**University of British Columbia, Vancouver**
Department of Computer Science

# CPSC 304 Project Cover Page

Milestone #: 2
Date: October 20th, 2023

Group Number: 112

| Name | Student Number | CS Alias (Userid) | Preferred E-mail Address |
|---|---|---|---|
| Alex Wu | 56909328 | t3x0n | alexwu120203@gmail.com |
| Steven Huang | 44912350 | t5e1u | steven2003huang@gmail.com |
| Colin Chen | 60726379 | m6z6r | colin.cchn@gmail.com |

By typing our names and student numbers in the above table, we certify that the work in the attached assignment was performed solely by those whose names and student IDs are included above. (In the case of Project Milestone 0, the main purpose of this page is for you to let us know your e-mail address, and then let us assign you to a TA for your project supervisor.)

In addition, we indicate that we are fully aware of the rules and consequences of plagiarism, as set forth by the Department of Computer Science and the University of British Columbia

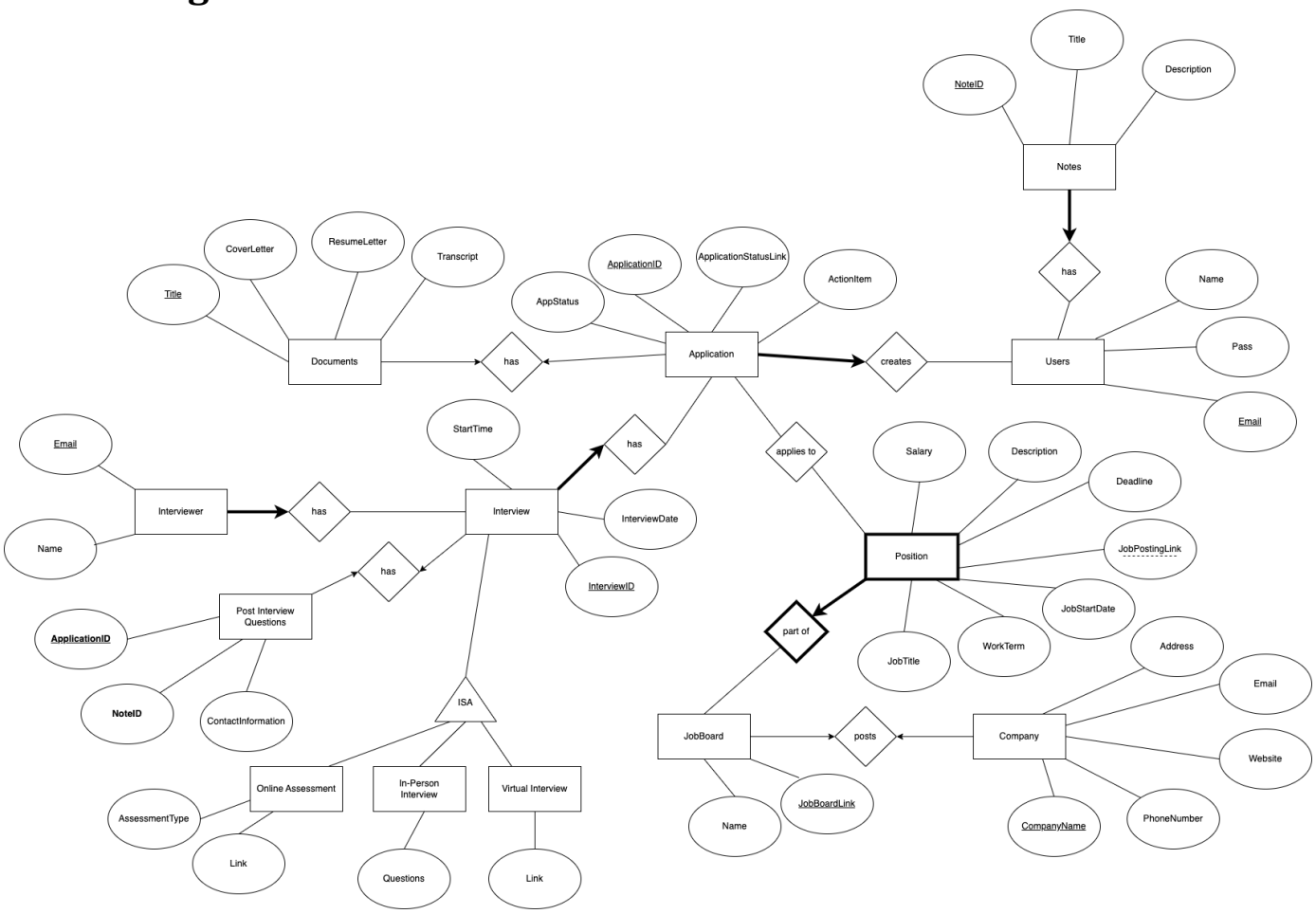
**University of British Columbia, Vancouver**
Department of Computer Science

---

# Project Summary:

Our project manages job applications submitted by individuals, providing an organized approach to job hunting. Users will be able to store and view information such as the application itself, documents related to the application, post-interview notes, etc..

# ER Diagram:



**University of British Columbia, Vancouver**
Department of Computer Science

# ER Diagram Schema:

Underlined attributes are primary keys. **Bolded** attributes are foreign keys. *Italicized* attributes are candidate keys.

| |
|---|
| Users (Email: varchar[100], Name: varchar[100], Pass: varchar[100]) |
| Notes (NoteID: int, Title: varchar[2000], Description: varchar[2000], **UserEmail**: varchar[100] NOT NULL) |
| Application(ApplicationID: int, AppStatus: varchar[20], ApplicationStatusLink: varchar[2000], ActionItem varchar[2000], **UserEmail:** varchar[100] NOT NULL) |
| Documents (Title: varchar[50], CoverLetter: varchar[2000], Resume: varchar[2000], Transcript: varchar[2000], **ApplicationID:** int NOT NULL) |
| Interview (InterviewID: int, **ApplicationID**: int NOT NULL, StartTime: char[5], InterviewDate: char[15]) |
| Position(JobPostingLink: varchar[2000], JobTitle: varchar[50], Description: varchar[2000], Deadline: char[10], Salary: int CHECK > 0, JobStartDate: char[15], WorkTerm: char[15], **JobBoardLink:** varchar[2000] NOT NULL, **CompanyName:** varchar[50] NOT NULL) |
| JobBoard(JobBoardLink: varchar[2000], Name: varchar[50], **CompanyName:** varchar[50] NOT NULL) |
| AppliesTo(**JobBoardLink:** varchar[2000], **JobPostingLink** varchar[2000] , **ApplicationID:** int) |
| Company (CompanyName: varchar[50], *Website*: varchar[2000], *Email*: varchar[50], Address: varchar[50], *Number*: varchar[50]) |
| Interviewer (Email: varchar[50], Name: varchar[50], **InterviewID:** int NOT NULL) |
| PostInterviewQuestion(**ApplicationID**: int, *ContactInformation:* varchar[200], **NoteID**: int, **InterviewID:** int) |
| ISA:<br><br>OnlineAssessment(**InterviewID:** int, AssessmentType: varchar[10], Link: varchar[2000], Industry: varchar[50])<br><br>InPersonInterview(**InterviewID:** int, Questions: varchar[2000])<br><br>VirtualInterview(**InterviewID:** int, Link: varchar[2000]) |

**University of British Columbia, Vancouver**
Department of Computer Science

# Functional Dependencies (FDs):

| | |
|---|---|
| Notes | NoteID → Title, Description |
| Users | Email → Name, Pass |
| Application | ApplicationID → ApplicationStatusLink, AppStatus, JobPostingLink |
| Documents | Title → CoverLetter, ResumeLetter, Transcript |
| Interviewer | Email → Name |
| Interview | InterviewID → Date, StartTime |
| Virtual Interview | InterviewID → Link |
| In-Person Interview | InterviewID → Questions |
| Online Assessment | InterviewID → AssessmentType, Link |
| Post Interview Questions | ApplicationID → ContactInformation, NoteID |
| JobBoard | JobBoardLink → Name |
| Position | JobBoardLink, JobTitle → Deadline, Description, Salary |
| Company | CompanyName → Website, Email, Address, Number |
| Non-Primary/ Candidate FD's: | JobStartTime → Work Term<br>JobBoardName, Description → Salary<br>AppStatus → ActionItem |

**University of British Columbia, Vancouver**
Department of Computer Science

# Normalization:

## Relations that require decomposition into 3NF:

Application(ApplicationID: int, AppStatus: varchar[20] , ActionItem: varchar[2000], **JobPostingLink**: varchar[2000], ApplicationStatusLink: varchar[2000], **UserEmail:** varchar[100])

*The key for application is ApplicationID These are the FDs:*

ApplicationID → ApplicationStatusLink, AppStatus, JobPostingLink, UserEmail, ActionItem
AppStatus → ActionItem

*Currently, this relation is not in 3NF because for the FD: Status → ActionItem, the left side is neither a superkey, nor is the right side part of the key. Next, we need to ensure it's in minimal cover.*

*Put in Standardized Form and Simplified LHS*
ApplicationID → ApplicationStatusLink
ApplicationID → AppStatus
ApplicationID → JobPostingLink
ApplicationID → UserEmail
ApplicationID → ActionItem
AppStatus → ActionItem

*Removed Redundant FDs*
ApplicationID → ApplicationStatusLink
ApplicationID → AppStatus
ApplicationID → JobPostingLink
ApplicationID → UserEmail
AppStatus → ActionItem

*With Synthesis*
ApplicationStatusLink(ApplicationID, ApplicationStatusLink)
ApplicationStatus(ApplicationID, AppStatus)
ApplicationJobPostingLink(ApplicationID, **JobPostingLink**)
ApplicationUserEmail(ApplicationID, **UserEmail**)
ApplicationActionItem(AppStatus, ActionItem)

We have created a relation for each FD. Also, we have confirmed one of the FDs contains the key. There are also no redundant relations.

Position(JobPostingLink: varchar[2000], JobTitle: varchar[50], Description: varchar[2000], Deadline: char[15], Salary: int, JobStartDate: char[15], WorkTerm: varchar[15], JobBoardLink**:** varchar[2000] NOT NULL, **CompanyName:** varchar[50] NOT NULL)

The key for Position is JobPostingLink, JobBoardLink
These are the FDs:

JobBoardLink, JobPostingLink → Deadline, Description, Salary, WorkTerm, JobStartDate, JobTitle, CompanyName
JobStartDate -> WorkTerm
JobTitle, Description -> Salary

*Put in Standardize Form:*

JobBoardLink, JobPostingLink → Deadline
JobBoardLink, JobPostingLink → Description
JobBoardLink, JobPostingLink → Salary
JobBoardLink, JobPostingLink → WorkTerm
JobBoardLink, JobPostingLink → JobStartDate
JobBoardLink, JobPostingLink → JobTitle
JobBoardLink, JobPostingLink → CompanyName
JobStartDate -> WorkTerm
JobTitle, Description -> Salary

*Minimize LHS:*
*(already minimize)*

*Remove Redundant:*

JobBoardLink, JobPostingLink → Deadline
JobBoardLink, JobPostingLink → Description
JobBoardLink, JobPostingLink → JobStartDate
JobBoardLink, JobPostingLink → JobTitle
JobBoardLink, JobPostingLink → CompanyName
JobStartDate -> WorkTerm
JobTitle, Description -> Salary

*We have created a relation for each FD. Also, we have confirmed one of the FDs contains the key. There are also no redundant relations.*

PositionDeadline(JobBoardLink, JobPostingLink, Deadline)
PositionDescription(JobBoardLink, JobPostingLink, Description)
PositionJobStartDate(JobBoardLink, JobPostingLink, JobStartDate)
PositionJobTitle(JobBoardLink, JobPostingLink, JobTitle)
PositionCompanyName(JobBoardLink, JobPostingLink, **CompanyName**)
PositionWorkTerm(JobStartDate, WorkTerm)
PositionSalary(JobTitle, Description, Salary)

**University of British Columbia, Vancouver**
Department of Computer Science

## Relations that were already in 3NF

| |
|---|
| Users (Email: varchar[100], Name: varchar[100], Pass: varchar[100]) |
| Notes (NoteID: int, Title: varchar[2000], Description: varchar[2000], **UserEmail**: varchar[100] NOT NULL) |
| Documents (Title: varchar[50], CoverLetter: varchar[2000], Resume: varchar[2000], Transcript: varchar[2000], **ApplicationID:** int NOT NULL) |
| Interview (InterviewID: int, **ApplicationID**: int NOT NULL, StartTime: char[5], InterviewDate: char[15]) |
| JobBoard(JobBoardLink: varchar[2000], Name: varchar[50], **CompanyName:** varchar[50] NOT NULL) |
| AppliesTo(**JobBoardLink:** varchar[2000], **JobPostingLink** varchar[2000] , **ApplicationID:** int) |
| Company (CompanyName: varchar[50], *Website*: varchar[2000], *Email*: varchar[50], Address: varchar[50], *Number*: varchar[50]) |
| Interviewer (Email: varchar[50], Name: varchar[50], **InterviewID:** int NOT NULL) |
| PostInterviewQuestion(**ApplicationID**: int, *ContactInformation:* varchar[200], **NoteID**: int, **InterviewID:** int) |
| OnlineAssessment(**InterviewID:** int, AssessmentType: varchar[50], Link: varchar[2000], Industry: varchar[50])<br><br>InPersonInterview(**InterviewID:** int, Questions: varchar[2000])<br><br>VirtualInterview(**InterviewID:** int, Link: varchar[2000]) |

## SQL DDL Statements:

The SQL DDL statements required to create all the tables from item #6. The statements should use the appropriate foreign keys, primary keys, UNIQUE constraints, etc.
Unless you know that you will always have exactly x characters for a given character, it is better to use the VARCHAR data type as opposed to a CHAR(Y). For example, UBC courses always use four characters to represent which department offers a course. In that case, you will want to use CHAR(4) for the department attribute in your SQL DDL statement. If you are trying to represent the name of a UBC course, you will want to use VARCHAR as the number of characters in a course name can vary greatly.

```
CREATE TABLE Users (
  Email VARCHAR(100),
  Name VARCHAR(100),
  Pass VARCHAR(100),
  PRIMARY KEY (Email)
);
```

```
CREATE TABLE Notes (
  NoteID INT,
  Title VARCHAR(2000),
  Description VARCHAR(2000),
  UserEmail VARCHAR(100) NOT NULL,
  PRIMARY KEY (NoteID),
  FOREIGN KEY (UserEmail) REFERENCES Users(Email) ON DELETE CASCADE
);
```

```
CREATE TABLE ApplicationStatusLink (
  ApplicationID INT,
  AppStatusLink VARCHAR(2000),
  PRIMARY KEY (ApplicationID)
);
```

```
CREATE TABLE ApplicationStatus (
  ApplicationID INT,
  AppStatus VARCHAR(20),
  PRIMARY KEY (ApplicationID)
);
```

```
CREATE TABLE ApplicationJobPostingLink (
  ApplicationID INT,
  JobPostingLink VARCHAR(2000),
  PRIMARY KEY (AppStatus),
```

```
  FOREIGN KEY (JobPostingLink) REFERENCES JobBoard_Position(JobPostingLink) ON
DELETE CASCADE
);
```

```
CREATE TABLE ApplicationUserEmail (
  ApplicationID INT,
  UserEmail VARCHAR(100) NOT NULL,
  PRIMARY KEY (ApplicationID),
  FOREIGN KEY (UserEmail) REFERENCES Users(Email) ON DELETE CASCADE
);
```

```
CREATE TABLE ApplicationActionItem (
  AppStatus VARCHAR(20),
  ActionItem VARCHAR(2000),
  PRIMARY KEY (AppStatus)
);
```

```
CREATE TABLE AppliesTo (
  ApplicationID INT,
  JobPostingLink VARCHAR(2000),
  JobBoardLink VARCHAR(2000),
  PRIMARY KEY (ApplicationID, JobPostingLink, JobBoardLink),
  FOREIGN KEY (ApplicationID) REFERENCES Application(ApplicationID) ON DELETE
CASCADE,
  FOREIGN KEY (JobBoardLink) REFERENCES JobBoard(JobBoardLink) ON DELETE
CASCADE,
  FOREIGN KEY (JobPostingLink) REFERENCES JobBoard_Position(JobPostingLink) ON
DELETE CASCADE
);
```

```
CREATE TABLE Documents (
  Title VARCHAR(50),
  CoverLetter VARCHAR(2000),
  ResumeLetter VARCHAR(2000),
  Transcript VARCHAR(2000),
  ApplicationID INT NOT NULL,
  PRIMARY KEY (Title),
  FOREIGN KEY (ApplicationID) REFERENCES Application(ApplicationID) ON DELETE
CASCADE
);
```

```
CREATE TABLE Interview (
  InterviewID INT,
  ApplicationID INT NOT NULL,
```

```
  StartTime CHAR(5),
  InterviewDate CHAR(15),
  PRIMARY KEY (InterviewID),
  FOREIGN KEY (ApplicationID) REFERENCES Application(ApplicationID) ON DELETE
CASCADE
);
```

```
CREATE TABLE Company (
  CompanyName VARCHAR(50),
  Website VARCHAR(2000),
  Email VARCHAR(50),
  Address VARCHAR(50),
  PhoneNumber VARCHAR(50),
  PRIMARY KEY (CompanyName)
);
```

```
CREATE TABLE JobBoard (
  JobBoardLink VARCHAR(2000),
  Name VARCHAR(50),
  CompanyName VARCHAR(50) NOT NULL,
  PRIMARY KEY (JobBoardLink),
  FOREIGN KEY (CompanyName) REFERENCES Company(CompanyName) ON DELETE
CASCADE
);
```

```
CREATE TABLE JobBoard_Position (
  JobBoardLink VARCHAR(2000),
  JobPostingLink VARCHAR(2000),
  Deadline CHAR(15),
  PRIMARY KEY (JobBoardLink, JobPostingLink),
  FOREIGN KEY (JobBoardLink) REFERENCES JobBoard(JobBoardLink) ON DELETE
CASCADE
);
```

```
CREATE TABLE JobBoard_PositionPay (
  JobTitle VARCHAR(50),
  Description VARCHAR(2000),
  Salary INT CHECK (Salary > 0),
  PRIMARY KEY (JobTitle, Description)
);
```

```
CREATE TABLE JobBoard_PositionBegins (
 JobStartDate CHAR(15),
 WorkTerm CHAR(15),
```

```
PRIMARY KEY (JobStartDate)
);
```

```
CREATE TABLE Interviewer (
  Email VARCHAR(50),
  Name VARCHAR(50),
  InterviewID INT NOT NULL,
  PRIMARY KEY (Email),
  FOREIGN KEY (InterviewID) REFERENCES Interview(InterviewID) ON DELETE
CASCADE
);
```

```
CREATE TABLE PostInterviewQuestion (
  ApplicationID INT,
  ContactInformation VARCHAR(200),
  NoteID INT,
  InterviewID INT,
  PRIMARY KEY (ApplicationID),
  FOREIGN KEY (ApplicationID) REFERENCES Application(ApplicationID) ON DELETE
CASCADE,
  FOREIGN KEY (NoteID) REFERENCES Notes(NoteID) ON DELETE CASCADE,
  FOREIGN KEY (InterviewID) REFERENCES Interview(InterviewID) ON DELETE
CASCADE
);
```

```
CREATE TABLE OnlineAssessment (
  InterviewID INT,
  AssessmentType VARCHAR(50),
  Link VARCHAR(2000),
  PRIMARY KEY (InterviewID),
  FOREIGN KEY (InterviewID) REFERENCES Interview(InterviewID) ON DELETE
CASCADE
);
```

```
CREATE TABLE InPersonInterview (
  InterviewID INT,
  Questions VARCHAR(2000),
  PRIMARY KEY (InterviewID),
  FOREIGN KEY (InterviewID) REFERENCES Interview(InterviewID) ON DELETE
CASCADE
);
```

```
CREATE TABLE VirtualInterview (
  InterviewID INT,
```

```
  Link VARCHAR(2000),
  PRIMARY KEY (InterviewID),
  FOREIGN KEY (InterviewID) REFERENCES Interview(InterviewID) ON DELETE
CASCADE
);
```

**University of British Columbia, Vancouver**
Department of Computer Science

---

## INSERT STATEMENTS:

INSERT statements to populate each table with at least 5 tuples. You will likely want to have more than 5 tuples so that you can have meaningful queries later.

```
INSERT INTO Users (Email, Name, Pass)
SELECT 'colin@ubc.ca', 'Colin', 'Chen' FROM dual
UNION ALL
SELECT 'steven@ubc.ca', 'Steven', 'Huang' FROM dual
UNION ALL
SELECT 'alex@ubc.ca', 'Alex', 'Wu' FROM dual
UNION ALL
SELECT 'extra1@ubc.ca', 'Extra', '1' FROM dual
UNION ALL
SELECT 'extra2@ubc.ca', 'Extra', '2' FROM dual;

SELECT Email, Name, Pass
FROM Users;
```

```
INSERT INTO Notes (NoteID, Title, Description, UserEmail)
SELECT 1, 'Todo List',  'Apply to Jobs, Prepare for Interview, Leetcode More',
'alexwu@gmail.com' FROM dual
UNION ALL
SELECT 2, 'Homework', 'CPSC 304 Milestone 2 Project', 'alexwu@gmail.com' FROM dual
UNION ALL
SELECT 3, 'Interviews', 'Oct 24th 2:00PM Amazon, Oct 30th 3:00PM Meta',
'alexwu@gmail.com' FROM dual
UNION ALL
SELECT 4, 'Leetcode Practice', 'Fizzbuzz, Valid Parentheses', 'alexwu@gmail.com' FROM
dual
UNION ALL
SELECT 5, 'Coffee Chat', 'Oct 30th Nathan Lee', 'alexwu@gmail.com' FROM dual
UNION ALL

SELECT NoteID, Title, Description, UserEmail
FROM Notes;
```

```
INSERT INTO Documents (Title, CoverLetter, ResumeLetter, Transcript, ApplicationID)
SELECT 'CS Rough Draft', 'https://drive.google.com/file/1a',
'https://drive.google.com/file/1a', 'https://drive.google.com/file/1a', 1 FROM dual
UNION ALL
SELECT 'CS Final Copy, 'https://drive.google.com/file/2a',
'https://drive.google.com/file/2a', 'https://drive.google.com/file/2a', 1 FROM dual
UNION ALL
```

```
SELECT 'Finance Rough Draft', 'https://drive.google.com/file/3a',
'https://drive.google.com/file/3a', 'https://drive.google.com/file/3a' 2 FROM dual
UNION ALL
SELECT 'Finance Final Copy, 'https://drive.google.com/file/4a',
'https://drive.google.com/file/4a', 'https://drive.google.com/file/4a' 2 FROM dual
UNION ALL
SELECT 'Marketing Rough Draft', 'https://drive.google.com/file/4a',
'https://drive.google.com/file/4a', 'https://drive.google.com/file/4a' 3 FROM dual
UNION ALL

SELECT Title, CoverLetter, ResumeLetter, Transcript, ApplicationID
FROM Documents;
```

```
INSERT INTO Interview (InterviewID, InterviewDate, StartTime, ApplicationID)
SELECT 1, '11/10/2023', '2:00PM PST', 3 FROM dual
UNION ALL
SELECT 2, '11/12/2023', '2:00PM PST', 1 FROM dual
UNION ALL
SELECT 3, '11/15/2023', '12:00PM PST', 1 FROM dual
UNION ALL
SELECT 4, '11/22/2023', '11:00AM PST', 2 FROM dual
UNION ALL
SELECT 5, '11/27/2023', '1:00PM PST', 2 FROM dual
UNION ALL

SELECT InterviewID, CoverLetter, ResumeLetter, Transcript, ApplicationID
FROM Interview;
```

```
INSERT INTO Interviewer (Email, Name, InterviewID)
SELECT 'recruiter1@gmail.com', 'John Smith', 2 FROM dual
UNION ALL
SELECT 'recruiter2@gmail.com', 'Nathan Suen', 3 FROM dual
UNION ALL
SELECT 'recruiter3@gmail.com', 'Tammy Goel', 3 FROM dual
UNION ALL
SELECT 'recruiter4@gmail.com', 'Rammy Hadyullath', 3 FROM dual
UNION ALL
SELECT 'recruiter5@gmail.com', 'Marcus Lai', 4 FROM dual
UNION ALL

SELECT Email, Name, InterviewID
FROM Interviewer;
```

```
INSERT INTO OnlineAssessment (InterviewID, AssessmentType, Link)
SELECT 1, 'Hackerrank' 'https://www.hackerrank.com' FROM dual
UNION ALL
SELECT 2, 'Leetcode' 'https://www.leetcode.com' FROM dual
UNION ALL
SELECT 3, 'IQTest' 'https://www.iqtest.com' FROM dual
UNION ALL
SELECT 4, PuzzleGames 'https://www.puzzlegames.com' FROM dual
UNION ALL
SELECT 5, 'CodeReview' 'https://www.codereview.com' FROM dual
UNION ALL

SELECT InterviewID, AssessmentType, Link
FROM OnlineAssessment;
```

```
INSERT INTO InPersonInterview (InterviewID, Questions)
SELECT 1, 'Tell me about yourself' FROM dual
UNION ALL
SELECT 2, 'Tell me a time had conflict with a teammate and how you resolved it' FROM
dual
UNION ALL
SELECT 3, 'What are your strengths and weaknesses' FROM dual
UNION ALL
SELECT 4, 'Why do you want to join our company?' FROM dual
UNION ALL
SELECT 5, 'What is your experience with Javascript?' FROM dual
UNION ALL

SELECT InterviewID, Questions
FROM InPersonInterview;
```

```
INSERT INTO VirtualInterview (InterviewID, Link)
SELECT 1, 'https://www.zoomus1.com' FROM dual
UNION ALL
SELECT 2, 'https://www.zoomus2.com' FROM dual
UNION ALL
SELECT 3, 'https://www.zoomus3.com' FROM dual
UNION ALL
SELECT 4, 'https://www.zoomus4.com' FROM dual
UNION ALL
SELECT 5, 'https://www.zoomus5.com' FROM dual
UNION ALL
```

```
SELECT InterviewID, Link
FROM VirtualInterview;
```

```
INSERT INTO PostInterviewQuestion (ApplicationID, NoteID, ContactInformation,
InterviewID)
SELECT 1, 1, 'recuiter1@gmail.com, '604-111-1111', 2 FROM dual
UNION ALL
SELECT 2, 2, 'recuiter2@gmail.com, '604-222-2222', 2FROM dual
UNION ALL
SELECT 3, 3, 'recuiter3@gmail.com, '604-333-3333', 1 FROM dual
UNION ALL
SELECT 4, 4, 'recuiter4@gmail.com, '604-444-4444', 3 FROM dual
UNION ALL
SELECT 5, 5, 'recuiter5@gmail.com, '604-555-5555', 1 FROM dual
UNION ALL

SELECT ApplicationID, NoteID, ContactInformation, InterviewID
FROM PostInterviewQuestion;
```

```
INSERT INTO JobBoard (JobBoardLink, Name, CompanyName)
SELECT 'https://www.appleboard.com', 'Linkedln', 'Meta' FROM dual
UNION ALL
SELECT 'https://www.metaboard.com',  'Glassdoor', 'Meta' FROM dual
UNION ALL
SELECT 'https://www.microsoftboard.com', 'Co-op', 'Apple' FROM dual
UNION ALL
SELECT 'https://www.netflixboard.com', 'Google Jobs', 'Google' FROM dual
UNION ALL
SELECT 'https://www.googleboard.com', 'SmartRecruiters', 'Microsoft' FROM dual
UNION ALL

SELECT JobBoardLink, Name, CompanyName
FROM JobBoard;
```

```
INSERT INTO Company (CompanyName, Website, Address, Email, PhoneNumber)
SELECT 'Google', 'https://www.google.com', '5435 Robson St', 'googleemail@gmail.com',
'778-111-1111' FROM dual
UNION ALL
SELECT 'Meta', 'https://www.meta.com', '2963 Kerr Ave', 'metaemail@gmail.com',
'778-222-2222' FROM dual
UNION ALL
SELECT 'Apple', 'https://www.apple.com', '1014 Shelby Dr', 'appleemail@gmail.com',
'778-333-3333' FROM dual
```

```
UNION ALL
SELECT 'Netflix', 'https://www.netflix.com', '3285 Silicon St', 'netflixemail@gmail.com',
'778-444-4444' FROM dual
UNION ALL
SELECT 'Microsoft', 'https://www.microsoft.com', '5009 Hornby Ave',
'microsoftemail@gmail.com', '778-555-5555' FROM dual
UNION ALL

SELECT CompanyName, Website, Address, Email, PhoneNumber
FROM Company;
```

```
INSERT INTO JobBoard (JobBoardLink, Name, CompanyName)
SELECT 'https://www.appleboard.com', 'Linkedln', 'Meta' FROM dual
UNION ALL
SELECT 'https://www.metaboard.com',  'Glassdoor', 'Meta' FROM dual
UNION ALL
SELECT 'https://www.microsoftboard.com', 'Co-op', 'Apple' FROM dual
UNION ALL
SELECT 'https://www.netflixboard.com', 'Google Jobs', 'Google' FROM dual
UNION ALL
SELECT 'https://www.googleboard.com', 'SmartRecruiters', 'Microsoft' FROM dual
UNION ALL

SELECT JobBoardLink, Name, CompanyName
FROM JobBoard;
```

```
INSERT INTO PositionDeadline (JobBoardLink, JobPostingLink, Deadline)
SELECT 'https://www.appleboard.com', 'https://www.jobposting1.com', 'November 3rd,
2023' FROM dual
UNION ALL
SELECT 'https://www.metaboard.com', 'https://www.jobposting2.com', 'November 5th,
2023' FROM dual
UNION ALL
SELECT 'https://www.microsoftboard.com', 'https://www.jobposting3.com', 'November
11th, 2023' FROM dual
UNION ALL
SELECT 'https://www.netflixboard.com', 'https://www.jobposting3.com', 'November
15th, 2023' FROM dual
UNION ALL
SELECT 'https://www.googleboard.com', 'https://www.jobposting4.com', 'November
23rd, 2023' FROM dual
UNION ALL
```

```
SELECT JobBoardLink, JobPostingLink, Deadline
FROM PositionDeadline;
```

```
INSERT INTO PositionDescription (JobBoardLink, JobPostingLink, Description)
SELECT 'https://www.appleboard.com', 'https://www.jobposting1.com', 'You may have experience or exposure to MERN Stack, JavaScript and HTML, Python, HTML/CSS' FROM dual
UNION ALL
SELECT 'https://www.metaboard.com', 'https://www.jobposting2.com', 'You may have experience or exposure to AI and Machine Learning Work Experience' FROM dual
UNION ALL
SELECT 'https://www.microsoftboard.com', 'https://www.jobposting3.com', 'You may have experience or exposure to UX/UI Design and Figma Experience' FROM dual
UNION ALL
SELECT 'https://www.googleboard.com', 'https://www.jobposting4.com', 'You may have experience or exposure to Financial Modelling with Excel' FROM dual
UNION ALL
SELECT 'https://www.netflixboard.com', 'https://www.jobposting5.com', 'You may have experience or exposure to Python and SQL' FROM dual
UNION ALL

SELECT JobBoardLink, JobPostingLink, Description
FROM PositionDescription;
```

```
INSERT INTO PositionJobStartDate (JobBoardLink, JobPostingLink, JobStartDate)
SELECT 'https://www.appleboard.com', 'https://www.jobposting1.com', 'October 25th, 2023' FROM dual
UNION ALL
SELECT 'https://www.metaboard.com', 'https://www.jobposting2.com', 'November 1st, 2023' FROM dual
UNION ALL
SELECT 'https://www.microsoftboard.com', 'https://www.jobposting3.com', 'November 4th, 2023' FROM dual
UNION ALL
SELECT 'https://www.googleboard.com', 'https://www.jobposting4.com', 'November 14th, 2023' FROM dual
UNION ALL
SELECT 'https://www.netflixboard.com', 'https://www.jobposting5.com', 'November 14th, 2023'  FROM dual
UNION ALL

SELECT JobBoardLink, JobPostingLink, JobStartDate
FROM PositionJobStartDate;
```

```
INSERT INTO PositionJobTitle (JobBoardLink, JobPostingLink, JobTitle)
SELECT 'https://www.appleboard.com', 'https://www.jobposting1.com', 'Software
Engineer' FROM dual
UNION ALL
SELECT 'https://www.metaboard.com', 'https://www.jobposting2.com', 'Full-Stack
Developer' FROM dual
UNION ALL
SELECT 'https://www.microsoftboard.com', 'https://www.jobposting3.com', 'Machine
Learning Software Engineer' FROM dual
UNION ALL
SELECT 'https://www.googleboard.com', 'https://www.jobposting4.com', 'Software
Engineer in Test' FROM dual
UNION ALL
SELECT 'https://www.netflixboard.com', 'https://www.jobposting5.com', 'Financial
Analyst'  FROM dual
UNION ALL

SELECT JobBoardLink, JobPostingLink, JobStartTitle
FROM PositionJobStartTitle;
```

```
INSERT INTO PositionJobCompany (JobBoardLink, JobPostingLink, Company)
SELECT 'https://www.appleboard.com', 'https://www.jobposting1.com', 'Google' FROM
dual
UNION ALL
SELECT 'https://www.metaboard.com', 'https://www.jobposting2.com', 'Facebook' FROM
dual
UNION ALL
SELECT 'https://www.microsoftboard.com', 'https://www.jobposting3.com', 'Twitter'
FROM dual
UNION ALL
SELECT 'https://www.googleboard.com', 'https://www.jobposting4.com', 'TikTok' FROM
dual
UNION ALL
SELECT 'https://www.netflixboard.com', 'https://www.jobposting5.com', 'Tesla'  FROM
dual
UNION ALL

SELECT JobBoardLink, JobPostingLink, Company
FROM PositionJobCompany;
```

```
INSERT INTO PositionJobCompany (JobBoardLink, JobPostingLink, Company)
```

```
SELECT 'https://www.appleboard.com', 'https://www.jobposting1.com', 'Google' FROM
dual
UNION ALL
SELECT 'https://www.metaboard.com', 'https://www.jobposting2.com', 'Facebook' FROM
dual
UNION ALL
SELECT 'https://www.microsoftboard.com', 'https://www.jobposting3.com', 'Twitter'
FROM dual
UNION ALL
SELECT 'https://www.googleboard.com', 'https://www.jobposting4.com', 'TikTok' FROM
dual
UNION ALL
SELECT 'https://www.netflixboard.com', 'https://www.jobposting5.com', 'Tesla'  FROM
dual
UNION ALL

SELECT JobBoardLink, JobPostingLink, Company
FROM PositionJobCompany;
```

```
INSERT INTO PositionWorkTerm (JobStartDate, WorkTerm)
SELECT 'Novemeber 2nd, 2023', 'Winter 2024' FROM dual
UNION ALL
SELECT 'Novemeber 10th, 2023', 'Winter 2024' FROM dual
UNION ALL
SELECT 'Novemeber 12th, 2023', 'Winter 2024' FROM dual
UNION ALL
SELECT 'Novemeber 28th, 2023', 'Winter 2024' FROM dual
UNION ALL
SELECT 'Novemeber 30th, 2023', 'Winter 2024' FROM dual
UNION ALL

SELECT JobStartDate, WorkTerm
FROM PositionWorkTerm;
```

```
INSERT INTO PositionSalary (JobTitle, Description, Salary)
SELECT 'Software Engineer I', 'level 1 engineer', 100000 FROM dual
UNION ALL
SELECT 'Software Engineer II', 'level 2 engineer', 200000 FROM dual
UNION ALL
SELECT 'Software Engineer III', 'level 3 engineer', 300000 FROM dual
UNION ALL
SELECT 'Software Engineer IV', 'level 4 engineer', 400000 FROM dual
UNION ALL
```

```
SELECT 'Software Engineer V', 'level 5 engineer', 500000  FROM dual
UNION ALL

SELECT JobTitle, Description, Salary
FROM PositionSalary;
```

```
INSERT INTO ApplicationStatusLink (ApplicationID, ApplicationStatusLink)
SELECT 1, 'https://www.appstatuslink1.com' FROM dual
UNION ALL
SELECT 2, 'https://www.appstatuslink2.com' FROM dual
UNION ALL
SELECT 3, 'https://www.appstatuslink3.com' FROM dual
UNION ALL
SELECT 4, 'https://www.appstatuslink4.com' FROM dual
UNION ALL
SELECT 5, 'https://www.appstatuslink5.com' FROM dual
UNION ALL

SELECT ApplicationID, ApplicationStatusLink
FROM ApplicationStatusLink;
```

```
INSERT INTO ApplicationStatus (ApplicationID, AppStatus)
SELECT 1, 'Applied' FROM dual
UNION ALL
SELECT 2, 'Not Applied' FROM dual
UNION ALL
SELECT 3, 'Interview' FROM dual
UNION ALL
SELECT 4, 'Waiting for Response' FROM dual
UNION ALL
SELECT 5, 'Applied' FROM dual
UNION ALL

SELECT ApplicationID, AppStatus
FROM ApplicationStatus;
```

```
INSERT INTO ApplicationJobPostingLink (ApplicationID, JobPostingLink)
SELECT 1, 'https://www.jobposting1.com' FROM dual
UNION ALL
SELECT 2, 'https://www.jobposting2.com' FROM dual
UNION ALL
SELECT 3, 'https://www.jobposting3.com' FROM dual
```

```
UNION ALL
SELECT 4, 'https://www.jobposting4.com' FROM dual
UNION ALL
SELECT 5, 'https://www.jobposting5.com' FROM dual
UNION ALL

SELECT ApplicationID, JobPostingLink
FROM ApplicationJobPostingLink;
```

```
INSERT INTO ApplicationUserEmail (ApplicationID, UserEmail)
SELECT 1, 'alexwu02@gmail.com' FROM dual
UNION ALL
SELECT 2, 'raivelmangaoil@gmail.com' FROM dual
UNION ALL
SELECT 3, 'colinchen@gmail.com' FROM dual
UNION ALL
SELECT 4, 'masonsuen@gmail.com' FROM dual
UNION ALL
SELECT 5, 'stevenhuang@gmail.com' FROM dual
UNION ALL

SELECT ApplicationID, UserEmail
FROM ApplicationUserEmail;
```

```
INSERT INTO ApplicationActionItem (AppStatus, ActionItem)
SELECT 'Applied', 'Complete Online Assessment' FROM dual
UNION ALL
SELECT 'Interview', 'Prepare for Interview' FROM dual
UNION ALL
SELECT 'Not Applied', 'Submit Application' FROM dual
UNION ALL
SELECT 'Applied', 'Offer Recevied' FROM dual
UNION ALL
SELECT 'Applied', 'Practice for Technical Interview' FROM dual
UNION ALL

SELECT AppStatus, ActionItem
FROM ApplicationActionItem;
```