

1 CDIP-UIU: Python & Data Analysis

2 Batch - 09

3

4 Welcome to the 2nd lecture on MySQL!

5

6

7 -----

8 | Some more DDL! |

9 -----

10

11 Foreign key --> for referring another table

12

13

14 CREATE TABLE IF NOT EXISTS tablename(
15

16 col1 datatype [NOT NULL] [DEFAULT def_val] [AUTO_INCREMENT],
17

18 col2 datatype [NOT NULL] [DEFAULT def_val] [AUTO_INCREMENT],
19

20 ...
21

22 ...
23

24 coln datatype [NOT NULL] [DEFAULT def_val] [AUTO_INCREMENT],
25

26

27

28 CONSTRAINT constraint_name PRIMARY KEY(col1, col2),
29

30 CONSTRAINT constraint_name1 UNIQUE(col3, col10),
31

32

33 CONSTRAINT constraint_name2 FOREIGN KEY(col11, col15)
34

35 REFERENCES ref_tbl_name(ref_tbl_colname1,
36

37 ref_tbl_colname2)
38

39 ON DELETE RESTRICT/CASCADE/SET NULL,
40

41

42 CONSTRAINT constraint_name2 FOREIGN KEY(col13, col20)
43

44 REFERENCES ref_tbl_name1(ref_tbl_colname1,
45

46 ref_tbl_colname2)
47

48 ON DELETE RESTRICT/CASCADE/SET NULL
49

50)
51

52

53

54 ON DELETE = RESTRICT --> won't let you delete
55

56 CASCADE --> deletes all child table data (the ones referring)
57

58 SET NULL --> will replace the value with null
59

60

61

62 Students table: (child table)
63

64 =====
65

66

67 -----
68

69 | ID (pk) | name | dob | cgpa | dept (fk) |
70

71 -----
72 | 1 | A | 01.01.95 | 3.50 | 1 | --> a student's info
73

74 -----
75

76 | 2 | B | 04.02.95 | 3.33 | 2 |
77

78 -----
79

80 | 3 | C | 18.02.95 | 3.63 | 3 |
81

82 -----
83

84 | 4 | A | 01.04.95 | 3.46 | 3 |
85

86 -----
87

88

89

90 Departments table: (parent table)
91

92 =====
93

94

95

```

58 -----
59 | ID (pk) | name   | head  | no_of_faculties |
60 -----
61 | 1       | CSE    | Dr. A | 20               |
62 -----
63 | 2       | EEE    | Dr. B | 18               |
64 -----
65 | 3       | BBA    | Dr. C | 22               |
66 -----
67 | 4       | Civil  | Dr. D | 23               |
68 -----
69
70
71
72 CREATE TABLE IF NOT EXISTS students (
73     id INT PRIMARY KEY AUTO_INCREMENT,
74     name VARCHAR(30),
75     dob DATE,
76     admission_date DATE,
77     cgpa FLOAT(3,2),
78     dept INT,
79
80     CONSTRAINT fk_students_departments FOREIGN KEY(dept) REFERENCES departments(id)
81     ON DELETE RESTRICT
82 );
83
84
85 How to update database configuration:
86 -----
87
88 1) how to add new column:
89 -----
90 ALTER TABLE tablename
91 ADD COLUMN colname datatype [PRIMARY KEY] [UNIQUE] [NOT NULL] [DEFAULT def_val]
92     [AUTO_INCREMENT]
93
94 2) how to delete existing column:
95 -----
96 ALTER TABLE tablename
97 DROP COLUMN colname
98
99
100 3) how to delete existing primary key:
101 -----
102 ALTER TABLE tablename
103 DROP PRIMARY KEY
104
105
106 4) how to add new primary key:
107 -----
108 ALTER TABLE tablename
109 ADD CONSTRAINT constraint_name PRIMARY KEY(col1, col2)
110
111
112 5) how to add unique constraint:
113 -----
114 ALTER TABLE tablename
115 ADD CONSTRAINT constraint_name1 UNIQUE(col3, col10)

```

```

116
117
118 6) how to delete existing unique constraint:
119 -----
120 ALTER TABLE tablename
121 DROP INDEX unique_constraint_name
122
123
124 7) how to add new FOREIGN KEY constraint:
125 -----
126 ALTER TABLE tablename
127 ADD CONSTRAINT constraint_name2 FOREIGN KEY(col11, col15)
128                                     REFERENCES ref_tbl_name(ref_tbl_colname1,
ref_tbl_colname2)
129                                     ON DELETE RESTRICT/CASCADE/SET NULL
130
131
132 8) how to delete FOREIGN KEY constraint:
133 -----
134 ALTER TABLE tablename
135 DROP FOREIGN KEY fk_constraint_name
136
137
138 9) how to add DEFAULT value constraint:
139 -----
140 ALTER TABLE tablename
141 ALTER COLUMN colname SET DEFAULT default_value
142
143
144 10) how to delete default condition:
145 -----
146 ALTER TABLE tablename
147 ALTER COLUMN colname DROP DEFAULT
148
149
150 ----- End of DDL -----
151
152
153 =====
154
155
156 -----
157 | Continuation of DML! |
158 -----
159
160
161 Updating data:
162 -----
163 UPDATE table_name
164 SET column1 = value1, column2 = value2, ...
165 WHERE condition;
166
167
168 Calculations inside select:
169 -----
170 SELECT col1 + col2,
171        col1 - col2,
172        col1 * col2,
173        col1 / col2,
174        col1 % col2,

```

```

175 FROM tablename
176
177
178 Aliasing:
179 -----
180 SELECT name AS `Student's Name`,
181         cgpa*100 AS 'CGPA Multiplied by 100'
182 FROM students
183
184
185 SELECT name AS `Student's Name`,
186         id AS 'ID',
187         cgpa AS 'CGPA',
188         (cgpa*100)+id AS 'CGPA Multiplied by 100 plus ID'
189 FROM students
190
191
192 Distinct data:
193 -----
194 Using 'DISTINCT' keyword
195
196 SELECT DISTINCT MANAGER_ID
197 FROM employees
198
199
200 Sorting:
201 -----
202 Using 'ORDER BY' keyword
203
204 - without sorting:
205 SELECT CONCAT(FIRST_NAME, ' ', LAST_NAME) as 'Full Name',
206         SALARY as 'Salary'
207 FROM employees
208
209 - with sorting
210
211 Example 1:
212 SELECT CONCAT(FIRST_NAME, ' ', LAST_NAME) as 'Full Name',
213         SALARY as 'Salary'
214 FROM employees
215 ORDER BY SALARY [DESC|ASC]
216
217 Example 2:
218 SELECT CONCAT(FIRST_NAME, ' ', LAST_NAME) as 'Full Name',
219         SALARY as 'Salary'
220 FROM employees
221 ORDER BY FIRST_NAME ASC
222
223 Example 3: --> (Sorting with multiple parameters)
224 SELECT CONCAT(FIRST_NAME, ' ', LAST_NAME) as 'Full Name',
225         SALARY as 'Salary'
226 FROM employees
227 ORDER BY SALARY ASC, FIRST_NAME DESC
228
229
230 Limiting:
231 -----
232
233 First 5 employees:
234 -----

```

```

235 SELECT EMPLOYEE_ID AS 'ID',
236        CONCAT(FIRST_NAME, ' ', LAST_NAME) as 'Full Name',
237        SALARY as 'Salary'
238 FROM employees
239 LIMIT 5
240
241
242
243 Last 5 employees:
244 -----
245 SELECT EMPLOYEE_ID AS 'ID',
246        CONCAT(FIRST_NAME, ' ', LAST_NAME) as 'Full Name',
247        SALARY as 'Salary'
248 FROM employees
249 ORDER BY EMPLOYEE_ID DESC
250 LIMIT 5
251
252
253 Offset:
254 -----
255 SELECT EMPLOYEE_ID AS 'ID',
256        CONCAT(FIRST_NAME, ' ', LAST_NAME) as 'Full Name',
257        SALARY as 'Salary'
258 FROM employees
259 ORDER BY EMPLOYEE_ID DESC
260 LIMIT 10, 5
261
262 --> LIMIT offset, no_of_rows
263
264
265
266 'LIKE' / String Matching
267 -----
268
269 Exact match
270 WHERE first_name='abc'
271
272 WHERE col = 'value'
273
274
275
276 String pattern match using 'LIKE' operator
277
278 % = 0 to inf character / (star operation) --> RegEx = Regular Expression
279 _ = exactly 1 character / (plus operation) --> RegEx = Regular Expression
280
281
282 Examples of %:
283 -----
284
285 1) %a%
286
287 example: fjdkjfdkjakfdjfdjfk
288         aaaa
289         bab
290         a
291         ba
292         ab
293
294

```

```

295 2) %
296
297 example: any combination of characters including ''
298
299
300 3) %a
301
302 example: fkdfjdkjfkджа
303          fdjdkja
304          ba
305          a
306
307
308 Examples of _:
309 -----
310
311 1) _a_
312
313 example: bab
314          cab
315          aaa
316
317
318 2) _a
319
320 example: ba
321          aa
322          ca
323
324
325 col LIKE 'pattern'
326
327 Example:
328 SELECT EMPLOYEE_ID, FIRST_NAME
329 FROM employees
330 WHERE FIRST_NAME LIKE '%l%'
331
332
333
334 ex: show those employees whose first name starts with 'a'
335 WHERE FIRST_NAME LIKE 'a%'
336
337 ex: name ends with 'a'
338 WHERE FIRST_NAME LIKE '%a'
339
340 ex: name contains 'abc'
341 WHERE FIRST_NAME LIKE '%abc%'
342
343 ex: name consists of 3 characters
344 WHERE FIRST_NAME LIKE '___'
345
346 ex: name contains at least 3 characters
347 WHERE FIRST_NAME LIKE '___%'
348
349
350

```