CSIT115 Data Management and Security

SQL - Data Definition Language (DDL)

Dr Janusz R. Getta

School of Computing and Information Technology -
University of Wollongong

SQL - Data Definition Statements

Outline

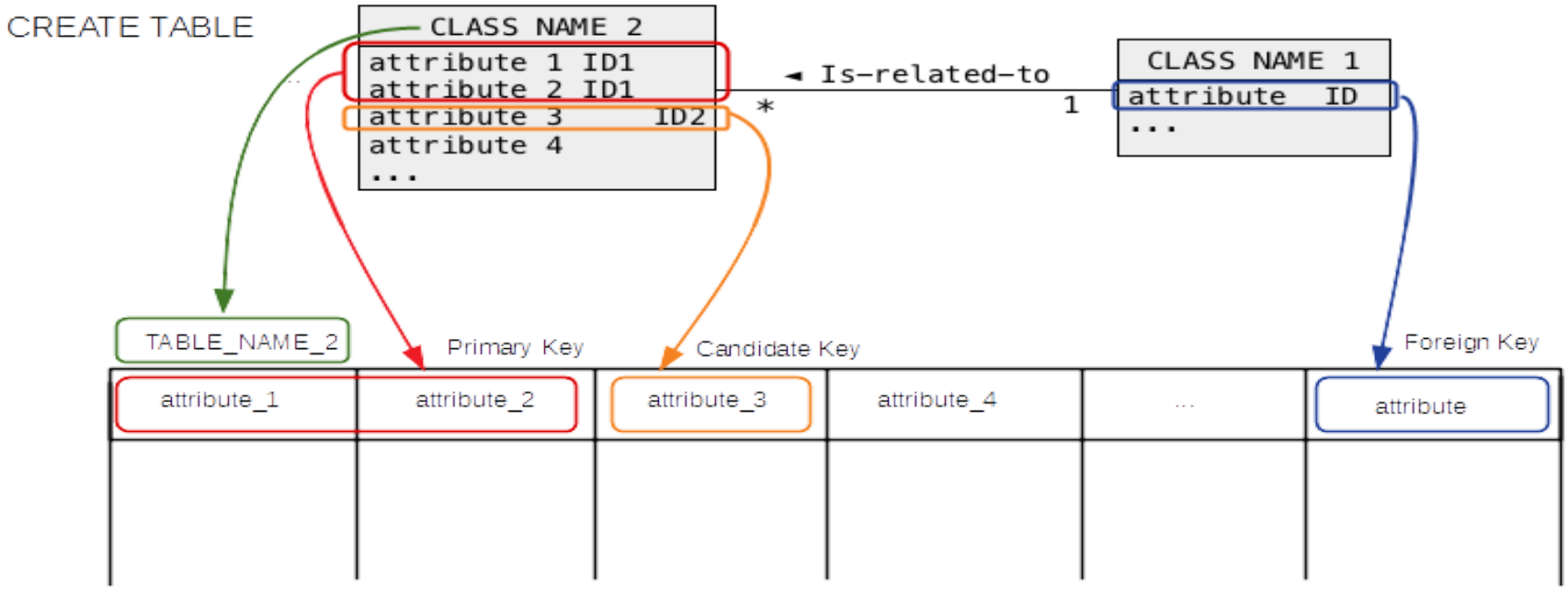
`CREATE TABLE` statement

`DROP TABLE` statement

`ALTER TABLE` statement

CREATE TABLE statement

CREATE TABLE statement creates a new relational table with a given name, given attribute names and types, and with the given logical consistency constraints



CREATE TABLE statement

Example:

```
DEPARTMENT                                                    Schema name
name | code | total staff | chair | budget                   Attribute names
```

- `name`: primary key, variable size string, no more than 50 characters, mandatory
- `code`: candidate key, fixed size string, precisely 5 characters
- `total staff`: total staff number, integer, range 1..50, mandatory
- `chair`: chaiperson, variable size string, no more than 50 characters, optional
- `budget`: real number, no more than 9 digits, one position after decimal dot, mandatory

CREATE TABLE statement

Example:

```
CREATE TABLE DEPARTMENT(                                                    Table name
name                VARCHAR(50)  NOT NULL,                                  Attribute name, type, and constraint
code                CHAR(5)      NOT NULL,                                  Attribute name, type, and constraint
total_staff_number DECIMAL(2)    NOT NULL,                                  Attribute name, type, and constraint
chair               VARCHAR(50)      NULL,                                  Attribute name, type, and constraint
budget              DECIMAL(9,1) NOT NULL,                                  Attribute name, type, and constraint
CONSTRAINT dept_pkey PRIMARY KEY(name),                                     Primary key constraint
CONSTRAINT dept_ckey1 UNIQUE(code),                                         Candidate key constraint
CONSTRAINT dept_ckey2 UNIQUE(chair),                                        Candidate key constraint
CONSTRAINT dept_check1                                                      Domain constraint
CHECK (total_staff_number BETWEEN 1 AND 50) );
```

CREATE TABLE statement

Table name

```
CREATE TABLE DEPARTMENT(                                    Table name
```

Attribute names

```
                                                            Attribute names
name                VARCHAR(50)         NOT NULL,
code                CHAR(5)             NOT NULL,
total_staff_number  DECIMAL(2)          NOT NULL,
chair               VARCHAR(50)             NULL,
budget              DECIMAL(9,1)        NOT NULL,
```

Attribute types

```
                                                            Attribute types
name                VARCHAR(50)         NOT NULL,
code                CHAR(5)             NOT NULL,
total_staff_number  DECIMAL(2)          NOT NULL,
chair               VARCHAR(50)             NULL,
budget              DECIMAL(9,1)        NOT NULL,
```

CREATE TABLE statement

NULL/NOT NULL constraints

NULL/NOT NULL constraints

```
name                VARCHAR(50)        NOT NULL,
code                CHAR(5)            NOT NULL,
total_staff_number  DECIMAL(2)         NOT NULL,
chair               VARCHAR(50)            NULL,
budget              DECIMAL(9,1)       NOT NULL,
```

Primary key constraint

Primary key constraint

```
CONSTRAINT dept_pkey PRIMARY KEY(name),
```

Candidate key constraints

Candidate key constraint

```
CONSTRAINT dept_ckey1 UNIQUE(code),
CONSTRAINT dept_ckey2 UNIQUE(chair),
```

Domain constraint

Domain constraint

```
CONSTRAINT dept_check1 CHECK (total_staff_number BETWEEN 1 AND 50) );
```

CREATE TABLE statement

"Bird's eye view"

CREATE TABLE statement

```
CREATE TABLE DEPARTMENT(
 name                VARCHAR(50)        NOT NULL,
 code                CHAR(5)            NOT NULL,
 total_staff_number  DECIMAL(2)         NOT NULL,
 chair               VARCHAR(50)            NULL,
 budget              DECIMAL(9,1)       NOT NULL,
  CONSTRAINT dept_pkey PRIMARY KEY(name),
  CONSTRAINT dept_ckey1 UNIQUE(code),
  CONSTRAINT dept_ckey2 UNIQUE(chair),
  CONSTRAINT dept_check1 CHECK (total_staff_number BETWEEN 1 AND 50) );
```

CREATE TABLE statement

Another example

COURSE	Schem name
cnum | title | credits | offered_by	Attributes

- `cnum`: primary key, fixed size string, 7 characters
- `title`: candidate key, variable size string. no longer than 100 characters, mandatory
- `credits`: integer, either 6 or 12, mandatory
- `offered_by`: foreign key, references department name, variable size string, no longer than 50 characters, optional

CREATE TABLE statement

Another example

```
CREATE TABLE COURSE(                                                    Table name
cnum        CHAR(7)      NOT NULL,                                      Attribute name, type, and constraint
title       VARCHAR(200) NOT NULL,                                      Attribute name, type, and constraint
credits     DECIMAL(2)   NOT NULL,                                      Attribute name, type, and constraint
offered_by  VARCHAR(50)      NULL,                                      Attribute name, type, and constraint
CONSTRAINT course_pkey PRIMARY KEY(cnum),                               Primary key constraint
CONSTRAINT course_check1 CHECK (credits IN (6, 12)),                    Domain constraint
CONSTRAINT course_fkey1 FOREIGN KEY(offered_by)                         Foreign key
                        REFERENCES DEPARTMENT(name)ON DELETE CASCADE );
```

CREATE TABLE statement

Table name

```
CREATE TABLE COURSE(                                    Table name
```

Attribute names

```
cnum            CHAR(7)          NOT NULL,              Attribute names
title           VARCHAR(200)     NOT NULL,
credits         DECIMAL(2)       NOT NULL,
offered_by      VARCHAR(50)          NULL,
```

Attribute types

```
cnum            CHAR(7)          NOT NULL,              Attribute types
title           VARCHAR(200)     NOT NULL,
credits         DECIMAL(2)       NOT NULL,
offered_by      VARCHAR(50)          NULL,
```

CREATE TABLE statement

NULL/NOT NULL constraints

```
cnum            CHAR(7)         NOT NULL,
title           VARCHAR(200)    NOT NULL,
credits         DECIMAL(2)      NOT NULL,
offered_by      VARCHAR(50)         NULL,
```

NULL/NOT NULL constraints

Primary key constraint

```
CONSTRAINT course_pkey PRIMARY KEY(cnum),
```

Primary key constraint

Domain constraint

```
CONSTRAINT course_check1 CHECK (credits IN (6, 12)),
```

Domain constraint

Foreign key constraint

```
CONSTRAINT course_fkey1 FOREIGN KEY(offered_by)
                        REFERENCES DEPARTMENT(name) ON DELETE CASCADE );
```

Foreign key constraint

CREATE TABLE statement

"Bird's eye view"

CREATE table statement

```
CREATE TABLE COURSE(
 cnum               CHAR(7)          NOT NULL,
 title              VARCHAR(200)     NOT NULL,
 credits            DECIMAL(2)       NOT NULL,
 offered_by         VARCHAR(50)          NULL,
  CONSTRAINT course_pkey PRIMARY KEY(cnum),
  CONSTRAINT course_check1 CHECK (credits IN (6, 12)),
  CONSTRAINT course_fkey1 FOREIGN KEY(offered_by)
                          REFERENCES DEPARTMENT(name) ON DELETE CASCADE );
```

A clause `ON DELETE CASCADE` means that if a row with a value of primary key referenced by a row with a value of foreign key in another or the same relational table is deleted ...

... then a row with a foreign key in another or the same relational table is automatically deleted

CREATE TABLE statement

Some of the attribute types:

- `VARCHAR(size)` Variable length string, maximum size 65535 bytes
- `CHAR(size)` Fixed length string, maximum size 255 bytes
- `INTEGER` Integer numbers in a range [-2147483648, 2147483647]
- `REAL(M,D)` Real numbers with total M digits and D digits after decimal point
- `DECIMAL(M)` Integer numbers with total M digits stored with exact precision
- `DECIMAL(M,D)` Real numbers with total M digits and D digits after decimal point stored with exact precision
- `DATE` dates, default entry format 'YYYY-MM-DD'
- `TIME` times, default entry format 'HH:MI:SS;
- `DATETIME` dates and times, default entry format 'YYYY-MM-DD HH:MI:SS'

SQL - Data Definition Statements

Outline

`CREATE TABLE` statement

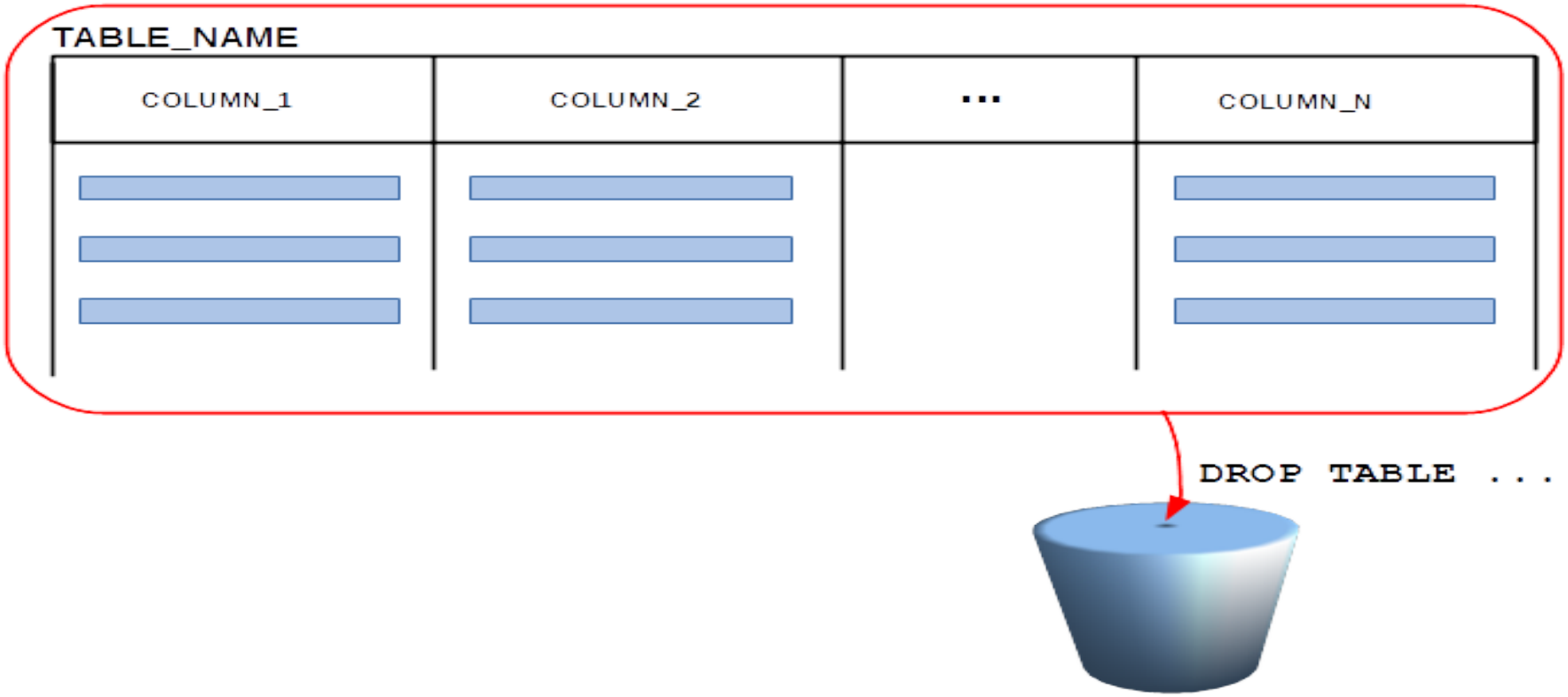
`DROP TABLE` statement

`ALTER TABLE` statement

DROP TABLE statement

Functionality:

- `DROP TABLE` statement permanently deletes the contents of relational table and removes its definition from a database
- A relational table that has been dropped can be recreated as an empty table by the execution of `CREATE TABLE` statement



DROP TABLE statement

Examples:

CREATE TABLE statement

```
CREATE TABLE DEPARTMENT(
 name                VARCHAR(50)        NOT NULL,
 code                CHAR(5)            NOT NULL,
 total_staff_number  DECIMAL(2)         NOT NULL,
 chair               VARCHAR(50)            NULL,
 budget              DECIMAL(9,1)       NOT NULL,
  CONSTRAINT dept_pkey PRIMARY KEY(name),
  CONSTRAINT dept_ckey1 UNIQUE(code),
  CONSTRAINT dept_ckey2 UNIQUE(chair),
  CONSTRAINT dept_check1 CHECK (total_staff_number BETWEEN 1 AND 50) );
```

CREATE TABLE statement

```
CREATE TABLE COURSE(
 cnum                CHAR(7)           NOT NULL,
 title               VARCHAR(200)      NOT NULL,
 credits             DECIMAL(2)        NOT NULL,
 offered_by          VARCHAR(50)           NULL,
  CONSTRAINT course_pkey PRIMARY KEY(cnum),
  CONSTRAINT course_check1 CHECK (credits IN (6, 12)),
  CONSTRAINT course_fkey1 FOREIGN KEY(offered_by)
                       REFERENCES DEPARTMENT(name) ON DELETE CASCADE )
```

DROP TABLE statement

Examples:

```
DROP TABLE COURSE;
```
DROP TABLE statement

```
DROP TABLE DEPARTMENT;
```
DROP TABLE statement

BEWARE !!!

```
DROP TABLE DEPARTMENT;
```
DROP TABLE statement

```
--------------
DROP TABLE DEPARTMENT
--------------

ERROR 1217 (23000): Cannot delete or update a parent row: a foreign key
constraint fails
```
Feedback message

- An order in which the relational tables are dropped is important !!!

SQL - Data Definition Statements

Outline

`CREATE TABLE` statement

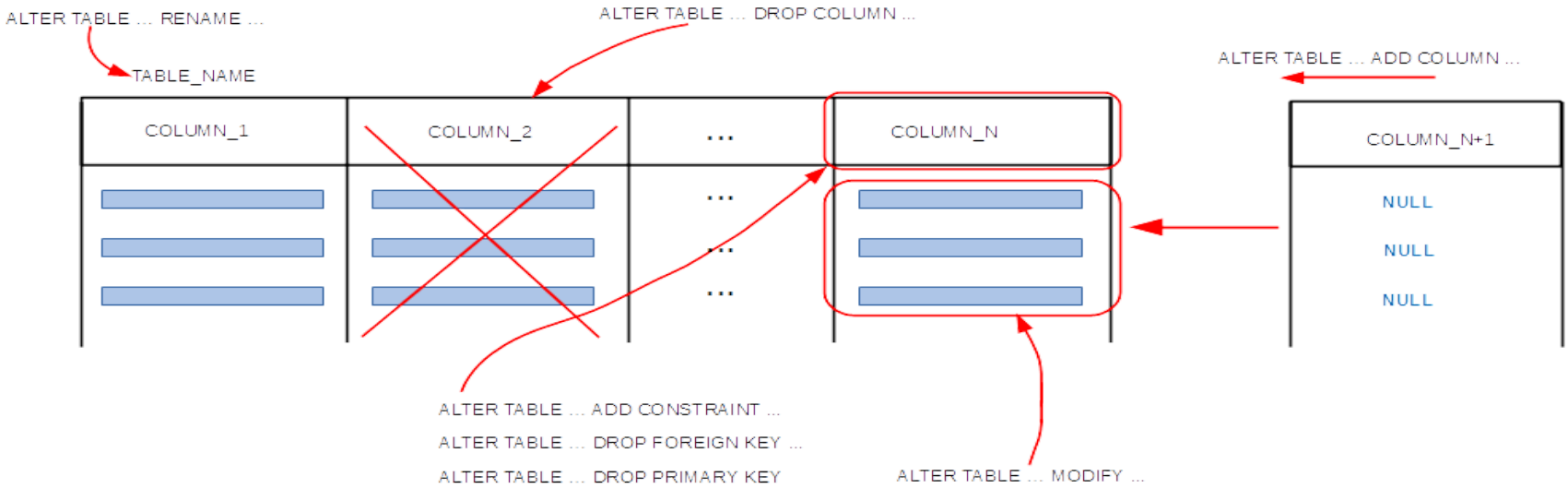
`DROP TABLE` statement

`ALTER TABLE` statement

ALTER TABLE statement

Functionality

- ALTER TABLE statement permanently changes a definition of relational table
- ALTER TABLE statement can be used to:
  - add, drop a column,
  - modify a type of column,
  - add, drop a consistency constraint,
  - rename a relational table



ALTER TABLE statement

Adding the attributes

CREATE TABLE statement

```
CREATE TABLE DEPARTMENT(
 name                VARCHAR(50)        NOT NULL,
 code                CHAR(5)            NOT NULL,
 total_staff_number  DECIMAL(2)         NOT NULL,
 chair               VARCHAR(50)            NULL,
 budget              DECIMAL(9,1)       NOT NULL,
  CONSTRAINT dept_pkey PRIMARY KEY(name),
  CONSTRAINT dept_ckey1 UNIQUE(code),
  CONSTRAINT dept_ckey2 UNIQUE(chair),
  CONSTRAINT dept_check1 CHECK (total_staff_number BETWEEN 1 AND 50) );
```

ALTER TABLE statement that adds an attribute

```
ALTER TABLE DEPARTMENT ADD COLUMN category VARCHAR(20);
```

ALTER TABLE statement that adds an attribute

```
ALTER TABLE DEPARTMENT ADD COLUMN vision_stmt VARCHAR(5000);
```

ALTER TABLE statement

Dropping an attribute

CREATE TABLE statement

```
CREATE TABLE DEPARTMENT(
 name                VARCHAR(50)        NOT NULL,
 code                CHAR(5)            NOT NULL,
 total_staff_number  DECIMAL(2)         NOT NULL,
 chair               VARCHAR(50)            NULL,
 budget              DECIMAL(9,1)       NOT NULL,
  CONSTRAINT dept_pkey PRIMARY KEY(name),
  CONSTRAINT dept_ckey1 UNIQUE(code),
  CONSTRAINT dept_ckey2 UNIQUE(chair),
  CONSTRAINT dept_check1 CHECK (total_staff_number BETWEEN 1 AND 50) );
```

ALTER TABLE statement that drops an attribute

```
ALTER TABLE DEPARTMENT DROP COLUMN budget;
```

ALTER TABLE statement

Changing a type of attribute

CREATE TABLE statement

```
CREATE TABLE DEPARTMENT(
 name                VARCHAR(50)        NOT NULL,
 code                CHAR(5)            NOT NULL,
 total_staff_number  DECIMAL(2)         NOT NULL,
 chair               VARCHAR(50)            NULL,
 budget              DECIMAL(9,1)       NOT NULL,
  CONSTRAINT dept_pkey PRIMARY KEY(name),
  CONSTRAINT dept_ckey1 UNIQUE(code),
  CONSTRAINT dept_ckey2 UNIQUE(chair),
  CONSTRAINT dept_check1 CHECK (total_staff_number BETWEEN 1 AND 50) );
```

ALTER TABLE statement that modifies a domain constraint

```
ALTER TABLE DEPARTMENT MODIFY code CHAR(6) NOT NULL;
```

ALTER TABLE statement that modifies NULL/NOT NULL constraint

```
ALTER TABLE DEPARTMENT MODIFY chair VARCHAR(80) NOT NULL;
```

ALTER TABLE statement

Adding a constraint

CREATE TABLE statement

```
CREATE TABLE DEPARTMENT(
 name                VARCHAR(50)        NOT NULL,
 code                CHAR(5)            NOT NULL,
 total_staff_number  DECIMAL(2)         NOT NULL,
 chair               VARCHAR(50)            NULL,
 budget              DECIMAL(9,1)       NOT NULL,
  CONSTRAINT dept_pkey PRIMARY KEY(name),
  CONSTRAINT dept_ckey1 UNIQUE(code),
  CONSTRAINT dept_ckey2 UNIQUE(chair),
  CONSTRAINT dept_check1 CHECK (total_staff_number BETWEEN 1 AND 50) );
```

ALTER TABLE statement that adds a domain constraint

```
ALTER TABLE DEPARTMENT ADD CONSTRAINT dept_check2
                       CHECK (code = UPPER(code));
```

ALTER TABLE statement

Dropping a constraint

CREATE TABLE statement

```
CREATE TABLE DEPARTMENT(
 name               VARCHAR(50)        NOT NULL,
 code               CHAR(5)            NOT NULL,
 total_staff_number DECIMAL(2)         NOT NULL,
 chair              VARCHAR(50)            NULL,
 budget             DECIMAL(9,1)       NOT NULL,
  CONSTRAINT dept_pkey PRIMARY KEY(name),
  CONSTRAINT dept_ckey1 UNIQUE(code),
  CONSTRAINT dept_ckey2 UNIQUE(chair),
  CONSTRAINT dept_check1 CHECK (total_staff_number BETWEEN 1 AND 50) );
```

CREATE TABLE statement

```
CREATE TABLE COURSE(
 cnum               CHAR(7)           NOT NULL,
 title              VARCHAR(200)      NOT NULL,
 credits            DECIMAL(2)        NOT NULL,
 offered_by         VARCHAR(50)           NULL,
  CONSTRAINT course_pkey PRIMARY KEY(cnum),
  CONSTRAINT course_check1 CHECK (credits IN (6, 12)),
  CONSTRAINT course_fkey1 FOREIGN KEY(offered_by)
                       REFERENCES DEPARTMENT(name) ON DELETE CASCADE )
```

ALTER TABLE statement that drops a foreign key

```
ALTER TABLE COURSE DROP FOREIGN KEY course_fkey1;
```

ALTER TABLE statement that drops a primary key

```
ALTER TABLE DEPARTMENT DROP PRIMARY KEY;
```

ALTER TABLE statement

Renaming a relational table

CREATE TABLE statement

```
CREATE TABLE DEPARTMENT(
 name                VARCHAR(50)        NOT NULL,
 code                CHAR(5)            NOT NULL,
 total_staff_number DECIMAL(2)          NOT NULL,
 chair               VARCHAR(50)            NULL,
 budget              DECIMAL(9,1)       NOT NULL,
  CONSTRAINT dept_pkey PRIMARY KEY(name),
  CONSTRAINT dept_ckey1 UNIQUE(code),
  CONSTRAINT dept_ckey2 UNIQUE(chair),
  CONSTRAINT dept_check1 CHECK (total_staff_number BETWEEN 1 AND 50) );
```

ALTER TABLE statement that renames a table

```
ALTER TABLE DEPARTMENT RENAME TO NEWDEPARTMENT;
```

References

T. Connoly, C. Begg, Database Systems, A Practical Approach to Design, Implementation, and Management, Chapters 7.1, 7.2, 7.3 (except 7.3.5, 7.3.6) SQL: Data Definition, Pearson Education Ltd, 2015

D. Darmawikarta, SQL for MySQL A Beginner's Tutorial, Chapter 1, pages 5-8, Brainy Software Inc. First Edition: June 2014

How to ... ? Cookbook, How to use data definition and basic data manipulation statements of SQL ? Recipe 4.1 How to create and how to alter the relational tables ?