Part 2

A) Is the Schema in 1NF?

No it is not.

We want all the attributes to be atomic

Department Location is not atomic

If we want to access the city of a given employee we would need to parse the string of location

Solution:

Assuming location is only in Canada

Split Location into 4 attributes:

- address_line
 - Street names and numbers are complicated with different units and subunits so the culmination can be put here as as string
- city
- province
- postalcode
 - Although province can most likely be found using the postalcode it is not a guarantee
 - Assuming a string a length 6 without the dash

Start and End date are not atomic

Since we are ignoring dates this does not matter

Potential Solution

Split Dates into the following:

- start_date into
 - O start_date_year
 - O start_date_month
 - O start_date_day
- end_date into
 - O end_date_year
 - O end_date_month
 - O end_date_day

Splitting Role Level

Although role level can technically be split into L and then the level number, it does not reduce the complexity because L is always in front a a level number. If other first letters were possible such as M1 then we would split the role_level into 2 attributes

New Relational Schema Diagram

Employee

- empID INT(11)
- emp_fname VARCHAR(20)
- emp_initials VARCHAR(3) emp_lname VARCHAR(20)
- job VARCHAR(100)
- deptID INT(11)
- salary INT(11)

Project

- projID INT(11)
- title VARCHAR(100) phase VARCHAR(20)
- budget DECIMAL(10.2) funds DECIMAL(10,2)

Assigned

- empID INT(11) projID INT(11)
- roleID INT(11)
- role_description VARCHAR(100)
- role_level VARCHAR(100)
- state_date DATE
- end_date DATE assigned_status VARCHAR(20)

Department

- deptID INT(11)
- deptName VARCHAR(100)
- address_line VARCHAR(100) city VARCHAR(100)
- province VARCHAR(100) postalcode VARCHAR(6)

B1) Is the 1NF Schema in 3NF?

Checking 3NF

Functional Dependencies

Department $F = \{ \text{deptID} \rightarrow \text{deptID}, \text{location} \}$

 $= \{ deptID \rightarrow deptID, address_line, city, province, postalcode \}$

 $F = \{\text{projID} \rightarrow \text{title}, \text{phase}, \text{budget}, \text{funds}\}\$ **Project**

 $F = \{\text{empID} \rightarrow \text{emp_fname}, \text{emp_initials}, \text{emp_lname}, \text{job}, \text{deptID}, \text{salary}\}$ **Employee**

Assigned

```
F = \{\text{empID}, \text{projID}, \text{roleID} \rightarrow \text{role\_description}, \text{role\_level}, \text{start\_date}, \text{end\_date}, \text{assigned\_status}, \}
                                    roleID \rightarrow role\_description, role\_level
                                                                                                                                                                    }
```

Since Department Project and Employee FD all have their primary key on the left hand side. Thus these tables pass 3NF

For Assigned, roleto is not a super key because it cannot identify start_date, end_date, or assigned_status thus because of FD roleID \rightarrow role_description, role_level, Assigned is not in 3NF

3NF Decomposition

We need to decompose Assigned to be in 3NF

```
R = (\text{empID}, \text{projID}, \text{roleID})
```

role_description, role_level, start_date, end_date, assigned_status)

```
F = \{\text{empID}, \text{projID}, \text{roleID} \rightarrow \text{role\_description}, \text{role\_level}, \text{start\_date}, \text{end\_date}, \text{assigned\_status}, \}
                                    roleID \rightarrow role\_description, role\_level
                                                                                                                                                                    }
```

Finding Canonical Cover

```
Proving role_description and role_level is extraneous in:
empID, projID, roleID \rightarrow role_description, role_level, start_date, end_date, assigned_status
Let:
F' = \{\text{empID}, \text{projID}, \text{roleID} \rightarrow \text{start\_date}, \text{end\_date}, \text{assigned\_status},
                              roleID \rightarrow role\_description, role\_level
                                                                                               }
                empID, projID, roleID \rightarrow roleID
                                                                                              {relexivity}
                                                                                                                (1)
                                                                                            \{given in F'\}
                                     roleID \rightarrow role\_description, role\_level
                                                                                                                (2)
                empID, projID, roleID \rightarrow role_description, role_level {transitivity 1,2}
                                                                                                                (3)
                                                                                            {given in F'} (4)
                empID, projID, roleID \rightarrow start\_date,
                                                 end_date,
                                                 assigned\_status
                                                                                              \{union 3,4\} (5)
                empID, projID, roleID \rightarrow role_description,
                                                 role_level,
                                                 start_date,
                                                 end_date,
                                                 assigned_status
Thus:
F_C = \{\text{empID}, \text{projID}, \text{roleID} \rightarrow \text{start\_date}, \text{end\_date}, \text{assigned\_status}, \}
                              roleID \rightarrow role\_description, role\_level
                                                                                               }
```

Decomposition

So we can split the relation into

 $R_1 = (\text{empID}, \text{projID}, \text{roleID}, \text{start_date}, \text{end_date}, \text{assigned_status})$

 $R_2 = (\text{roleID}, \text{role_description}, \text{role_level})$

Since both these relations only have super keys on the left of the FD they are both in 3NF Let R_1 be named Assigned and R_2 be Roles

New Relational Schema Diagram

	Employee
F	empID INT(11)
	emp_fname VARCHAR(20)
	emp_initials VARCHAR(3)
1-	emp_lname VARCHAR(20)
-	job VARCHAR(100)
-	deptID INT(11)
Ŀ	salary INT(11)

	Project
-	projID INT(11)
-	title VARCHAR(100)
-	phase VARCHAR(20)
	budget DECIMAL(10,2)
-	funds DECIMAL(10,2)

	Assign	ed
	empID INT(11)	
	projID INT(11)	
-	roleID INT(11)	
-	state_date DATE	
	end_date DATE	
-	assigned_status	VARCHAR(20)

	Depar emerie
-	deptID INT(11)
-	deptName VARCHAR(100)
	address_line VARCHAR(100)
	city VARCHAR(100)
	province VARCHAR(100)
-	postalcode VARCHAR(6)



B2) Is the 1NF Schema in BCNF?

Checking BCNF

```
Functional Dependencies
```

```
\textbf{Department} \hspace{1cm} F = \{ \text{deptID} \rightarrow \text{deptID}, \text{location} \}
```

 $= \{ \text{deptID} \rightarrow \text{deptID}, \text{address_line}, \text{city}, \text{province}, \text{postalcode} \}$

Project $F = \{\text{projID} \rightarrow \text{title, phase, budget, funds}\}$

 $F = \{\text{empID} \rightarrow \text{emp_fname, emp_initials, emp_lname, job, deptID, salary}\}$

Assigned

```
F = \{ \text{empID, projID, roleID} \rightarrow \text{role\_description, role\_level, start\_date, end\_date, assigned\_status,} \\ \text{roleID} \rightarrow \text{role\_description, role\_level} \\ \}
```

Since Department Project and Employee FD all have their primary key on the left hand side. Thus these tables pass BCNF For Assigned, role10 is not a super key because it cannot identify $start_date$, end_date or $assigned_status$ thus because of FD $role10 \rightarrow role_description$, $role_level$, Assigned is not in BCNF

BCNF Decomposition

```
R = (empID, projID, roleID role_description, role_level, start_date, end_date, assigned_status)
```

```
F = \{ \text{empID, projID, roleID} \rightarrow \text{role\_description, role\_level, start\_date, end\_date, assigned\_status,} \\ \text{roleID} \rightarrow \text{role\_description, role\_level} \\ \}
```

We already know that this relations is not in BCNF

The functional dependency:

```
roleID \rightarrow role\_description, role\_level
```

is a nontrivial FD that holds on R and roleID is not a super key for R since $(roleID)^+ = \{roleID, role_description, role_level\}$

So we will split R into:

 $R - \{\text{role_description, role_level}\} = R_1(\text{empID, projID, roleID, start_date, end_date, assigned_status})$

 R_2 (roleID, role_description, role_level)

Checking R_1 and R_2 for BCNF

```
R_1 is in BCNF since <code>empID</code>, <code>projID</code>, <code>roleID</code> is a super key for R_1 (empID, projID, roleID)^+ = \{empID, projID, roleID \\ role\_description, role\_level, start\_date, end\_date, assigned\_status\}
```

 R_2 is in BCNF since ${}^{
m roleID}$ is a super key for R_2

 $\bullet \ \ (\mathrm{role})^+ = \{\mathrm{roleID}, \mathrm{role_description}, \mathrm{role_level}\}$

Let R_1 be named Assigned and R_2 be Roles

New Relational Schema Diagram

	Employee
-	empID INT(11)
-	emp_fname VARCHAR(20)
l-	emp_initials VARCHAR(3)
	emp_lname VARCHAR(20)
ŀ	job VARCHAR(100)
ŀ	deptID INT(11)
I-	salary INT(11)

Project	Assigned
ojID INT(11) tle VARCHAR(100) ase VARCHAR(20) dget DECIMAL(10,2) nds DECIMAL(10,2)	- empID INT(11) - projID INT(11) - roleID INT(11) - state_date DATE - end_date DATE - assigned_status VARCHAR(20)

	Depar emerie
-	deptID INT(11)
-	deptName VARCHAR(100)
-	address_line VARCHAR(100)
-	city VARCHAR(100)
-	province VARCHAR(100)
-	postalcode VARCHAR(6)

- <u>roleID INT(11)</u> - role_description VARCHAR(100) - role_level VARCHAR(100)