#36', 'Instituto Tecnol  gico de Le  n', 'User-friendly zero tolerance architecture
89 insert into Qualifications (qualification_id, user_degree, resumes, years_experience, cover_letter, license) values ('Job#37', 'Universidad Nacional de Lomas de Zamora', 'Total optimizing data-warehouse',
90 insert into Qualifications (qualification_id, user_degree, resumes, years_experience, cover_letter, license) values ('Job#38', null, 'Customizable regional website', null, null, 'n/a');

```

```

104 insert into Applicant ( application_id, application_job_status, applicant_loyalty) values ('Job#1', 'Interview', 'Premium');
105 insert into Applicant ( application_id, application_job_status, applicant_loyalty) values ('Job#2', 'Offer', 'Premium');
106 insert into Applicant ( application_id, application_job_status, applicant_loyalty) values ('Job#3', 'Accepted', 'Free');
107 insert into Applicant ( application_id, application_job_status, applicant_loyalty) values ('Job#4', null, 'Premium');
108 insert into Applicant ( application_id, application_job_status, applicant_loyalty) values ('Job#5', 'Open', 'Basic');
109 insert into Applicant ( application_id, application_job_status, applicant_loyalty) values ('Job#6', 'Closed', 'Free');
110 insert into Applicant ( application_id, application_job_status, applicant_loyalty) values ('Job#7', 'Waitlist', 'Premium');
111 insert into Applicant ( application_id, application_job_status, applicant_loyalty) values ('Job#8', null, 'Free');
112 insert into Applicant ( application_id, application_job_status, applicant_loyalty) values ('Job#9', 'Offer', 'Basic');
113 insert into Applicant ( application_id, application_job_status, applicant_loyalty) values ('Job#10', 'Accepted', 'Basic');
114 insert into Applicant ( application_id, application_job_status, applicant_loyalty) values ('Job#11', 'Declined', 'Premium');
115 insert into Applicant ( application_id, application_job_status, applicant_loyalty) values ('Job#12', 'Open', 'Free');
116 insert into Applicant ( application_id, application_job_status, applicant_loyalty) values ('Job#13', 'Closed', 'Basic');
117 insert into Applicant ( application_id, application_job_status, applicant_loyalty) values ('Job#14', 'Waitlist', 'Free');
118 insert into Applicant ( application_id, application_job_status, applicant_loyalty) values ('Job#15', 'Interview', 'Free');
119 insert into Applicant ( application_id, application_job_status, applicant_loyalty) values ('Job#16', 'Offer', 'Basic');
120 insert into Applicant ( application_id, application_job_status, applicant_loyalty) values ('Job#17', 'Accepted', 'Basic');
121 insert into Applicant ( application_id, application_job_status, applicant_loyalty) values ('Job#18', null, 'Free');
122 insert into Applicant ( application_id, application_job_status, applicant_loyalty) values ('Job#19', 'Open', 'Basic');
123 insert into Applicant ( application_id, application_job_status, applicant_loyalty) values ('Job#20', 'Closed', 'Premium');
124 insert into Applicant ( application_id, application_job_status, applicant_loyalty) values ('Job#21', 'Waitlist', 'Premium');
125 insert into Applicant ( application_id, application_job_status, applicant_loyalty) values ('Job#22', 'Interview', 'Free');
126 insert into Applicant ( application_id, application_job_status, applicant_loyalty) values ('Job#23', 'Offer', 'Premium');
127 insert into Applicant ( application_id, application_job_status, applicant_loyalty) values ('Job#24', 'Accepted', 'Premium');
128 insert into Applicant ( application_id, application_job_status, applicant_loyalty) values ('Job#25', 'Declined', 'Premium');
129 insert into Applicant ( application_id, application_job_status, applicant_loyalty) values ('Job#26', 'Open', 'Free');
130 insert into Applicant ( application_id, application_job_status, applicant_loyalty) values ('Job#27', 'Closed', 'Basic');
131 insert into Applicant ( application_id, application_job_status, applicant_loyalty) values ('Job#28', 'Waitlist', 'Basic');
132 insert into Applicant ( application_id, application_job_status, applicant_loyalty) values ('Job#29', 'Interview', 'Basic');
133 insert into Applicant ( application_id, application_job_status, applicant_loyalty) values ('Job#30', 'Offer', 'Free');
134 insert into Applicant ( application_id, application_job_status, applicant_loyalty) values ('Job#31', 'Accepted', 'Premium');
135 insert into Applicant ( application_id, application_job_status, applicant_loyalty) values ('Job#32', 'Declined', 'Free');
136 insert into Applicant ( application_id, application_job_status, applicant_loyalty) values ('Job#33', 'Open', 'Basic');
137 insert into Applicant ( application_id, application_job_status, applicant_loyalty) values ('Job#34', 'Closed', 'Basic');
138 insert into Applicant ( application_id, application_job_status, applicant_loyalty) values ('Job#35', 'Waitlist', 'Free');
139 insert into Applicant ( application_id, application_job_status, applicant_loyalty) values ('Job#36', 'Interview', 'Premium');
140 insert into Applicant ( application_id, application_job_status, applicant_loyalty) values ('Job#37', 'Offer', 'Free');
141 insert into Applicant ( application_id, application_job_status, applicant_loyalty) values ('Job#38', 'Accepted', 'Free');
142 insert into Applicant ( application_id, application_job_status, applicant_loyalty) values ('Job#39', null, 'Basic');
143 insert into Applicant ( application_id, application_job_status, applicant_loyalty) values ('Job#40', 'Open', 'Free');
144 insert into Applicant ( application_id, application_job_status, applicant_loyalty) values ('Job#41', 'Closed', 'Free');

```



```

155
156 insert into Subscription (subscription_id, subscription_rate, subscription_type) values ('Job#1', 0, 'Free');
157 insert into Subscription (subscription_id, subscription_rate, subscription_type) values ('Job#2', 4.99, 'Basic');
158 insert into Subscription (subscription_id, subscription_rate, subscription_type) values ('Job#3', 10.99, 'Premium');

```

```

160
161 insert into Accounts (account_id, account_name, account_password, user_email, user_birthdate, user_address, user_phone) values ('Job#1', 'ppatastone3', 'DDeeing', 'lamesa@loc.gov', '19/11/1981', '99227 Wai
162 insert into Accounts (account_id, account_name, account_password, user_email, user_birthdate, user_address, user_phone) values ('Job#2', 'desashrdier1', 'WkImM9vWjy', 'amesha@as.net', '23/10/2013', '
163 insert into Accounts (account_id, account_name, account_password, user_email, user_birthdate, user_address, user_phone) values ('Job#3', 'johantone0', 'DDeeing', 'hadda@loc.gov', '19/11/1981', '99227 G
164 insert into Accounts (account_id, account_name, account_password, user_email, user_birthdate, user_address, user_phone) values ('Job#4', 'desaudier1', 'WkImM9vWjy', 'tuminini@as.net', '13/10/2013', '70
165 insert into Accounts (account_id, account_name, account_password, user_email, user_birthdate, user_address, user_phone) values ('Job#5', 'awapol2', 'QV7TAQR', 'slee2@youtu.be', '15/12/2016', '123 Vermont 1
166 insert into Accounts (account_id, account_name, account_password, user_email, user_birthdate, user_address, user_phone) values ('Job#6', 'odcoman3', 'vWu1QyV7', 'ldeabrow3@apache.org', '22/06/2006', '5 1
167 insert into Accounts (account_id, account_name, account_password, user_email, user_birthdate, user_address, user_phone) values ('Job#7', 'afayne4', 'LcSEgDLs', 'dallanson4@igmodo.com', null, null, '130
168 insert into Accounts (account_id, account_name, account_password, user_email, user_birthdate, user_address, user_phone) values ('Job#8', 'casch5', 'h7FRKq', 'bhickin5@fcc.gov', '08/12/2017', '1295 Rockefe
169 insert into Accounts (account_id, account_name, account_password, user_email, user_birthdate, user_address, user_phone) values ('Job#9', 'cooklynaghe6', 'UuqUu', 'hbeed6@engadger.com', '14/05/1968',
170 insert into Accounts (account_id, account_name, account_password, user_email, user_birthdate, user_address, user_phone) values ('Job#10', 'mtitherington7', 'Q7uAZcZ3', 'bjoanic7@github.com', '06/12/2004',
171 insert into Accounts (account_id, account_name, account_password, user_email, user_birthdate, user_address, user_phone) values ('Job#11', 'dciochillo8', 'syipRm', 'ckeroft8@pages.com', '26/04/2020',
172 insert into Accounts (account_id, account_name, account_password, user_email, user_birthdate, user_address, user_phone) values ('Job#12', 'awipper9', 'BaSL6tKzIq', 'ophipp9@earthtimes.com', '16/05/196
173 insert into Accounts (account_id, account_name, account_password, user_email, user_birthdate, user_address, user_phone) values ('Job#13', 'shisacta', 'QZC63Hc4t2', 'fanterchama@ga.gov', '04/11/1981',
174 insert into Accounts (account_id, account_name, account_password, user_email, user_birthdate, user_address, user_phone) values ('Job#14', 'mascopyn0', 'F8uYXm6y', 'jcdard8@dailyemotion.com', '17/05/1993
175 insert into Accounts (account_id, account_name, account_password, user_email, user_birthdate, user_address, user_phone) values ('Job#15', 'dkenefick0', 'GWSvta', 'tcoollerc@people.com.cn', '19/06/1983',
176 insert into Accounts (account_id, account_name, account_password, user_email, user_birthdate, user_address, user_phone) values ('Job#16', 'lbrised', '887Lm7m', 'Klarned@agondesign.com', '05/01/1970', '0
177 insert into Accounts (account_id, account_name, account_password, user_email, user_birthdate, user_address, user_phone) values ('Job#17', 'ccorbitte', 'UEclCz7K', 'tmelleney8@alum.net', '25/02/1963', '1
178 insert into Accounts (account_id, account_name, account_password, user_email, user_birthdate, user_address, user_phone) values ('Job#18', 'caexten', 'bWLnCvz', 'bspeakef@psu.edu', '30/07/2007', '435 Hoc
179 insert into Accounts (account_id, account_name, account_password, user_email, user_birthdate, user_address, user_phone) values ('Job#19', 'aveazsg', 'LuyOVRY', 'mqogerty8@forbes.com', '06/02/1988', '7 Bur
180 insert into Accounts (account_id, account_name, account_password, user_email, user_birthdate, user_address, user_phone) values ('Job#20', 'clewdidgeh', 'JXzU9v3a3', 'ymowleyh@lundi.de', '23/09/2011', '
181 insert into Accounts (account_id, account_name, account_password, user_email, user_birthdate, user_address, user_phone) values ('Job#21', 'icroally', 'McB1jC04tG2', 'vgublarj@wfo.com', '16/09/2018', '4
182 insert into Accounts (account_id, account_name, account_password, user_email, user_birthdate, user_address, user_phone) values ('Job#22', 'wborlowk', 'oFlnFmBS5D', 'mkimkievch@kyourube.com', '28/01/19
183 insert into Accounts (account_id, account_name, account_password, user_email, user_birthdate, user_address, user_phone) values ('Job#23', 'bshgard1', 'QpK80Ry', 'mfrabner1@auda.org.au', '03/07/1975', '0
184 insert into Accounts (account_id, account_name, account_password, user_email, user_birthdate, user_address, user_phone) values ('Job#24', 'agehrtm', 'nolcLrQdVtD', 'sharenan@epac.org', '22/12/2000',
185 insert into Accounts (account_id, account_name, account_password, user_email, user_birthdate, user_address, user_phone) values ('Job#25', 'adegreyn', 'fmk1EKp1ib', 'nanncin@ymantec.com', '18/03/2018',
186 insert into Accounts (account_id, account_name, account_password, user_email, user_birthdate, user_address, user_phone) values ('Job#26', 'ctcoopo', 'lufjyf', 'lsybyey@kickstarter.com', '08/12/1993', '4
187 insert into Accounts (account_id, account_name, account_password, user_email, user_birthdate, user_address, user_phone) values ('Job#27', 'gdrezerp', 'alv6iHZA', 'jmarchip@creativecommons.org', '14/08/1
188 insert into Accounts (account_id, account_name, account_password, user_email, user_birthdate, user_address, user_phone) values ('Job#28', 'kaleksandrovq', 'obvi7V3', 'lritleyq@type.com', '02/12/1980',
189 insert into Accounts (account_id, account_name, account_password, user_email, user_birthdate, user_address, user_phone) values ('Job#29', 'wgerisson', 'dXZp2H1', 'sejepsenen@army.mil', '29/01/2018', '20
190 insert into Accounts (account_id, account_name, account_password, user_email, user_birthdate, user_address, user_phone) values ('Job#30', 'tpieruccis', 'lBtkhk', 'aswatman@espy.cz', '15/09/1961', '2725
191 insert into Accounts (account_id, account_name, account_password, user_email, user_birthdate, user_address, user_phone) values ('Job#31', 'mmarlorr', 'e0i335z', 'vbenoit@technorati.com', '25/09/1967', '2
192 insert into Accounts (account_id, account_name, account_password, user_email, user_birthdate, user_address, user_phone) values ('Job#32', 'jbredu', 'bDCFmalBN', 'sgoudon@openresource.org', null, null,
193 insert into Accounts (account_id, account_name, account_password, user_email, user_birthdate, user_address, user_phone) values ('Job#33', 'wbyman', 'vrtspz3p1d', 'zvirgigin@rees.com', null, null, '1455
194 insert into Accounts (account_id, account_name, account_password, user_email, user_birthdate, user_address, user_phone) values ('Job#34', 'kgogyn', 'sy2u30f5un', 'hmcourtow@smugmug.com', '15/11/1965',
195 insert into Accounts (account_id, account_name, account_password, user_email, user_birthdate, user_address, user_phone) values ('Job#35', 'nmayoxrs', 'qBpE3s1fcr', 'abedurrow@ted.com', '26/02/2006', '1
196 insert into Accounts (account_id, account_name, account_password, user_email, user_birthdate, user_address, user_phone) values ('Job#36', 'jbraneyy', 'Xf6ZgkR', 'rbusbny@utexas.edu', '19/08/2015', '192
197 insert into Accounts (account_id, account_name, account_password, user_email, user_birthdate, user_address, user_phone) values ('Job#37', 'dhunnicotz', 'GWNtH2UR', 'lrcowley8@tervista.org', '07/02/2000
198 insert into Accounts (account_id, account_name, account_password, user_email, user_birthdate, user_address, user_phone) values ('Job#38', 'apuvial0', 'UfoOH90', 'frukat10@mail.ru', '01/12/1991', '1 Dahle

```

```

217 insert into Department (depart_id, depart_manager, depart_name, num_employees) values ('Job#2', 'Hallsy', 'Product Management', 105);
218 insert into Department (depart_id, depart_manager, depart_name, num_employees) values ('Job#3', 'Samuele', 'Research and Development', 294);
219 insert into Department (depart_id, depart_manager, depart_name, num_employees) values ('Job#4', 'Linc', 'Accounting', 142);
220 insert into Department (depart_id, depart_manager, depart_name, num_employees) values ('Job#5', 'Mendy', 'Training', 221);
221 insert into Department (depart_id, depart_manager, depart_name, num_employees) values ('Job#6', 'Munroe', 'Support', 224);
222 insert into Department (depart_id, depart_manager, depart_name, num_employees) values ('Job#7', 'Garrett', 'Engineering', 89);
223 insert into Department (depart_id, depart_manager, depart_name, num_employees) values ('Job#8', 'Laurence', 'Services', 174);
224 insert into Department (depart_id, depart_manager, depart_name, num_employees) values ('Job#9', 'Base', 'Support', 196);
225 insert into Department (depart_id, depart_manager, depart_name, num_employees) values ('Job#10', 'Brooks', 'Support', 165);
226 insert into Department (depart_id, depart_manager, depart_name, num_employees) values ('Job#11', 'Charlie', 'Sales', 25);
227 insert into Department (depart_id, depart_manager, depart_name, num_employees) values ('Job#12', 'Goran', 'Research and Development', 50);
228 insert into Department (depart_id, depart_manager, depart_name, num_employees) values ('Job#13', 'Reynard', 'Research and Development', 297);
229 insert into Department (depart_id, depart_manager, depart_name, num_employees) values ('Job#14', 'Dillie', 'Legal', null);
230 insert into Department (depart_id, depart_manager, depart_name, num_employees) values ('Job#15', 'Edlin', 'Research and Development', 117);
231 insert into Department (depart_id, depart_manager, depart_name, num_employees) values ('Job#16', 'Filmore', 'Legal', 197);
232 insert into Department (depart_id, depart_manager, depart_name, num_employees) values ('Job#17', 'Deck', 'Product Management', 41);
233 insert into Department (depart_id, depart_manager, depart_name, num_employees) values ('Job#18', 'Noe', 'Training', 70);
234 insert into Department (depart_id, depart_manager, depart_name, num_employees) values ('Job#19', 'Webster', 'Training', 67);
235 insert into Department (depart_id, depart_manager, depart_name, num_employees) values ('Job#20', 'Rob', 'Training', 141);
236 insert into Department (depart_id, depart_manager, depart_name, num_employees) values ('Job#21', 'Lewiss', 'Human Resources', 154);
237 insert into Department (depart_id, depart_manager, depart_name, num_employees) values ('Job#22', 'Garrik', 'Business Development', 247);
238 insert into Department (depart_id, depart_manager, depart_name, num_employees) values ('Job#23', 'Wake', 'Accounting', 249);
239 insert into Department (depart_id, depart_manager, depart_name, num_employees) values ('Job#24', 'Merrill', 'Research and Development', 226);
240 insert into Department (depart_id, depart_manager, depart_name, num_employees) values ('Job#25', 'Fairleigh', 'Accounting', 184);
241 insert into Department (depart_id, depart_manager, depart_name, num_employees) values ('Job#26', 'Waring', 'Services', null);
242 insert into Department (depart_id, depart_manager, depart_name, num_employees) values ('Job#27', 'Gibb', 'Business Development', null);
243 insert into Department (depart_id, depart_manager, depart_name, num_employees) values ('Job#28', 'Hew', 'Marketing', 38);
244 insert into Department (depart_id, depart_manager, depart_name, num_employees) values ('Job#29', 'Lucian', 'Research and Development', null);
245 insert into Department (depart_id, depart_manager, depart_name, num_employees) values ('Job#30', 'Mackenzie', 'Human Resources', 235);
246 insert into Department (depart_id, depart_manager, depart_name, num_employees) values ('Job#31', 'Hurleigh', 'Engineering', 287);
247 insert into Department (depart_id, depart_manager, depart_name, num_employees) values ('Job#32', 'Berkeley', 'Services', null);
248 insert into Department (depart_id, depart_manager, depart_name, num_employees) values ('Job#33', 'Somerset', 'Human Resources', 144);
249 insert into Department (depart_id, depart_manager, depart_name, num_employees) values ('Job#34', 'Alexei', 'Sales', 251);
250 insert into Department (depart_id, depart_manager, depart_name, num_employees) values ('Job#35', 'Darren', 'Accounting', 124);
251 insert into Department (depart_id, depart_manager, depart_name, num_employees) values ('Job#36', 'Marcel', 'Support', null);
252 insert into Department (depart_id, depart_manager, depart_name, num_employees) values ('Job#37', 'Niall', 'Engineering', null);
253 insert into Department (depart_id, depart_manager, depart_name, num_employees) values ('Job#38', 'Keir', 'Services', 197);
254 insert into Department (depart_id, depart_manager, depart_name, num_employees) values ('Job#39', 'Franzen', 'Product Management', null);
255 insert into Department (depart_id, depart_manager, depart_name, num_employees) values ('Job#40', 'Halsy', 'Engineering', 97);
256 insert into Department (depart_id, depart_manager, depart_name, num_employees) values ('Job#41', 'Christie', 'Marketing', 32);
257 insert into Department (depart_id, depart_manager, depart_name, num_employees) values ('Job#42', 'Del', 'Support', 177);
258 insert into Department (depart_id, depart_manager, depart_name, num_employees) values ('Job#43', 'Kahaleel', 'Human Resources', null);

```



```

267 insert into Employees ( employee_id, start_date, employee_title) values ('Job#1', '30/09/2019', 'Statistician III');
268 insert into Employees ( employee_id, start_date, employee_title) values ('Job#2', '14/02/2020', 'Senior Sales Associate');
269 insert into Employees ( employee_id, start_date, employee_title) values ('Job#3', '29/06/2019', 'Administrative Officer');
270 insert into Employees ( employee_id, start_date, employee_title) values ('Job#4', '20/06/2019', 'Electrical Engineer');
271 insert into Employees ( employee_id, start_date, employee_title) values ('Job#5', null, 'Systems Administrator II');
272 insert into Employees ( employee_id, start_date, employee_title) values ('Job#6', '20/11/2019', 'VP Sales');
273 insert into Employees ( employee_id, start_date, employee_title) values ('Job#7', '14/07/2020', 'Executive Secretary');
274 insert into Employees ( employee_id, start_date, employee_title) values ('Job#8', null, 'Safety Technician I');
275 insert into Employees ( employee_id, start_date, employee_title) values ('Job#9', '16/10/2019', 'Human Resources Manager');
276 insert into Employees ( employee_id, start_date, employee_title) values ('Job#10', '29/12/2018', 'Environmental Specialist');
277 insert into Employees ( employee_id, start_date, employee_title) values ('Job#11', '10/12/2018', 'Occupational Therapist');
278 insert into Employees ( employee_id, start_date, employee_title) values ('Job#12', '01/01/2019', 'Financial Advisor');
279 insert into Employees ( employee_id, start_date, employee_title) values ('Job#13', '08/06/2020', 'Statistician III');
280 insert into Employees ( employee_id, start_date, employee_title) values ('Job#14', '13/09/2020', 'Programmer Analyst I');
281 insert into Employees ( employee_id, start_date, employee_title) values ('Job#15', '28/03/2019', 'Assistant Media Planner');
282 insert into Employees ( employee_id, start_date, employee_title) values ('Job#16', '05/09/2020', 'Biostatistician IV');
283 insert into Employees ( employee_id, start_date, employee_title) values ('Job#17', '19/10/2020', 'Speech Pathologist');
284 insert into Employees ( employee_id, start_date, employee_title) values ('Job#18', '04/06/2020', 'General Manager');
285 insert into Employees ( employee_id, start_date, employee_title) values ('Job#19', '17/10/2020', 'Desktop Support Technician');
286 insert into Employees ( employee_id, start_date, employee_title) values ('Job#20', null, 'Geological Engineer');
287 insert into Employees ( employee_id, start_date, employee_title) values ('Job#21', '15/05/2020', 'VP Marketing');
288 insert into Employees ( employee_id, start_date, employee_title) values ('Job#22', '10/08/2019', 'General Manager');
289 insert into Employees ( employee_id, start_date, employee_title) values ('Job#23', null, 'Nurse');
290 insert into Employees ( employee_id, start_date, employee_title) values ('Job#24', '03/09/2020', 'Marketing Assistant');
291 insert into Employees ( employee_id, start_date, employee_title) values ('Job#25', '14/12/2019', 'Speech Pathologist');
292 insert into Employees ( employee_id, start_date, employee_title) values ('Job#26', '14/04/2020', 'Compensation Analyst');
293 insert into Employees ( employee_id, start_date, employee_title) values ('Job#27', null, 'Account Executive');
294 insert into Employees ( employee_id, start_date, employee_title) values ('Job#28', null, 'Social Worker');
295 insert into Employees ( employee_id, start_date, employee_title) values ('Job#29', '18/09/2020', 'Cost Accountant');
296 insert into Employees ( employee_id, start_date, employee_title) values ('Job#30', '08/04/2019', 'Data Coordinator');
297 insert into Employees ( employee_id, start_date, employee_title) values ('Job#31', '31/08/2019', 'Programmer Analyst IV');
298 insert into Employees ( employee_id, start_date, employee_title) values ('Job#32', '28/01/2019', 'Statistician III');
299 insert into Employees ( employee_id, start_date, employee_title) values ('Job#33', '27/12/2019', 'Accounting Assistant II');
300 insert into Employees ( employee_id, start_date, employee_title) values ('Job#34', '25/11/2019', 'Assistant Professor');
301 insert into Employees ( employee_id, start_date, employee_title) values ('Job#35', '10/06/2020', 'Structural Analysis Engineer');
302 insert into Employees ( employee_id, start_date, employee_title) values ('Job#36', '09/09/2020', 'Analyst Programmer');
303 insert into Employees ( employee_id, start_date, employee_title) values ('Job#37', '19/03/2020', 'Help Desk Technician');
304 insert into Employees ( employee_id, start_date, employee_title) values ('Job#38', null, 'Legal Assistant');
305 insert into Employees ( employee_id, start_date, employee_title) values ('Job#39', '21/12/2019', 'Programmer Analyst III');
306 insert into Employees ( employee_id, start_date, employee_title) values ('Job#40', '17/07/2020', 'Analog Circuit Design manager');

```

## SQL Populate Tables

## **JAVA GUI Code:**

### **Database.java**

```
/*
 * To change this license header, choose License Headers in Project Properties.
 * To change this template file, choose Tools | Templates
 * and open the template in the editor.
 */
package com.edmond.jobbank.mavenproject1;

import java.sql.*;

/**
 *
 * @author Edmond
 */
public class Database {
    static Connection conn = null;

    public static Connection getConnectionToDB(){
        try {
            String serverName = "oracle.scs.ryerson.ca";
            String serverPort = "1521";
            String sid = "orcl";
            String url = "jdbc:oracle:thin:@" + serverName + ":" +
                serverPort + ":" + sid;
            String username = "p6chu";
            String password = "06084595";
            conn = DriverManager.getConnection(url,username,password);
        } catch (SQLException e) {
            System.err.format("SQL State: %s\n%s", e.getSQLState(), e.getMessage());
        } catch (Exception e) {
            e.printStackTrace();
        }
        return conn;
    }
}
```

### **JobBankUI.java**

```
/*
 * To change this license header, choose License Headers in Project Properties.
 * To change this template file, choose Tools | Templates
 * and open the template in the editor.
 */
package com.edmond.jobbank.mavenproject1;
```



```

/**
 *
 * @author Edmond
 */
public class JobBankUI {
    /**
     * @param args the command line arguments
     */
    public static void main(String[] args) {
        Intro introPanel = Intro.getInstance();
        introPanel.setVisible(true);
    }
}

```

Account.java

```

/**
 * To change this license header, choose License Headers in Project Properties.
 * To change this template file, choose Tools | Templates
 * and open the template in the editor.
 */
package com.edmond.jobbank.mavenproject1;

import java.io.*;
import java.sql.*;
import org.apache.ibatis.jdbc.ScriptRunner;

/**
 *
 * @author Brian
 */
public class Account extends javax.swing.JFrame {

    private static FrontPage frontpage;
    private static Account accountpage;
    static Connection con;
    private static String[] account;
    /**
     * Creates new form Account
     */
    private Account() {
        initComponents();
    }

    public static Account getInstance(FrontPage fPanel, Connection conn, String[] accountS) {
        if(accountpage == null) {
            accountpage = new Account();
            con = conn;
            frontpage = fPanel;
            account = accountS;

```

```

    }
    return accountpage;
}

public void updateText() {
    UserUsername_jlabel.setText(account[1]);
    UserPassword_jlabel.setText(account[2]);
    UserName_jlabel.setText(account[3]);
    UserEmail_jlabel.setText(account[4]);
    UserBirthdate_jlabel.setText(account[5]);
    UserAddress_jlabel.setText(account[6]);
    UserPhone_jlabel.setText(account[7]);
}

/**
 * This method is called from within the constructor to initialize the form.
 * WARNING: Do NOT modify this code. The content of this method is always
 * regenerated by the Form Editor.
 */
@SuppressWarnings("unchecked")
// <editor-fold defaultstate="collapsed" desc="Generated Code">//GEN-BEGIN: initComponents
private void initComponents() {

    jPanel1 = new javax.swing.JPanel();
    Close_button = new javax.swing.JButton();
    Phone_jlabel = new javax.swing.JLabel();
    Password_jlabel = new javax.swing.JLabel();
    Email_jlabel = new javax.swing.JLabel();
    Name_jlabel = new javax.swing.JLabel();
    Username_jlabel = new javax.swing.JLabel();
    Birthdate_jlabel = new javax.swing.JLabel();
    Title = new javax.swing.JLabel();
    Address_jlabel = new javax.swing.JLabel();
    UserUsername_jlabel = new javax.swing.JLabel();
    UserPassword_jlabel = new javax.swing.JLabel();
    UserSubscription_jlabel = new javax.swing.JLabel();
    UserName_jlabel = new javax.swing.JLabel();
    UserAddress_jlabel = new javax.swing.JLabel();
    UserBirthdate_jlabel = new javax.swing.JLabel();
    UserPhone_jlabel = new javax.swing.JLabel();
    Subscription_jlabel = new javax.swing.JLabel();
    Back_Button = new javax.swing.JButton();
    UserEmail_jlabel = new javax.swing.JLabel();
    ChangeSubscription_Button = new javax.swing.JButton();
    Qualifications_button = new javax.swing.JButton();

    setDefaultCloseOperation(javax.swing.WindowConstants.EXIT_ON_CLOSE);

    Close_button.setText("Close");

```

```

Close_button.addActionListener(new java.awt.event.ActionListener() {
    public void actionPerformed(java.awt.event.ActionEvent evt) {
        Close_buttonActionPerformed(evt);
    }
});

Phone_jlabel.setFont(new java.awt.Font("Tahoma", 0, 12)); // NOI18N
Phone_jlabel.setText("Phone");

Password_jlabel.setFont(new java.awt.Font("Tahoma", 0, 12)); // NOI18N
Password_jlabel.setText("Password");

Email_jlabel.setFont(new java.awt.Font("Tahoma", 0, 12)); // NOI18N
Email_jlabel.setText("Email");

Name_jlabel.setFont(new java.awt.Font("Tahoma", 0, 12)); // NOI18N
Name_jlabel.setText("Name");

Username_jlabel.setFont(new java.awt.Font("Tahoma", 0, 12)); // NOI18N
Username_jlabel.setText("Username");

Birthdate_jlabel.setFont(new java.awt.Font("Tahoma", 0, 12)); // NOI18N
Birthdate_jlabel.setText("Birthdate");

Title.setFont(new java.awt.Font("Tahoma", 0, 24)); // NOI18N
Title.setText("Account Details");

Address_jlabel.setFont(new java.awt.Font("Tahoma", 0, 12)); // NOI18N
Address_jlabel.setText("Address");

UserUsername_jlabel.setFont(new java.awt.Font("Tahoma", 0, 12)); // NOI18N

UserPassword_jlabel.setFont(new java.awt.Font("Tahoma", 0, 12)); // NOI18N

UserSubscription_jlabel.setFont(new java.awt.Font("Tahoma", 0, 12)); // NOI18N

UserName_jlabel.setFont(new java.awt.Font("Tahoma", 0, 12)); // NOI18N

UserAddress_jlabel.setFont(new java.awt.Font("Tahoma", 0, 12)); // NOI18N

UserBirthdate_jlabel.setFont(new java.awt.Font("Tahoma", 0, 12)); // NOI18N

UserPhone_jlabel.setFont(new java.awt.Font("Tahoma", 0, 12)); // NOI18N

Subscription_jlabel.setFont(new java.awt.Font("Tahoma", 0, 12)); // NOI18N
Subscription_jlabel.setText("Subscription");

Back_Button.setText("Back");
Back_Button.addActionListener(new java.awt.event.ActionListener() {

```

```

        public void actionPerformed(java.awt.event.ActionEvent evt) {
            Back_ButtonActionPerformed(evt);
        }
    });

    UserEmail_jlabel.setFont(new java.awt.Font("Tahoma", 0, 12)); // NOI18N

    ChangeSubscription_Button.setText("Change Subscription");
    ChangeSubscription_Button.addActionListener(new java.awt.event.ActionListener() {
        public void actionPerformed(java.awt.event.ActionEvent evt) {
            ChangeSubscription_ButtonActionPerformed(evt);
        }
    });

    Qualifications_button.setText("Qualifications");
    Qualifications_button.addActionListener(new java.awt.event.ActionListener() {
        public void actionPerformed(java.awt.event.ActionEvent evt) {
            Qualifications_buttonActionPerformed(evt);
        }
    });

    javax.swing.GroupLayout jPanel1Layout = new javax.swing.GroupLayout(jPanel1);
    jPanel1.setLayout(jPanel1Layout);
    jPanel1Layout.setHorizontalGroup(
        jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
            .addGroup(jPanel1Layout.createSequentialGroup()
                .addGap(20, 20, 20)
                .addGroup(jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
                    .addGroup(jPanel1Layout.createSequentialGroup()
                        .addGap(33, 33, 33)
                        .addGroup(jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.TRAILING)
                            .addGroup(jPanel1Layout.createSequentialGroup()
                                .addGroup(jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING, false)
                                    .addGroup(jPanel1Layout.createSequentialGroup()
                                        .addComponent(Address_jlabel)
                                        .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED,
                                            javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE)
                                        .addComponent(UserAddress_jlabel,
                                            javax.swing.GroupLayout.PREFERRED_SIZE, 260, javax.swing.GroupLayout.PREFERRED_SIZE))
                                    .addGroup(jPanel1Layout.createSequentialGroup()
                                        .addComponent(Birthdate_jlabel)
                                        .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED, 97,
                                            Short.MAX_VALUE)
                                        .addComponent(UserBirthdate_jlabel,
                                            javax.swing.GroupLayout.PREFERRED_SIZE, 260, javax.swing.GroupLayout.PREFERRED_SIZE))
                                    .addGroup(jPanel1Layout.createSequentialGroup()
                                        .addComponent(Name_jlabel)

```

```

        .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED,
javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE)
        .addComponent(UserName_jlabel, javax.swing.GroupLayout.PREFERRED_SIZE,
260, javax.swing.GroupLayout.PREFERRED_SIZE))
        .addGroup(javax.swing.GroupLayout.Alignment.TRAILING,
jPanel1Layout.createSequentialGroup())

    .addGroup(jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
        .addComponent(Phone_jlabel)
        .addComponent(Email_jlabel))
        .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED,
javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE)

    .addGroup(jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING, false)
        .addComponent(UserPhone_jlabel, javax.swing.GroupLayout.DEFAULT_SIZE,
260, Short.MAX_VALUE)
        .addComponent(UserEmail_jlabel, javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE))))

    .addGroup(jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING, false)
        .addGroup(javax.swing.GroupLayout.Alignment.TRAILING,
jPanel1Layout.createSequentialGroup())
            .addComponent(Subscription_jlabel)
            .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED,
javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE)
            .addComponent(UserSubscription_jlabel,
javax.swing.GroupLayout.PREFERRED_SIZE, 260, javax.swing.GroupLayout.PREFERRED_SIZE))
            .addGroup(jPanel1Layout.createSequentialGroup())
                .addComponent>Password_jlabel)
                .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED,
javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE)
                .addComponent(UserPassword_jlabel,
javax.swing.GroupLayout.PREFERRED_SIZE, 260, javax.swing.GroupLayout.PREFERRED_SIZE))
            .addGroup(jPanel1Layout.createSequentialGroup())
                .addComponent(Username_jlabel)
                .addGap(93, 93, 93)
                .addComponent(UserUsername_jlabel,
javax.swing.GroupLayout.PREFERRED_SIZE, 260, javax.swing.GroupLayout.PREFERRED_SIZE)))
        .addComponent(ChangeSubscription_Button,
javax.swing.GroupLayout.Alignment.LEADING))
        .addGap(0, 166, Short.MAX_VALUE))
    .addGroup(jPanel1Layout.createSequentialGroup())
        .addComponent(Title)
        .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED,
javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE)
        .addComponent(Qualifications_button)
        .addGap(52, 52, 52)))
    .addGroup(jPanel1Layout.createSequentialGroup())
        .addGroup(jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)

```

```

        .addComponent(Close_button)
        .addComponent(Back_Button))
        .addGap(0, 0, Short.MAX_VALUE))
    );
    JPanel1Layout.setVerticalGroup(
        JPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
        .addGroup(javax.swing.GroupLayout.Alignment.TRAILING,
jPanel1Layout.createSequentialGroup()
        .addComponent(Back_Button)
        .addGap(4, 4, 4)
        .addGroup(jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
        .addComponent(Title)
        .addComponent(Qualifications_button))
        .addGap(11, 11, 11)
        .addGroup(jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
        .addComponent(Username_jlabel)
        .addComponent(UserUsername_jlabel, javax.swing.GroupLayout.PREFERRED_SIZE, 13,
javax.swing.GroupLayout.PREFERRED_SIZE))
        .addGap(16, 16, 16)
        .addGroup(jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASELINE)
        .addComponent>Password_jlabel)
        .addComponent(UserPassword_jlabel, javax.swing.GroupLayout.PREFERRED_SIZE, 15,
javax.swing.GroupLayout.PREFERRED_SIZE))
        .addGap(18, 18, 18)
        .addGroup(jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
        .addComponent(UserSubscription_jlabel, javax.swing.GroupLayout.PREFERRED_SIZE, 16,
javax.swing.GroupLayout.PREFERRED_SIZE)
        .addComponent(Subscription_jlabel))
        .addGap(21, 21, 21)
        .addComponent(ChangeSubscription_Button)
        .addGap(18, 18, 18)
        .addGroup(jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASELINE)
        .addComponent(Name_jlabel)
        .addComponent(UserName_jlabel))
        .addGap(16, 16, 16)
        .addGroup(jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
        .addComponent(Birthdate_jlabel)
        .addComponent(UserBirthdate_jlabel, javax.swing.GroupLayout.PREFERRED_SIZE, 13,
javax.swing.GroupLayout.PREFERRED_SIZE))
        .addGap(16, 16, 16)
        .addGroup(jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
        .addComponent(Address_jlabel)
        .addComponent(UserAddress_jlabel, javax.swing.GroupLayout.PREFERRED_SIZE, 13,
javax.swing.GroupLayout.PREFERRED_SIZE))
        .addGap(16, 16, 16)
        .addGroup(jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
        .addComponent(Phone_jlabel)
        .addComponent(UserPhone_jlabel, javax.swing.GroupLayout.PREFERRED_SIZE, 13,
javax.swing.GroupLayout.PREFERRED_SIZE))

```

```

        .addGap(18, 18, 18)
        .addGroup(jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
            .addComponent(UserEmail_jlabel, javax.swing.GroupLayout.PREFERRED_SIZE, 15,
javax.swing.GroupLayout.PREFERRED_SIZE)
            .addComponent(Email_jlabel))
        .addGap(35, 35, 35)
        .addComponent(Close_button))
    );

    javax.swing.GroupLayout layout = new javax.swing.GroupLayout(getContentPane());
    getContentPane().setLayout(layout);
    layout.setHorizontalGroup(
        layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
            .addGap(0, 626, Short.MAX_VALUE)
            .addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
                .addGroup(layout.createSequentialGroup()
                    .addGap(0, 0, Short.MAX_VALUE)
                    .addComponent(jPanel1, javax.swing.GroupLayout.PREFERRED_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE, javax.swing.GroupLayout.PREFERRED_SIZE)
                    .addGap(0, 0, Short.MAX_VALUE)))
            );
    layout.setVerticalGroup(
        layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
            .addGap(0, 408, Short.MAX_VALUE)
            .addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
                .addGroup(layout.createSequentialGroup()
                    .addGap(0, 0, Short.MAX_VALUE)
                    .addComponent(jPanel1, javax.swing.GroupLayout.PREFERRED_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE, javax.swing.GroupLayout.PREFERRED_SIZE)
                    .addGap(0, 0, Short.MAX_VALUE)))
            );

    pack();
} // </editor-fold> // GEN-END: initComponents

private void Close_buttonActionPerformed(java.awt.event.ActionEvent evt)
{ // GEN-FIRST: event_Close_buttonActionPerformed
    ScriptRunner sr = new ScriptRunner(con);
    try {
        sr.runScript(new BufferedReader(new
FileReader("C:\\Users\\Edmond\\Documents\\Workspace\\cps510\\jobbankuimaven\\scripts\\DropTables.s
ql")));
    } catch (FileNotFoundException e) {
        e.printStackTrace();
    }
    System.exit(0);
} // GEN-LAST: event_Close_buttonActionPerformed

```

```

        private void ChangeSubscription_ButtonActionPerformed(java.awt.event.ActionEvent evt)
        {
            //GEN-FIRST:event_ChangeSubscription_ButtonActionPerformed
            // TODO add your handling code here:
            //GEN-LAST:event_ChangeSubscription_ButtonActionPerformed

        }

        private void Qualifications_buttonActionPerformed(java.awt.event.ActionEvent evt)
        {
            //GEN-FIRST:event_Qualifications_buttonActionPerformed
            // TODO add your handling code here:
            //GEN-LAST:event_Qualifications_buttonActionPerformed

        }

        private void Back_ButtonActionPerformed(java.awt.event.ActionEvent evt)
        {
            //GEN-FIRST:event_Back_ButtonActionPerformed
            accountpage.setVisible(false);
            frontpage.setVisible(true);
            //GEN-LAST:event_Back_ButtonActionPerformed

        }

        // Variables declaration - do not modify//GEN-BEGIN:variables
        private javax.swing.JLabel Address_jlabel;
        private javax.swing.JButton Back_Button;
        private javax.swing.JLabel Birthdate_jlabel;
        private javax.swing.JButton ChangeSubscription_Button;
        private javax.swing.JButton Close_button;
        private javax.swing.JLabel Email_jlabel;
        private javax.swing.JLabel Name_jlabel;
        private javax.swing.JLabel Password_jlabel;
        private javax.swing.JLabel Phone_jlabel;
        private javax.swing.JButton Qualifications_button;
        private javax.swing.JLabel Subscription_jlabel;
        private javax.swing.JLabel Title;
        private javax.swing.JLabel UserAddress_jlabel;
        private javax.swing.JLabel UserBirthdate_jlabel;
        private javax.swing.JLabel UserEmail_jlabel;
        private javax.swing.JLabel UserName_jlabel;
        private javax.swing.JLabel UserPassword_jlabel;
        private javax.swing.JLabel UserPhone_jlabel;
        private javax.swing.JLabel UserSubscription_jlabel;
        private javax.swing.JLabel UserUsername_jlabel;
        private javax.swing.JLabel Username_jlabel;
        private javax.swing.JPanel jPanel1;
        // End of variables declaration//GEN-END:variables
    }

```

## FrontPage.java

```

/*
 * To change this license header, choose License Headers in Project Properties.
 * To change this template file, choose Tools | Templates

```



```

* and open the template in the editor.
*/
package com.edmond.jobbank.mavenproject1;

import java.io.*;
import java.sql.*;
import org.apache.ibatis.jdbc.ScriptRunner;

/**
 *
 * @author Brian
 */
public class FrontPage extends javax.swing.JFrame {

    private static FrontPage frontpage;
    private static Intro introPanel;
    static Connection con;
    private static String[] account;
    /**
     * Creates new form FrontPage
     */
    private FrontPage() {
        initComponents();
    }

    public static FrontPage getInstance(Intro iPanel, Connection conn, String[] accountS) {
        if(frontpage == null) {
            frontpage = new FrontPage();
        }
        con = conn;
        introPanel = iPanel;
        account = accountS;
        return frontpage;
    }

    /**
     * This method is called from within the constructor to initialize the form.
     * WARNING: Do NOT modify this code. The content of this method is always
     * regenerated by the Form Editor.
     */
    @SuppressWarnings("unchecked")
    // <editor-fold defaultstate="collapsed" desc="Generated Code">//GEN-BEGIN:initComponents
    private void initComponents() {

        FrontPage_jpanel = new javax.swing.JPanel();
        Jobs_Button = new javax.swing.JButton();
        Close_Button = new javax.swing.JButton();
        Account_Button = new javax.swing.JButton();
        Logout_Button = new javax.swing.JButton();

```

```

Introduction_Scroll = new javax.swing.JScrollPane();
Introduction_text = new javax.swing.JTextArea();
Welcome_Label = new javax.swing.JLabel();
Name = new javax.swing.JLabel();

setDefaultCloseOperation(javax.swing.WindowConstants.EXIT_ON_CLOSE);

Jobs_Button.setText("Jobs");
Jobs_Button.addActionListener(new java.awt.event.ActionListener() {
    public void actionPerformed(java.awt.event.ActionEvent evt) {
        Jobs_ButtonActionPerformed(evt);
    }
});

Close_Button.setText("Close");
Close_Button.addActionListener(new java.awt.event.ActionListener() {
    public void actionPerformed(java.awt.event.ActionEvent evt) {
        Close_ButtonActionPerformed(evt);
    }
});

Account_Button.setText("My Account");
Account_Button.addActionListener(new java.awt.event.ActionListener() {
    public void actionPerformed(java.awt.event.ActionEvent evt) {
        Account_ButtonActionPerformed(evt);
    }
});

Logout_Button.setText("Logout");
Logout_Button.addActionListener(new java.awt.event.ActionListener() {
    public void actionPerformed(java.awt.event.ActionEvent evt) {
        Logout_ButtonActionPerformed(evt);
    }
});

Introduction_text.setColumns(20);
Introduction_text.setRows(5);
Introduction_text.setText("Welcome!\n\nAt the Job Bank you will be able to find thousands of
opportunities\n\nin which you will be able to choose from. No matter what, the Job\nBank will always have a
option for you. From start ups to Large\nincorporations, all the jobs can be found here.\n\nUse the My
Accounts button to go to you account and see your\npersonal information and edit your settings. You can
also change\nyour subscription status here to unlock more features.\n\nUse the Jobs button to go and
see our large and diverse Job Bank.\nHere you will be able to find all sorts of jobs and the one just\nright
for you.\n\nGood Luck!");
Introduction_Scroll.setViewportViewView(Introduction_text);

Welcome_Label.setFont(new java.awt.Font("Tahoma", 0, 24)); // NOI18N
Welcome_Label.setText("Welcome");

```

```

Name.setFont(new java.awt.Font("Tahoma", 0, 24)); // NOI18N
Name.setText("      ");

    javax.swing.GroupLayout FrontPage_jpanelLayout = new
javax.swing.GroupLayout(FrontPage_jpanel);
    FrontPage_jpanel.setLayout(FrontPage_jpanelLayout);
    FrontPage_jpanelLayout.setHorizontalGroup(
        FrontPage_jpanelLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
            .addGroup(javax.swing.GroupLayout.Alignment.TRAILING,
FrontPage_jpanelLayout.createSequentialGroup()
                .addComponent(Close_Button)
                .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED,
javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE)
                .addComponent(Login_Button, javax.swing.GroupLayout.PREFERRED_SIZE, 89,
javax.swing.GroupLayout.PREFERRED_SIZE))
            .addGroup(FrontPage_jpanelLayout.createSequentialGroup()
                .addComponent(Account_Button)
                .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)
                .addComponent(Logout_Button, javax.swing.GroupLayout.PREFERRED_SIZE, 89,
javax.swing.GroupLayout.PREFERRED_SIZE)
                .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)
                .addGroup(FrontPage_jpanelLayout.createSequentialGroup()
                    .addComponent(Welcome_Label)
                    .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)
                    .addComponent(Name))
                .addGroup(FrontPage_jpanelLayout.createSequentialGroup()
                    .addComponent(Introduction_Scroll, javax.swing.GroupLayout.PREFERRED_SIZE, 549,
javax.swing.GroupLayout.PREFERRED_SIZE))
                .addGap(19, 19, 19))
    );
    FrontPage_jpanelLayout.setVerticalGroup(
        FrontPage_jpanelLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
            .addGroup(FrontPage_jpanelLayout.createSequentialGroup()
                .addGap(34, 34, 34)
                .addGroup(FrontPage_jpanelLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASELINE)
                    .addComponent(Name)
                    .addComponent(Welcome_Label))
                .addGap(18, 18, 18)
                .addGroup(FrontPage_jpanelLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASELINE)
                    .addComponent(Name)
                    .addComponent(Welcome_Label))
                .addGap(18, 18, 18)
    );

```

```

        .addComponent(Account_Button)
        .addComponent(Jobs_Button))
        .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.UNRELATED)
        .addComponent(Introduction_Scroll, javax.swing.GroupLayout.PREFERRED_SIZE, 314,
javax.swing.GroupLayout.PREFERRED_SIZE)
        .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED, 29,
Short.MAX_VALUE)

.addGroup(FrontPage_jpanelLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASELINE
)

        .addComponent(Login_Button)
        .addComponent(Close_Button)))
);

javax.swing.GroupLayout layout = new javax.swing.GroupLayout(getContentPane());
getContentPane().setLayout(layout);
layout.setHorizontalGroup(
    layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
        .addGap(0, 586, Short.MAX_VALUE)
        .addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
            .addGroup(layout.createSequentialGroup()
                .addGap(0, 0, Short.MAX_VALUE)
                .addComponent(FrontPage_jpanel, javax.swing.GroupLayout.PREFERRED_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE, javax.swing.GroupLayout.PREFERRED_SIZE)
                .addGap(0, 0, Short.MAX_VALUE)))
        );
layout.setVerticalGroup(
    layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
        .addGap(0, 481, Short.MAX_VALUE)
        .addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
            .addGroup(layout.createSequentialGroup()
                .addGap(0, 0, Short.MAX_VALUE)
                .addComponent(FrontPage_jpanel, javax.swing.GroupLayout.PREFERRED_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE, javax.swing.GroupLayout.PREFERRED_SIZE)
                .addGap(0, 0, Short.MAX_VALUE)))
        );

pack();
} // </editor-fold> // GEN-END: initComponents

private void Jobs_ButtonActionPerformed(java.awt.event.ActionEvent evt)
{ // GEN-FIRST: event_Jobs_ButtonActionPerformed
    Jobs jobsPanel = Jobs.getInstance(this, con);
    jobsPanel.setVisible(true);
    this.setVisible(false);
} // GEN-LAST: event_Jobs_ButtonActionPerformed

private void Close_ButtonActionPerformed(java.awt.event.ActionEvent evt)
{ // GEN-FIRST: event_Close_ButtonActionPerformed

```

```

        ScriptRunner sr = new ScriptRunner(con);
        try {
            sr.runScript(new BufferedReader(new
FileReader("C:\\Users\\Edmond\\Documents\\Workspace\\cps510\\jobbankuimaven\\scripts\\DropTables.s
ql")));
        } catch (FileNotFoundException e) {
            e.printStackTrace();
        }
        System.exit(0);
    } //GEN-LAST:event_Close_ButtonActionPerformed

    private void Account_ButtonActionPerformed(java.awt.event.ActionEvent evt)
    { //GEN-FIRST:event_Account_ButtonActionPerformed
        Account accountPanel = Account.getInstance(this, con, account);
        accountPanel.setVisible(true);
        accountPanel.updateText();
        this.setVisible(false);
    } //GEN-LAST:event_Account_ButtonActionPerformed

    private void Logout_ButtonActionPerformed(java.awt.event.ActionEvent evt)
    { //GEN-FIRST:event_Logout_ButtonActionPerformed
        frontpage.setVisible(false);
        introPanel.setVisible(true);
    } //GEN-LAST:event_Logout_ButtonActionPerformed

    // Variables declaration - do not modify//GEN-BEGIN:variables
    private javax.swing.JButton Account_Button;
    private javax.swing.JButton Close_Button;
    private javax.swing.JPanel FrontPage_jpanel;
    private javax.swing.JScrollPane Introduction_Scroll;
    private javax.swing.JTextArea Introduction_text;
    private javax.swing.JButton Jobs_Button;
    private javax.swing.JButton Logout_Button;
    private javax.swing.JLabel Name;
    private javax.swing.JLabel Welcome_Label;
    // End of variables declaration//GEN-END:variables
}

```

## Intro.java

```

/*
 * To change this license header, choose License Headers in Project Properties.
 * To change this template file, choose Tools | Templates
 * and open the template in the editor.
 */
package com.edmond.jobbank.mavenproject1;

import java.io.*;

```

```

import java.sql.Connection;
import org.apache.ibatis.jdbc.ScriptRunner;

/**
 *
 * @author Brian
 */
public class Intro extends javax.swing.JFrame {

    private static Intro introPanel;
    static Connection conn = Database.getConnectionToDB();
    static ScriptRunner sr = new ScriptRunner(conn);
    /**
     * Creates new form Intro
     */
    private Intro() {
        initComponents();
    }

    public static Intro getInstance(){
        if(introPanel == null) {
            introPanel = new Intro();
            createTables();
            populateTables();
        }
        return introPanel;
    }

    private static void createTables() {
        try {
            sr.runScript(new BufferedReader(new
FileReader("C:\\Users\\Edmond\\Documents\\Workspace\\cps510\\jobbankuimaven\\scripts\\CreateTable
s.sql")));
        } catch (FileNotFoundException e) {
            e.printStackTrace();
        }
    }

    private static void dropTables() {
        try {
            sr.runScript(new BufferedReader(new
FileReader("C:\\Users\\Edmond\\Documents\\Workspace\\cps510\\jobbankuimaven\\scripts\\DropTables.s
ql")));
        } catch (FileNotFoundException e) {
            e.printStackTrace();
        }
    }

    private static void populateTables() {

```

```

    try {
        sr.runScript(new BufferedReader(new
FileReader("C:\\Users\\Edmond\\Documents\\Workspace\\cps510\\jobbankuimaven\\scripts\\PopulateTabl
es.sql"))));
    } catch (FileNotFoundException e) {
        e.printStackTrace();
    }
}

/**
 * This method is called from within the constructor to initialize the form.
 * WARNING: Do NOT modify this code. The content of this method is always
 * regenerated by the Form Editor.
 */
@SuppressWarnings("unchecked")
// <editor-fold defaultstate="collapsed" desc="Generated Code"> //GEN-BEGIN: initComponents
private void initComponents() {

    Intro_jpanel = new javax.swing.JPanel();
    NewUser_Button = new javax.swing.JButton();
    ReturningUser_Button = new javax.swing.JButton();
    Close_Button = new javax.swing.JButton();
    Title = new javax.swing.JLabel();

    setDefaultCloseOperation(javax.swing.WindowConstants.EXIT_ON_CLOSE);

    NewUser_Button.setText("New User");
    NewUser_Button.addActionListener(new java.awt.event.ActionListener() {
        public void actionPerformed(java.awt.event.ActionEvent evt) {
            NewUser_ButtonActionPerformed(evt);
        }
    });

    ReturningUser_Button.setText("Returning User");
    ReturningUser_Button.addActionListener(new java.awt.event.ActionListener() {
        public void actionPerformed(java.awt.event.ActionEvent evt) {
            ReturningUser_ButtonActionPerformed(evt);
        }
    });

    Close_Button.setText("Close");
    Close_Button.addActionListener(new java.awt.event.ActionListener() {
        public void actionPerformed(java.awt.event.ActionEvent evt) {
            Close_ButtonActionPerformed(evt);
        }
    });

    Title.setFont(new java.awt.Font("Calibri", 0, 24)); // NOI18N
    Title.setHorizontalAlignment(javax.swing.SwingConstants.CENTER);

```





```

        layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
        .addGap(0, 300, Short.MAX_VALUE)
        .addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
        .addComponent(Intro_jpanel, javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE))
        );

    pack();
} // </editor-fold> // GEN-END: initComponents

    private void NewUser_ButtonActionPerformed(java.awt.event.ActionEvent evt)
{ // GEN-FIRST: event_NewUser_ButtonActionPerformed
    newUserPanel newPanel = newUserPanel.getInstance(this, conn);
    newPanel.setVisible(true);
    this.setVisible(false);
} // GEN-LAST: event_NewUser_ButtonActionPerformed

    private void ReturningUser_ButtonActionPerformed(java.awt.event.ActionEvent evt)
{ // GEN-FIRST: event_ReturningUser_ButtonActionPerformed
    Returning returnUserPanel = Returning.getInstance(this, conn);
    returnUserPanel.setVisible(true);
    this.setVisible(false);
} // GEN-LAST: event_ReturningUser_ButtonActionPerformed

    private void Close_ButtonActionPerformed(java.awt.event.ActionEvent evt)
{ // GEN-FIRST: event_Close_ButtonActionPerformed
    dropTables();
    System.exit(0);
} // GEN-LAST: event_Close_ButtonActionPerformed

    // Variables declaration - do not modify // GEN-BEGIN: variables
    private javax.swing.JButton Close_Button;
    private javax.swing.JPanel Intro_jpanel;
    private javax.swing.JButton NewUser_Button;
    private javax.swing.JButton ReturningUser_Button;
    private javax.swing.JLabel Title;
    // End of variables declaration // GEN-END: variables
}

```

## Returning.java

```

/*
 * To change this license header, choose License Headers in Project Properties.
 * To change this template file, choose Tools | Templates
 * and open the template in the editor.
 */
package com.edmond.jobbank.mavenproject1;

```

```

import java.io.*;
import java.sql.*;
import org.apache.ibatis.jdbc.ScriptRunner;
/**
 *
 * @author Brian
 */
public class Returning extends javax.swing.JFrame {

    private static Returning returnPanel;
    private static Intro introPanel;
    static Connection con;

    public static Returning getInstance(Intro iPanel, Connection conn) {
        if(returnPanel == null){
            returnPanel = new Returning();
            introPanel = iPanel;
            con = conn;
        }
        return returnPanel;
    }

    /**
     * Creates new form Returning
     */
    private Returning() {
        initComponents();
    }

    private boolean fillAccountInfo(String[] aString, String username, String password) {
        try {
            String query = "SELECT * FROM Accounts WHERE account_name = '"+username+"' AND account_password= '"+password+"'";
            int i = 0 ;
            Statement st = con.createStatement();
            st.executeQuery(query);
            ResultSet rs = st.executeQuery(query);

            while (rs.next()) {
                for(int j = 0 ; j < aString.length; j++) {
                    aString[j] = rs.getString(j+1);
                }
            }

            st.close();

        } catch (SQLException e) {
            e.printStackTrace();
        }
    }

```

```

    if (aString[0] == null) {
        return false;
    } else
        return true;
}

/**
 * This method is called from within the constructor to initialize the form.
 * WARNING: Do NOT modify this code. The content of this method is always
 * regenerated by the Form Editor.
 */
@SuppressWarnings("unchecked")
// <editor-fold defaultstate="collapsed" desc="Generated Code"> //GEN-BEGIN: initComponents
private void initComponents() {

    Returning_jpanel = new javax.swing.JPanel();
    Title = new javax.swing.JLabel();
    Username_jlabel = new javax.swing.JLabel();
    Password_jlabel = new javax.swing.JLabel();
    Username_text = new javax.swing.JTextField();
    Password_text = new javax.swing.JTextField();
    Login_Button = new javax.swing.JButton();
    Close_Button = new javax.swing.JButton();
    Back_Button = new javax.swing.JButton();

    setDefaultCloseOperation(javax.swing.WindowConstants.EXIT_ON_CLOSE);

    Title.setFont(new java.awt.Font("Tahoma", 0, 36)); // NOI18N
    Title.setText("Welcome Back");

    Username_jlabel.setFont(new java.awt.Font("Tahoma", 0, 12)); // NOI18N
    Username_jlabel.setText("Username");

    Password_jlabel.setFont(new java.awt.Font("Tahoma", 0, 12)); // NOI18N
    Password_jlabel.setText("Password");

    Username_text.addActionListener(new java.awt.event.ActionListener() {
        public void actionPerformed(java.awt.event.ActionEvent evt) {
            Username_textActionPerformed(evt);
        }
    });

    Password_text.addActionListener(new java.awt.event.ActionListener() {
        public void actionPerformed(java.awt.event.ActionEvent evt) {
            Password_textActionPerformed(evt);
        }
    });
}

```

```

Login_Button.setText("Login");
Login_Button.addActionListener(new java.awt.event.ActionListener() {
    public void actionPerformed(java.awt.event.ActionEvent evt) {
        Login_ButtonActionPerformed(evt);
    }
});

Close_Button.setText("Close");
Close_Button.addActionListener(new java.awt.event.ActionListener() {
    public void actionPerformed(java.awt.event.ActionEvent evt) {
        Close_ButtonActionPerformed(evt);
    }
});

Back_Button.setText("Back");
Back_Button.addActionListener(new java.awt.event.ActionListener() {
    public void actionPerformed(java.awt.event.ActionEvent evt) {
        Back_ButtonActionPerformed(evt);
    }
});

javax.swing.GroupLayout Returning_jpanelLayout = new
javax.swing.GroupLayout(Returning_jpanel);
Returning_jpanel.setLayout(Returning_jpanelLayout);
Returning_jpanelLayout.setHorizontalGroup(
    Returning_jpanelLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
        .addGroup(javax.swing.GroupLayout.Alignment.TRAILING,
Returning_jpanelLayout.createSequentialGroup()
        .addContainerGap(86, Short.MAX_VALUE)

.addGroup(Returning_jpanelLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
        .addGroup(Returning_jpanelLayout.createSequentialGroup()

.addGroup(Returning_jpanelLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.TRAILING)
        .addComponent(Username_jlabel)
        .addComponent>Password_jlabel))
        .addGroup(Returning_jpanelLayout.createSequentialGroup()

.addGroup(Returning_jpanelLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING,
false)
        .addComponent>Password_text)
        .addComponent(Username_text, javax.swing.GroupLayout.PREFERRED_SIZE, 115,
javax.swing.GroupLayout.PREFERRED_SIZE)))
        .addGroup(Returning_jpanelLayout.createSequentialGroup()
        .addGap(100, 100, 100)
        .addComponent(Login_Button))
        .addComponent>Title))
        .addGroup(Returning_jpanelLayout.createSequentialGroup()
        .addGap(83, 83, 83))
        .addGroup(Returning_jpanelLayout.createSequentialGroup()

```

```

.addGroup(Returning_jpanelLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.TRAILING)
    .addComponent(Close_Button)
    .addComponent(Back_Button))
.addGap(0, 0, Short.MAX_VALUE))
);
Returning_jpanelLayout.setVerticalGroup(
    Returning_jpanelLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
    .addGroup(Returning_jpanelLayout.createSequentialGroup())
    .addComponent(Back_Button)
    .addGap(8, 8, 8)
    .addComponent(Title)
    .addGap(43, 43, 43)

.addGroup(Returning_jpanelLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.TRAILING)
    .addComponent(Username_jlabel)
    .addComponent(Username_text, javax.swing.GroupLayout.PREFERRED_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE, javax.swing.GroupLayout.PREFERRED_SIZE))
.addGap(26, 26, 26)

.addGroup(Returning_jpanelLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.TRAILING)
    .addComponent(Password_text, javax.swing.GroupLayout.PREFERRED_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE, javax.swing.GroupLayout.PREFERRED_SIZE)
    .addComponent(Password_jlabel))
.addGap(27, 27, 27)
.addComponent(Login_Button)
.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED, 43,
Short.MAX_VALUE)
.addComponent(Close_Button))
);

javax.swing.GroupLayout layout = new javax.swing.GroupLayout(getContentPane());
getContentPane().setLayout(layout);
layout.setHorizontalGroup(
    layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
    .addGap(0, 400, Short.MAX_VALUE)
    .addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
        .addGroup(layout.createSequentialGroup())
        .addGap(0, 0, Short.MAX_VALUE)
        .addComponent(Returning_jpanel, javax.swing.GroupLayout.PREFERRED_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE, javax.swing.GroupLayout.PREFERRED_SIZE)
        .addGap(0, 0, Short.MAX_VALUE)))
);
layout.setVerticalGroup(
    layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
    .addGap(0, 300, Short.MAX_VALUE)
    .addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
        .addGroup(layout.createSequentialGroup())
        .addGap(0, 0, Short.MAX_VALUE)

```

```

        .addComponent(Returning_jpanel, javax.swing.GroupLayout.PREFERRED_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE, javax.swing.GroupLayout.PREFERRED_SIZE)
        .addGap(0, 0, Short.MAX_VALUE)))
    );

    pack();
} // </editor-fold> // GEN-END: initComponents

private void Password_textActionPerformed(java.awt.event.ActionEvent evt)
{ // GEN-FIRST:event_Password_textActionPerformed
    // TODO add your handling code here:
} // GEN-LAST:event_Password_textActionPerformed

private void Login_ButtonActionPerformed(java.awt.event.ActionEvent evt)
{ // GEN-FIRST:event_Login_ButtonActionPerformed
    String username, password;
    String[] account = new String[8];
    username = Username_text.getText();
    password = Password_text.getText();
    boolean isfilled = fillAccountInfo(account, username, password);
    if (isfilled) {
        FrontPage fp = FrontPage.getInstance(introPanel, con, account);
        fp.setVisible(true);
        this.setVisible(false);
    }
} // GEN-LAST:event_Login_ButtonActionPerformed

private void Back_ButtonActionPerformed(java.awt.event.ActionEvent evt)
{ // GEN-FIRST:event_Back_ButtonActionPerformed
    introPanel.setVisible(true);
    this.setVisible(false);
} // GEN-LAST:event_Back_ButtonActionPerformed

private void Username_textActionPerformed(java.awt.event.ActionEvent evt)
{ // GEN-FIRST:event_Username_textActionPerformed
    // TODO add your handling code here:
} // GEN-LAST:event_Username_textActionPerformed

private void Close_ButtonActionPerformed(java.awt.event.ActionEvent evt)
{ // GEN-FIRST:event_Close_ButtonActionPerformed
    ScriptRunner sr = new ScriptRunner(con);
    try {
        sr.runScript(new BufferedReader(new
FileReader("C:\\Users\\Edmond\\Documents\\Workspace\\cps510\\jobbankuimaven\\scripts\\DropTables.s
ql")));
    } catch (FileNotFoundException e) {
        e.printStackTrace();
    }
    System.exit(0);
}

```

```

} //GEN-LAST:event_Close_ButtonActionPerformed

// Variables declaration - do not modify //GEN-BEGIN:variables
private javax.swing.JButton Back_Button;
private javax.swing.JButton Close_Button;
private javax.swing.JButton Login_Button;
private javax.swing.JLabel Password_jlabel;
private javax.swing.JTextField Password_text;
private javax.swing.JPanel Returning_jpanel;
private javax.swing.JLabel Title;
private javax.swing.JLabel Username_jlabel;
private javax.swing.JTextField Username_text;
// End of variables declaration //GEN-END:variables
}

```

## NewUserPanel.java

```

/*
 * To change this license header, choose License Headers in Project Properties.
 * To change this template file, choose Tools | Templates
 * and open the template in the editor.
 */
package com.edmond.jobbank.mavenproject1;

import java.io.*;
import java.sql.*;
import org.apache.ibatis.jdbc.ScriptRunner;

/**
 *
 * @author Brian
 */
public class newUserPanel extends javax.swing.JFrame {
    private final Intro introPanel;
    private static newUserPanel instance ;
    static Connection con;

    /**
     * Creates new form New
     * @param panel
     */
    private newUserPanel(Intro panel) {
        initComponents();
        introPanel = panel;
    }
}

```

```

public static newUserPanel getInstance(Intro panel, Connection conn){
    if(instance == null){
        instance = new newUserPanel(panel);
        con = conn;
    }
    return instance;
}

private String getAccountId() {
    //find the row size and label the accountId to be 1 more of the last id
    int count=0;
    try {
        String query = "SELECT COUNT(*) AS rowcount FROM Accounts";
        Statement st = con.createStatement();
        ResultSet rs = st.executeQuery(query);

        rs.next();
        count = rs.getInt("rowcount") ;
        count++;
        rs.close();
    } catch (SQLException e) {
        e.printStackTrace();
    }
    return "Job#" + count;
}

private void populateNewAccount(String account_id, String account_name, String account_password,
String user_name,
        String user_email, String user_birthdate, String user_address, String user_phone) {
    String query = "insert into Accounts ( account_id, account_name, account_password, user_name,
user_email, user_birthdate, user_address, user_phone) values
('"+account_id+"','"+account_name+"','"+account_password+"', 'a', 'a', 'a', 'a', 'a')";
    try {
        Statement st = con.createStatement();
        st.executeQuery(query);
        System.out.println("insert into Accounts ( account_id, account_name, account_password,
user_name, user_email, user_birthdate, user_address, user_phone) values
('"+account_id+"','"+account_name+"','"+account_password+"', '"+user_name+"', '"+user_email+"',
 '"+user_birthdate+"', '"+user_address+"', '"+user_phone+"')");

        st.close();
    } catch (SQLException e) {
        e.printStackTrace();
    }
}

/**
 * This method is called from within the constructor to initialize the form.
 * WARNING: Do NOT modify this code. The content of this method is always

```



```

* regenerated by the Form Editor.
*/
@SuppressWarnings("unchecked")
// <editor-fold defaultstate="collapsed" desc="Generated Code"> //GEN-BEGIN: initComponents
private void initComponents() {

    New_jpanel = new javax.swing.JPanel();
    Username_jlabel = new javax.swing.JLabel();
    Title = new javax.swing.JLabel();
    Username_text = new javax.swing.JTextField();
    Close_jlabel = new javax.swing.JButton();
    Password_text = new javax.swing.JTextField();
    Password_jlabel = new javax.swing.JLabel();
    ReType_text = new javax.swing.JTextField();
    ReType_jlabel = new javax.swing.JLabel();
    Name_text = new javax.swing.JTextField();
    Name_jlabel = new javax.swing.JLabel();
    Birthdate_text = new javax.swing.JTextField();
    Birthdate_jlabel = new javax.swing.JLabel();
    Address_text = new javax.swing.JTextField();
    Address_jlabel = new javax.swing.JLabel();
    Phone_text = new javax.swing.JTextField();
    Phone_jlabel = new javax.swing.JLabel();
    Email_text = new javax.swing.JTextField();
    Email_jlabel = new javax.swing.JLabel();
    Back_Button = new javax.swing.JButton();
    Create_Button = new javax.swing.JButton();

    setDefaultCloseOperation(javax.swing.WindowConstants.EXIT_ON_CLOSE);

    Username_jlabel.setFont(new java.awt.Font("Tahoma", 0, 12)); // NOI18N
    Username_jlabel.setText("Username");

    Title.setFont(new java.awt.Font("Tahoma", 0, 24)); // NOI18N
    Title.setText("Account Details");

    Username_text.addActionListener(new java.awt.event.ActionListener() {
        public void actionPerformed(java.awt.event.ActionEvent evt) {
            Username_textActionPerformed(evt);
        }
    });

    Close_jlabel.setText("Close");
    Close_jlabel.addActionListener(new java.awt.event.ActionListener() {
        public void actionPerformed(java.awt.event.ActionEvent evt) {
            Close_jlabelActionPerformed(evt);
        }
    });
}

```

```

Password_text.addActionListener(new java.awt.event.ActionListener() {
    public void actionPerformed(java.awt.event.ActionEvent evt) {
        Password_textActionPerformed(evt);
    }
});

Password_jlabel.setFont(new java.awt.Font("Tahoma", 0, 12)); // NOI18N
Password_jlabel.setText("Password");

ReType_text.addActionListener(new java.awt.event.ActionListener() {
    public void actionPerformed(java.awt.event.ActionEvent evt) {
        ReType_textActionPerformed(evt);
    }
});

ReType_jlabel.setFont(new java.awt.Font("Tahoma", 0, 12)); // NOI18N
ReType_jlabel.setText("Re-Type Password");

Name_text.addActionListener(new java.awt.event.ActionListener() {
    public void actionPerformed(java.awt.event.ActionEvent evt) {
        Name_textActionPerformed(evt);
    }
});

Name_jlabel.setFont(new java.awt.Font("Tahoma", 0, 12)); // NOI18N
Name_jlabel.setText("Name");

Birthdate_text.addActionListener(new java.awt.event.ActionListener() {
    public void actionPerformed(java.awt.event.ActionEvent evt) {
        Birthdate_textActionPerformed(evt);
    }
});

Birthdate_jlabel.setFont(new java.awt.Font("Tahoma", 0, 12)); // NOI18N
Birthdate_jlabel.setText("Birthdate");

Address_text.addActionListener(new java.awt.event.ActionListener() {
    public void actionPerformed(java.awt.event.ActionEvent evt) {
        Address_textActionPerformed(evt);
    }
});

Address_jlabel.setFont(new java.awt.Font("Tahoma", 0, 12)); // NOI18N
Address_jlabel.setText("Address");

Phone_text.addActionListener(new java.awt.event.ActionListener() {
    public void actionPerformed(java.awt.event.ActionEvent evt) {
        Phone_textActionPerformed(evt);
    }
}

```



```

        .addGroup(javax.swing.GroupLayout.Alignment.TRAILING,
New_jpanelLayout.createSequentialGroup()
        .addComponent(Password_jlabel)
        .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED,
javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE)
        .addComponent(Password_text, javax.swing.GroupLayout.PREFERRED_SIZE,
229, javax.swing.GroupLayout.PREFERRED_SIZE))
        .addGroup(javax.swing.GroupLayout.Alignment.TRAILING,
New_jpanelLayout.createSequentialGroup()
        .addComponent(ReType_jlabel)
        .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED,
45, Short.MAX_VALUE)
        .addComponent(ReType_text, javax.swing.GroupLayout.PREFERRED_SIZE,
229, javax.swing.GroupLayout.PREFERRED_SIZE))
        .addGroup(New_jpanelLayout.createSequentialGroup()
        .addComponent(Name_jlabel)
        .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED,
javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE)
        .addComponent(Name_text, javax.swing.GroupLayout.PREFERRED_SIZE,
229, javax.swing.GroupLayout.PREFERRED_SIZE))
        .addGroup(New_jpanelLayout.createSequentialGroup()
        .addComponent(Birthdate_jlabel)
        .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED,
javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE)
        .addComponent(Birthdate_text, javax.swing.GroupLayout.PREFERRED_SIZE,
229, javax.swing.GroupLayout.PREFERRED_SIZE))
        .addGroup(javax.swing.GroupLayout.Alignment.TRAILING,
New_jpanelLayout.createSequentialGroup()
        .addComponent(Address_jlabel)
        .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED,
javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE)
        .addComponent(Address_text, javax.swing.GroupLayout.PREFERRED_SIZE,
229, javax.swing.GroupLayout.PREFERRED_SIZE))
        .addGroup(javax.swing.GroupLayout.Alignment.TRAILING,
New_jpanelLayout.createSequentialGroup()
        .addComponent(Phone_jlabel)
        .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED,
javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE)
        .addComponent(Phone_text, javax.swing.GroupLayout.PREFERRED_SIZE,
229, javax.swing.GroupLayout.PREFERRED_SIZE))
        .addGroup(javax.swing.GroupLayout.Alignment.TRAILING,
New_jpanelLayout.createSequentialGroup()
        .addComponent(Email_jlabel)
        .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED,
javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE)
        .addComponent(Email_text, javax.swing.GroupLayout.PREFERRED_SIZE, 229,
javax.swing.GroupLayout.PREFERRED_SIZE))))
        .addGroup(New_jpanelLayout.createSequentialGroup()
        .addComponent(Title)

```

```

        .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED, 245,
javax.swing.GroupLayout.PREFERRED_SIZE))))
        .addGroup(New_jpanelLayout.createSequentialGroup()
            .addComponent(Back_Button)
            .addGap(0, 0, Short.MAX_VALUE)))
        .addGap(61, 61, 61))
    .addGroup(New_jpanelLayout.createSequentialGroup()
        .addComponent(Close_jlabel)
        .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED,
javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE)
        .addComponent(Create_Button)
        .addGap(23, 23, 23))
    );
    New_jpanelLayout.setVerticalGroup(
        New_jpanelLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
        .addGroup(New_jpanelLayout.createSequentialGroup()
            .addGap(1, 1, 1)
            .addComponent(Back_Button)
            .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)
            .addComponent(Title)
            .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)

        .addGroup(New_jpanelLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.TRAILING)
            .addComponent(Username_jlabel)
            .addComponent(Username_text, javax.swing.GroupLayout.PREFERRED_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE, javax.swing.GroupLayout.PREFERRED_SIZE))
            .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.UNRELATED)

        .addGroup(New_jpanelLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.TRAILING)
            .addComponent>Password_jlabel)
            .addComponent>Password_text, javax.swing.GroupLayout.PREFERRED_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE, javax.swing.GroupLayout.PREFERRED_SIZE))
            .addGap(11, 11, 11)

        .addGroup(New_jpanelLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.TRAILING)
            .addComponent(ReType_jlabel)
            .addComponent(ReType_text, javax.swing.GroupLayout.PREFERRED_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE, javax.swing.GroupLayout.PREFERRED_SIZE))
            .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.UNRELATED)

        .addGroup(New_jpanelLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.TRAILING)
            .addComponent(Name_jlabel)
            .addComponent(Name_text, javax.swing.GroupLayout.PREFERRED_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE, javax.swing.GroupLayout.PREFERRED_SIZE))
            .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.UNRELATED)

        .addGroup(New_jpanelLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.TRAILING)
            .addComponent(Birthdate_jlabel)

```

```

        .addComponent(Birthdate_text, javax.swing.GroupLayout.PREFERRED_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE, javax.swing.GroupLayout.PREFERRED_SIZE))
        .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.UNRELATED)

    .addGroup(New_jpanelLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.TRAILING)
        .addComponent(Address_jlabel)
        .addComponent(Address_text, javax.swing.GroupLayout.PREFERRED_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE, javax.swing.GroupLayout.PREFERRED_SIZE))
        .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.UNRELATED)

    .addGroup(New_jpanelLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.TRAILING)
        .addComponent(Phone_jlabel)
        .addComponent(Phone_text, javax.swing.GroupLayout.PREFERRED_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE, javax.swing.GroupLayout.PREFERRED_SIZE))
        .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.UNRELATED)

    .addGroup(New_jpanelLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.TRAILING)
        .addComponent(Email_jlabel)
        .addComponent(Email_text, javax.swing.GroupLayout.PREFERRED_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE, javax.swing.GroupLayout.PREFERRED_SIZE))
        .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED, 11,
Short.MAX_VALUE)

    .addGroup(New_jpanelLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
        .addGroup(javax.swing.GroupLayout.Alignment.TRAILING,
New_jpanelLayout.createSequentialGroup()
            .addComponent(Create_Button)
            .addContainerGap())
        .addComponent(Close_jlabel, javax.swing.GroupLayout.Alignment.TRAILING)))
    );

    javax.swing.GroupLayout layout = new javax.swing.GroupLayout(getContentPane());
    getContentPane().setLayout(layout);
    layout.setHorizontalGroup(
        layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
            .addGap(0, 488, Short.MAX_VALUE)
            .addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
                .addGroup(layout.createSequentialGroup()
                    .addGap(0, 0, Short.MAX_VALUE)
                    .addComponent(New_jpanel, javax.swing.GroupLayout.PREFERRED_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE, javax.swing.GroupLayout.PREFERRED_SIZE)
                    .addGap(0, 0, Short.MAX_VALUE)))
            );
    layout.setVerticalGroup(
        layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
            .addGap(0, 347, Short.MAX_VALUE)
            .addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
                .addGroup(layout.createSequentialGroup()
                    .addGap(0, 0, Short.MAX_VALUE)

```

```

        .addComponent(New_jpanel, javax.swing.GroupLayout.PREFERRED_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE, javax.swing.GroupLayout.PREFERRED_SIZE)
        .addGap(0, 0, Short.MAX_VALUE)))
    );

    pack();
} // </editor-fold> // GEN-END: initComponents

private void Username_textActionPerformed(java.awt.event.ActionEvent evt)
{ // GEN-FIRST:event_Username_textActionPerformed
    // TODO add your handling code here:
} // GEN-LAST:event_Username_textActionPerformed

private void Close_jlabelActionPerformed(java.awt.event.ActionEvent evt)
{ // GEN-FIRST:event_Close_jlabelActionPerformed
    ScriptRunner sr = new ScriptRunner(con);
    try {
        sr.runScript(new BufferedReader(new
FileReader("C:\\Users\\Edmond\\Documents\\Workspace\\cps510\\jobbankuimaven\\scripts\\DropTables.s
ql")));
    } catch (FileNotFoundException e) {
        e.printStackTrace();
    }
    System.exit(0);
} // GEN-LAST:event_Close_jlabelActionPerformed

private void Password_textActionPerformed(java.awt.event.ActionEvent evt)
{ // GEN-FIRST:event_Password_textActionPerformed
    // TODO add your handling code here:
} // GEN-LAST:event_Password_textActionPerformed

private void ReType_textActionPerformed(java.awt.event.ActionEvent evt)
{ // GEN-FIRST:event_ReType_textActionPerformed
    // TODO add your handling code here:
} // GEN-LAST:event_ReType_textActionPerformed

private void Name_textActionPerformed(java.awt.event.ActionEvent evt)
{ // GEN-FIRST:event_Name_textActionPerformed
    // TODO add your handling code here:
} // GEN-LAST:event_Name_textActionPerformed

private void Birthdate_textActionPerformed(java.awt.event.ActionEvent evt)
{ // GEN-FIRST:event_Birthdate_textActionPerformed
    // TODO add your handling code here:
} // GEN-LAST:event_Birthdate_textActionPerformed

private void Address_textActionPerformed(java.awt.event.ActionEvent evt)
{ // GEN-FIRST:event_Address_textActionPerformed
    // TODO add your handling code here:
}

```

```

} //GEN-LAST:event_Address_textActionPerformed

private void Phone_textActionPerformed(java.awt.event.ActionEvent evt)
{ //GEN-FIRST:event_Phone_textActionPerformed
    // TODO add your handling code here:
} //GEN-LAST:event_Phone_textActionPerformed

private void Email_textActionPerformed(java.awt.event.ActionEvent evt)
{ //GEN-FIRST:event_Email_textActionPerformed
    // TODO add your handling code here:
} //GEN-LAST:event_Email_textActionPerformed

private void Back_ButtonActionPerformed(java.awt.event.ActionEvent evt)
{ //GEN-FIRST:event_Back_ButtonActionPerformed
    introPanel.setVisible(true);
    this.setVisible(false);
} //GEN-LAST:event_Back_ButtonActionPerformed

private void Create_ButtonActionPerformed(java.awt.event.ActionEvent evt)
{ //GEN-FIRST:event_Create_ButtonActionPerformed
    String[] account = new String[8];
    account[0] = this.getAccountId();
    account[1] = Username_text.getText();
    account[2] = Password_text.getText();
    account[3] = Name_text.getText();
    account[4] = Email_text.getText();
    account[5] = Birthdate_text.getText();
    account[6] = Address_text.getText();
    account[7] = Phone_text.getText();

    populateNewAccount(account[0],account[1],account[2],account[3],account[4],account[5],account[6],account[7]);

    FrontPage fp = FrontPage.getInstance(introPanel,con, account);
    fp.setVisible(true);
    this.setVisible(false);
} //GEN-LAST:event_Create_ButtonActionPerformed

// Variables declaration - do not modify //GEN-BEGIN:variables
private javax.swing.JLabel Address_jlabel;
private javax.swing.JTextField Address_text;
private javax.swing.JButton Back_Button;
private javax.swing.JLabel Birthdate_jlabel;
private javax.swing.JTextField Birthdate_text;
private javax.swing.JButton Close_jlabel;
private javax.swing.JButton Create_Button;
private javax.swing.JLabel Email_jlabel;

```



```

private javax.swing.JTextField Email_text;
private javax.swing.JLabel Name_jlabel;
private javax.swing.JTextField Name_text;
private javax.swing.JPanel New_jpanel;
private javax.swing.JLabel Password_jlabel;
private javax.swing.JTextField Password_text;
private javax.swing.JLabel Phone_jlabel;
private javax.swing.JTextField Phone_text;
private javax.swing.JLabel ReType_jlabel;
private javax.swing.JTextField ReType_text;
private javax.swing.JLabel Title;
private javax.swing.JLabel Username_jlabel;
private javax.swing.JTextField Username_text;
// End of variables declaration//GEN-END:variables
}

```

## Jobs.java

```

/*
 * To change this license header, choose License Headers in Project Properties.
 * To change this template file, choose Tools | Templates
 * and open the template in the editor.
 */
package com.edmond.jobbank.mavenproject1;

import java.io.*;
import java.sql.*;
import org.apache.ibatis.jdbc.ScriptRunner;
/**
 *
 * @author Brian
 */
public class Jobs extends javax.swing.JFrame {
    static Connection con;
    private static FrontPage frontpage;
    private static Jobs jobpage;
    /**
     * Creates new form Jobs
     */
    private Jobs() {
        initComponents();
    }

    public static Jobs getInstance(FrontPage fPage, Connection conn) {
        if(jobpage == null) {
            con = conn;
            frontpage = fPage;
            jobpage = new Jobs();
        }
    }
}

```

```

        return jobpage;
    }

    public void updateText() {
        String output ="Job ID \t Work Period \t Company \t Salary\t Job Title\t\t Job Location\t Job Type\t
Job Deadline\n";
        try {
            String query = "SELECT * FROM jobs";
            Statement st = con.createStatement();
            ResultSet rs = st.executeQuery(query);

            while(rs.next()) {
                for(int j = 0 ; j < 8; j++) {
                    output += rs.getString(j+1)+"\t";
                }
                output += "\n";
            }
            JobList_text.setText(output);
        } catch (SQLException e) {

        }
    }
}

/**
 * This method is called from within the constructor to initialize the form.
 * WARNING: Do NOT modify this code. The content of this method is always
 * regenerated by the Form Editor.
 */
@SuppressWarnings("unchecked")
// <editor-fold defaultstate="collapsed" desc="Generated Code"> //GEN-BEGIN: initComponents
private void initComponents() {

    Jobs_jpanel = new javax.swing.JPanel();
    JobList_jpanel = new javax.swing.JPanel();
    JobList_scroll = new javax.swing.JScrollPane();
    JobList_text = new javax.swing.JTextArea();
    Apply_button = new javax.swing.JButton();
    Apply_text = new javax.swing.JTextField();
    Title_jlabel = new javax.swing.JLabel();
    Close_button = new javax.swing.JButton();
    Back_button = new javax.swing.JButton();

    setDefaultCloseOperation(javax.swing.WindowConstants.EXIT_ON_CLOSE);

    JobList_text.setColumns(20);
    JobList_text.setRows(5);
    JobList_scroll.setViewportView(JobList_text);

    javax.swing.GroupLayout JobList_jpanelLayout = new javax.swing.GroupLayout(JobList_jpanel);

```

```

JobList_jpanel.setLayout(JobList_jpanelLayout);
JobList_jpanelLayout.setHorizontalGroup(
    JobList_jpanelLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
        .addComponent(JobList_scroll, javax.swing.GroupLayout.DEFAULT_SIZE, 715,
Short.MAX_VALUE)
    );
JobList_jpanelLayout.setVerticalGroup(
    JobList_jpanelLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
        .addComponent(JobList_scroll, javax.swing.GroupLayout.DEFAULT_SIZE, 466,
Short.MAX_VALUE)
    );

Apply_button.setText("Apply");
Apply_button.addActionListener(new java.awt.event.ActionListener() {
    public void actionPerformed(java.awt.event.ActionEvent evt) {
        Apply_buttonActionPerformed(evt);
    }
});

Apply_text.addActionListener(new java.awt.event.ActionListener() {
    public void actionPerformed(java.awt.event.ActionEvent evt) {
        Apply_textActionPerformed(evt);
    }
});

Title_jlabel.setFont(new java.awt.Font("Tahoma", 0, 24)); // NOI18N
Title_jlabel.setText("Job Search");

Close_button.setText("Close");
Close_button.addActionListener(new java.awt.event.ActionListener() {
    public void actionPerformed(java.awt.event.ActionEvent evt) {
        Close_buttonActionPerformed(evt);
    }
});

Back_button.setText("Back");
Back_button.addActionListener(new java.awt.event.ActionListener() {
    public void actionPerformed(java.awt.event.ActionEvent evt) {
        Back_buttonActionPerformed(evt);
    }
});

javax.swing.GroupLayout Jobs_jpanelLayout = new javax.swing.GroupLayout(Jobs_jpanel);
Jobs_jpanel.setLayout(Jobs_jpanelLayout);
Jobs_jpanelLayout.setHorizontalGroup(
    Jobs_jpanelLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
        .addGroup(Jobs_jpanelLayout.createSequentialGroup()
            .addGap(36, 36, 36)

```

```

.addGroup(Jobs_jpanelLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
    .addComponent(Title_jlabel)
    .addComponent(JobList_jpanel, javax.swing.GroupLayout.PREFERRED_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE, javax.swing.GroupLayout.PREFERRED_SIZE)
    .addGroup(javax.swing.GroupLayout.Alignment.TRAILING,
Jobs_jpanelLayout.createSequentialGroup())
    .addComponent(Apply_text, javax.swing.GroupLayout.PREFERRED_SIZE, 426,
javax.swing.GroupLayout.PREFERRED_SIZE)
    .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)
    .addComponent(Apply_button)
    .addGap(8, 8, 8)))
.addContainerGap(41, Short.MAX_VALUE))
.addGroup(Jobs_jpanelLayout.createSequentialGroup())

.addGroup(Jobs_jpanelLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
    .addComponent(Close_button)
    .addComponent(Back_button))
.addGap(0, 0, Short.MAX_VALUE))
);
Jobs_jpanelLayout.setVerticalGroup(
Jobs_jpanelLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
.addGroup(Jobs_jpanelLayout.createSequentialGroup())
    .addComponent(Back_button)
    .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)
    .addComponent(Title_jlabel)
    .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)

.addGroup(Jobs_jpanelLayout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASELINE)
    .addComponent(Apply_button)
    .addComponent(Apply_text, javax.swing.GroupLayout.PREFERRED_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE, javax.swing.GroupLayout.PREFERRED_SIZE))
.addGap(18, 18, 18)
    .addComponent(JobList_jpanel, javax.swing.GroupLayout.PREFERRED_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE, javax.swing.GroupLayout.PREFERRED_SIZE)
    .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED,
javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE)
    .addComponent(Close_button))
);

javax.swing.GroupLayout layout = new javax.swing.GroupLayout(getContentPane());
getContentPane().setLayout(layout);
layout.setHorizontalGroup(
    layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
    .addGap(0, 792, Short.MAX_VALUE)
    .addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
        .addGroup(layout.createSequentialGroup()
            .addComponent(Title_jlabel)
            .addComponent(JobList_jpanel, javax.swing.GroupLayout.PREFERRED_SIZE,
                javax.swing.GroupLayout.DEFAULT_SIZE, javax.swing.GroupLayout.PREFERRED_SIZE)
            .addComponent(Apply_text, javax.swing.GroupLayout.PREFERRED_SIZE, 426,
                javax.swing.GroupLayout.PREFERRED_SIZE)
            .addComponent(Apply_button)
            .addComponent(Close_button)
            .addGap(0, 1, Short.MAX_VALUE))
        .addGroup(layout.createSequentialGroup()
            .addComponent(Back_button)
            .addGap(0, 1, Short.MAX_VALUE))
    )
);

```

```

        .addComponent(Jobs_jpanel, javax.swing.GroupLayout.PREFERRED_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE, javax.swing.GroupLayout.PREFERRED_SIZE)
        .addGap(0, 0, Short.MAX_VALUE)))
    );
    layout.setVerticalGroup(
        layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
        .addGap(0, 600, Short.MAX_VALUE)
        .addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
            .addGroup(layout.createSequentialGroup())
                .addGap(0, 0, Short.MAX_VALUE)
                .addComponent(Jobs_jpanel, javax.swing.GroupLayout.PREFERRED_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE, javax.swing.GroupLayout.PREFERRED_SIZE)
                .addGap(0, 0, Short.MAX_VALUE)))
    );

    pack();
} // </editor-fold> // GEN-END: initComponents

private void Apply_textActionPerformed(java.awt.event.ActionEvent evt)
{ // GEN-FIRST: event_Apply_textActionPerformed
    // TODO add your handling code here:
} // GEN-LAST: event_Apply_textActionPerformed

private void Close_buttonActionPerformed(java.awt.event.ActionEvent evt)
{ // GEN-FIRST: event_Close_buttonActionPerformed
    ScriptRunner sr = new ScriptRunner(con);
    try {
        sr.runScript(new BufferedReader(new
FileReader("C:\\Users\\Edmond\\Documents\\Workspace\\cps510\\jobbankuimaven\\scripts\\DropTables.s
ql")));
    } catch (FileNotFoundException e) {
        e.printStackTrace();
    }
    System.exit(0);
} // GEN-LAST: event_Close_buttonActionPerformed

private void Back_buttonActionPerformed(java.awt.event.ActionEvent evt)
{ // GEN-FIRST: event_Back_buttonActionPerformed
    jobpage.setVisible(false);
    frontpage.setVisible(true);
} // GEN-LAST: event_Back_buttonActionPerformed

private void Apply_buttonActionPerformed(java.awt.event.ActionEvent evt)
{ // GEN-FIRST: event_Apply_buttonActionPerformed
    System.out.println(Apply_text.getText()+" applied!");
} // GEN-LAST: event_Apply_buttonActionPerformed

// Variables declaration - do not modify // GEN-BEGIN: variables
private javax.swing.JButton Apply_button;

```

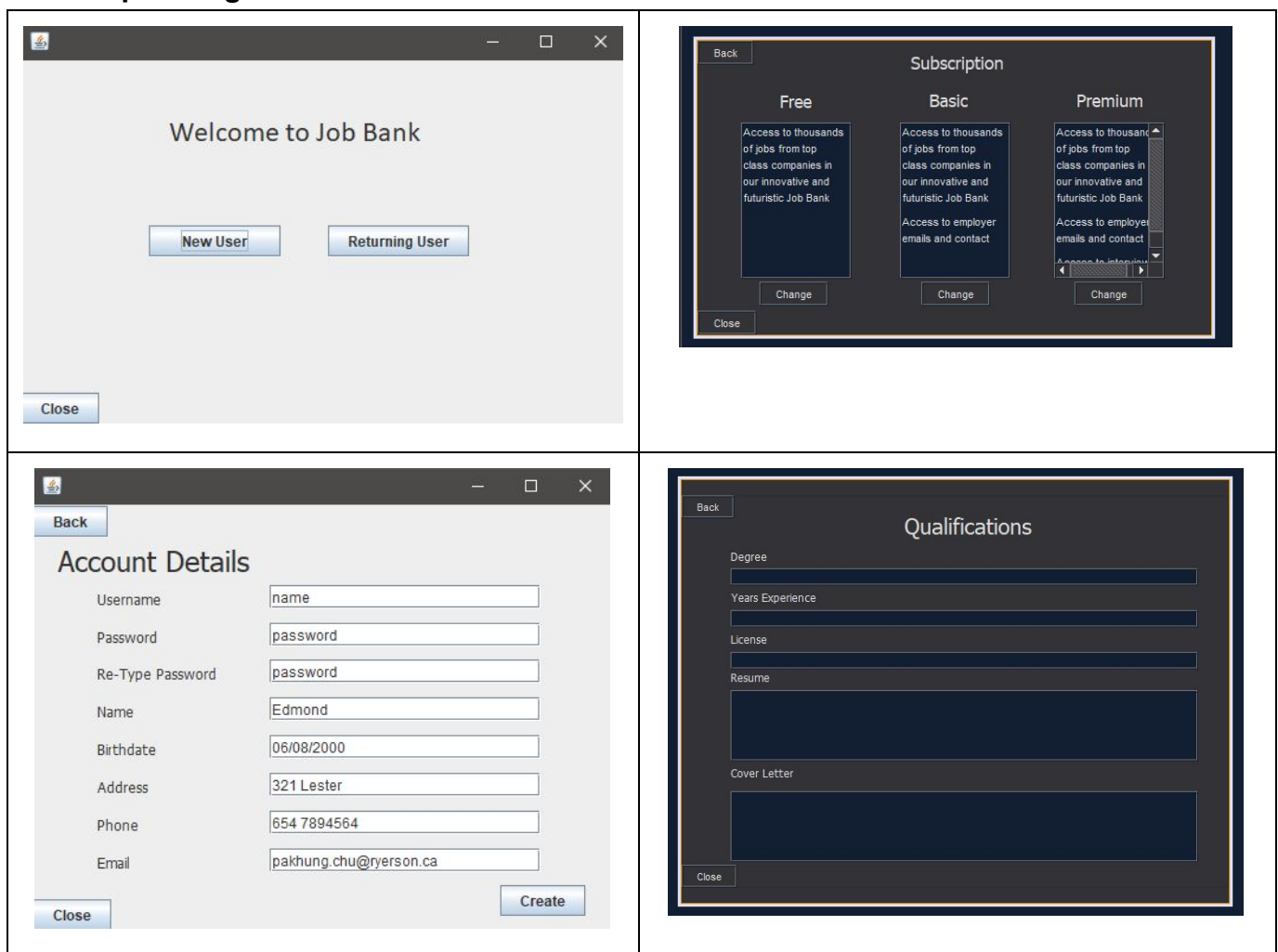
```

private javax.swing.JTextField Apply_text;
private javax.swing.JButton Back_button;
private javax.swing.JButton Close_button;
private javax.swing.JPanel JobList_jpanel;
private javax.swing.JScrollPane JobList_scroll;
private javax.swing.JTextArea JobList_text;
private javax.swing.JPanel Jobs_jpanel;
private javax.swing.JLabel Title_label;
// End of variables declaration//GEN-END:variables

}

```

### Sample Images from GUI:



Welcome

My Account

Jobs

Welcome!

At the Job Bank you will be able to find thousands of opportunities in which you will be able to choose from. No matter what, the Job Bank will always have a option for you. From start ups to Large corporations, all the jobs can be found here.

Use the My Accounts button to go to you account and see your personal information and edit your settings. You can also change your subscription status here to unlock more features.

Use the Jobs button to go and see our large and diverse Job Bank. Here you will be able to find all sorts of jobs and the one just right for you.

Good Luck!

Close

Logout

Back

Account Details

Qualifications

Username

name

Password

password

Subscription

Change Subscription

Name

Edmond

Birthdate

06/08/2000

Address

321 Lester

Phone

654 7894564

Email

pakhung.chu@ryerson.ca

Close

Back

Job Search

Apply

Job ID	Work Period	Company	Salary	Job Title	Job Location	Job Type
Job#1	June 2023	Eabox	\$94960.91	Accountant I	Poland	Contract
Job#2	Aug 2021	Photobug	\$97442.65	Recouter	China	Part-time
Job#3	Dec-2020	Npath	\$94123.85	Occupational Therapist	Indonesia	Internship
Job#4	Apr 2020	Meevee	\$63863.70	Senior Cost Accountant	China	Hourly
Job#5	May 2021	Twitterwire	\$67890.17	Compensation Analyst	Indonesia	Temporary
Job#6	June 2023	Skinix	\$92008.53	Software Test Engineer IV	Argentina	Temporary
Job#7	Dec-2020	Realpoint	\$58867.32	Project Manager	China	Full-time
Job#8	Apr 2020	Zoomcast	\$50664.76	Financial Advisor	China	Temporary
Job#9	Dec-2020	Shuffledrive	\$93700.02	Professor	Russia	Temporary
Job#10	May 2021	Kaymbo	\$69384.08	Quality Control Specialist	Japan	Part-time
Job#11	Aug 2021	Thoughtblab	\$84525.39	Legal Assistant Canada	Internship	2021-10-24 12
Job#12	June 2023	Stigflire	\$82415.02	Sales Representative	Czech Republic	Internship
Job#13	May 2021	Quimba	\$90239.96	Account Executive	Indonesia	Contract
Job#14	June 2023	Ooba	\$65394.41	Project Manager	China	Full-time
Job#15	May 2021	Wikizz	\$65038.73	Staff Accountant I	Philippines	Part-time
Job#16	May 2021	Skymoodle	\$63098.10	Executive Secretary	Indonesia	Hourly
Job#17	May 2021	Skimia	\$50615.84	Professor	Botswana	Temporary
Job#18	Aug 2022	Twitterworks	\$99451.85	Geological Engineer	Vietnam	Hourly
Job#19	May 2021	Cogidoo	\$93237.87	Senior Sales Associate	Peru	Temporary
Job#20	Apr 2020	Tazz	\$56555.67	Environmental Specialist	Brazil	Full-time
Job#21	Aug 2022	Yala	\$68330.29	Accounting Assistant I	Indonesia	Contract
Job#22	Aug 2022	Tagopia	\$56428.05	Research Assistant III	Tajikistan	Internship
Job#23	Aug 2022	Zoonder	\$97542.25	Project Manager	Armenia	Internship
Job#24	Aug 2022	Dynazzy	\$85232.79	Nurse	Macedonia	Contract
Job#25	Aug 2021	Meedoo	\$51918.94	Geologist IV	Brazil	Hourly
Job#26	Dec-2020	Oyonder	\$72584.19	null	null	Part-time
Job#27	Apr 2020	Fanoodle	\$74612.53	Occupational Therapist	Philippines	Internship

Close

Back

Welcome Back

Username

name

Password

password

Login

Close

70

## **Conclusion:**

Working on this Online Job Bank Database Management System has provided us with a foundation in all aspects of database design and implementation. Through the course of the 10 assignments, we realized how important it was to represent data in a clear and concise manner. With the theoretical knowledge of understanding entity relationship diagrams, relational schema design, functional dependencies, normalization, etc., we were able to turn various meaningless sets of data into a useful and accessible database.

In the technical aspect, we familiarize ourselves with the services provided by Oracle 11g and SQL. Using the tools provided, we learned how to create tables, drop tables, populate tables and create custom queries that can display query information in any format. Finally, creating a GUI using Java also familiarized us with the interactions between the front-end and back-end components of the database.

This project was also crucial in exercising our teamwork and communication skills along with improving our technical abilities with project management and software development.

Overall, working on the job bank was different, challenging yet an enjoyable experience. It truly unlocked our potential to use the knowledge and skills that we have acquired in this course.