

CSCI 3901

**Software Development Concepts**



**DALHOUSIE**  
**UNIVERSITY**

Faculty of Computer Science

**Lab 9: “Database Design”**

Kishan Kahodariya

B00864907

Dhruv Patel

B00868931

## Part – 01 Key Constraints

**Make following changes to the databases:**

- (a) Add an entry to course with web id of NULL**
  - insert into course value (4," csci4521", null);
- (b) Add an entry to course with web id of 2**
  - insert into course value (5," csci4420", 2);
- (c) Add an entry to course with web id of 4**
  - insert into course value (6," csci4720", 4);
- (d) Add an entry to web with web id of 5**
  - insert into web value (5, " www.slack.com");

**Explain how you would identify all the entries with bad web\_id keys.**

- Select c.web\_id from course c left join web w on c.web\_id=w.web\_id where w.web\_id is null AND c.web\_id is not null;

**Determine whether the foreign key constraint can be successfully added with the above command in each of the following cases:**

- a) course contains a web id that is a bad foreign key**
  - It fails because foreign key constraint fails
- b) course contains a web id that is NULL**
  - It will be successfully added
- c) course contains only web ids that are in web**
  - It will be successfully added

**Add web id as a foreign key in course, making any changes to either table necessary to do so.**

- You can either update or either delete the bad web\_id foreign key and then add it to course table.
  1. update web set web\_id=4 where web\_id=5;
  2. alter table course add foreign key (web\_id) references web (web\_id)

### **Explain what it means if you allow a foreign key column to be NULL**

- Course table will allow adding a column as foreign\_key having NULL from other table if that column is allowed to have NULL value.

### **Explain what it means if you do not allow a foreign key column to be NULL**

- For every course there exists web\_id in web table which means no new course can be added unless there is corresponding web\_id.

### **Make the following changes to the database and report on their success or failure:**

- Delete course key 2 in course**
  - delete from course where course\_key=2
  - This query will succeed
- Delete web id 3 in web**
  - delete from web where web\_id=3
  - This query will not succeed
- Delete web (i.e. using drop table web;)**
  - drop table web;
  - This query will not succeed
- Delete course (i.e. using drop table course;)**
  - drop table course;
  - This query will succeed

### **Explain why the previous commands succeeded or failed.**

- This query succeeds as course table doesn't contain any foreign key references.
- This query fails as this will have bad foreign key i.e. web\_id=3 in course table.
- This query fails as it will also delete foreign key references from course table.
- We can delete course table as it doesn't have any foreign key references from other table.

## Part – 02 Database Design from ERD

### SQL Query

```
create table ds_EVENT(  
    Event_ID int primary key,  
    Event_Date date,  
    Event_Location varchar(20),  
    Event_Time time  
);
```

```
create table ds_MENU(  
    Menu_ID int primary key,  
    Menu_description varchar(20),  
    Menu_type varchar(20)  
);
```

```
create table ds_DISH(  
    Dish_ID int primary key,  
    Dish_Name varchar(20),  
    Prep_Time time  
);
```

```
create table ds_DISH_Ingredient(  
    Ingredient varchar(20),  
    Dish_id int NOT NULL REFERENCES ds_DISH(Dish_ID) ON DELETE cascade  
);
```

```
describe ds_DISH_Ingredient;
```

```
create table ds_WORK_SCHEDULE(  
    Event_ID int,  
    Emp_ID int,  
    Start_Time time,  
    End_Time time,
```

```
Position time,  
foreign key(Event_ID) references ds_EVENT(Event_ID),  
foreign key(Emp_ID) references ds_STAFF(Emp_ID)  
);
```

```
create table ds_STAFF(  
    Emp_ID int primary key,  
    Supervisor_ID int,  
    Staff_Name varchar(20),  
    Salary int,  
    foreign key(Supervisor_ID) references ds_STAFF(Emp_ID)  
);
```

```
create table ds_STAFF_Skill(  
    Skill varchar(20),  
    Emp_ID int NOT NULL REFERENCES  
    ds_STAFF(Emp_ID) ON DELETE cascade  
);
```

```
alter table ds_WORK_SCHEDULE add foreign key (Event_ID) references  
ds_EVENT(Event_ID);  
alter table ds_WORK_SCHEDULE modify Event_ID int not null;
```

```
alter table ds_WORK_SCHEDULE add foreign key (Emp_ID) references  
ds_STAFF(Emp_ID);  
alter table ds_WORK_SCHEDULE modify Emp_ID int not null;
```

```
alter table ds_EVENT add Menu_ID int;  
alter table ds_EVENT add foreign key (Menu_ID) references ds_MENU(Menu_ID);  
alter table ds_EVENT modify Menu_ID int not null;
```

```
create table ds_Contains(  
    Menu_ID int,  
    Dish_ID int  
);
```

```
alter table ds_Contains add foreign key (Menu_ID) references ds_MENU(Menu_ID);  
alter table ds_Contains modify Menu_ID int not null;
```

```
alter table ds_Contains add foreign key (Dish_ID) references ds_DISH(Dish_ID);  
alter table ds_Contains modify Dish_ID int not null;
```

## Questions

### **How can foreign key constraints help to maintain the integrity of data in your database?**

- Foreign key constraints allow us to develop relationships among tables such as One to One, One to Many, Many to One, Many to Many. This constraint helps us to integrate data as per our requirement

### **Is there only one valid design for a database with a given ER diagram?**

- No there can be multiple design according to what user interprets from ERD. For example, in our ERD we know that Dish contains multi valued attribute Ingredient but is this attribute one to many or many to many is not specified, these can create ambiguity as many ingredients may also present in dish