**10/14/2019**

CREATE TABLE IF NOT EXISTS Department (

 deptCode VARCHAR(4),

 deptName VARCHAR(30) NOT NULL, --

 schoolCode VARCHAR(3),

 numFaculty TINYINT,

 CONSTRAINT Department\_deptCode\_pk PRIMARY KEY (deptCode),

 CONSTRAINT Department\_name\_ck UNIQUE (deptName), -- CK: unique not null

 CONSTRAINT Department\_schoolCode\_fk FOREIGN KEY (schoolCode)

 REFERENCES School(schoolCode)

);

<https://dev.mysql.com/doc/refman/8.0/en/integer-types.html>

**Table 11.1 Required Storage and Range for Integer Types Supported by MySQL**

| **Type** | **Storage (Bytes)** | **Minimum Value Signed** | **Minimum Value Unsigned** | **Maximum Value Signed** | **Maximum Value Unsigned** |
| --- | --- | --- | --- | --- | --- |
| TINYINT | 1 | -128 | 0 | 127 | 255 |
| SMALLINT | 2 | -32768 | 0 | 32767 | 65535 |
| MEDIUMINT | 3 | -8388608 | 0 | 8388607 | 16777215 |
| INT | 4 | -2147483648 | 0 | 2147483647 | 4294967295 |
| BIGINT | 8 | -263 | 0 | 263-1 | 264-1 |

mysql> desc department;

+------------+-------------+------+-----+---------+-------+

| Field | Type | Null | Key | Default | Extra |

+------------+-------------+------+-----+---------+-------+

| deptCode | varchar(4) | NO | PRI | NULL | |

| deptName | varchar(30) | YES | UNI | NULL | |

| schoolCode | varchar(3) | YES | MUL | NULL | |

| numFaculty | tinyint(4) | YES | | NULL | |

+------------+-------------+------+-----+---------+-------+

4 rows in set (0.02 sec)

PRI: primary

UNI: Unique
MUL: Multivalued

* DDL:
	1. CREATE TABLE
	2. CREATE DATABASE: a database contains a collection of related tables for an application.
	3. CREATE VIEW
	4. CREATE INDEX
	5. CREATE PROCEDURE
	6. CREATE FUNCTION
	7. CREATE TRIGGER
	8. ALTER (No ALTER TRIGGER and ALTER INDEX)
	9. DROP
* Constraint: to implement certain constraints in your data model.
	1. NOT NULL: attributes cannot have an null value.
	2. UNIQUE: KEY; the set of attributes must be unique for each row:
	3. PRIMARY KEY: unique, not null, and used for the physical structure of the relation.
	4. FOREIGN KEY
	5. CHECK: for a Boolean conditions on the columns.
	6. DEFAULT: insert a default.

Fall 2018:

(1) The student with id 100001 has received a new grade of A in the class with id 10000.

1.1 The student with id 100001, classId = 10000



stuId = 100001, classId = 10000

UPDATE [LOW\_PRIORITY] [IGNORE] table\_name

SET

    column\_name1 = expr1,

    column\_name2 = expr2,

    ...

[WHERE

    condition];

UPDATE Enroll
SET grade = ‘A’
WHERE stuId = 100001
AND classId = 10000;



mysql> desc faculty; +----------+-------------+------+-----+---------+-------+ | Field | Type | Null | Key | Default | Extra | +----------+-------------+------+-----+---------+-------+ | facId | int(11) | NO | PRI | NULL | | | fname | varchar(20) | NO | | NULL | | | lname | varchar(20) | NO | | NULL | | | deptCode | varchar(4) | NO | MUL | NULL | | | rank | varchar(25) | YES | | NULL | | +----------+-------------+------+-----+---------+-------+ 5 rows in set (0.00 sec)

CREATE TABLE IF NOT EXISTS Faculty (

 facId INT NOT NULL,

 fname VARCHAR(20) NOT NULL,

 lname VARCHAR(20) NOT NULL,

 deptCode VARCHAR(4) NOT NULL,

 `rank` VARCHAR(25),

 CONSTRAINT Faculty\_facId\_pk PRIMARY KEY (facId),

 CONSTRAINT Faculty\_deptCode\_fk FOREIGN KEY (deptCode)

 REFERENCES Department(deptCode));

CREATE TABLE IF NOT EXISTS Department (

 deptCode VARCHAR(4),

 deptName VARCHAR(30),

 schoolCode VARCHAR(3),

 numFaculty TINYINT,

 CONSTRAINT Department\_deptCode\_pk PRIMARY KEY (deptCode),

 CONSTRAINT Department\_name\_ck UNIQUE (deptName),

 CONSTRAINT Department\_schoolCode\_fk FOREIGN KEY (schoolCode)

 REFERENCES School(schoolCode)

);