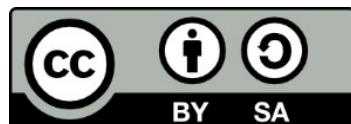


Βάσεις Δεδομένων

Ενότητα # 5: Από το λογικό στο φυσικό (SQL) μοντέλο

Γεώργιος Ευαγγελίδης, Γεωργία Κολωνιάρη
Τμήμα Εφαρμοσμένης Πληροφορικής



Ευρωπαϊκή Ένωση
Ευρωπαϊκό Κοινωνικό Ταμείο



ΥΠΟΥΡΓΕΙΟ ΠΑΙΔΕΙΑΣ & ΘΡΗΣΚΕΥΜΑΤΩΝ, ΠΟΛΙΤΙΣΜΟΥ & ΑΘΛΗΤΙΣΜΟΥ
ΕΙΔΙΚΗ ΥΠΗΡΕΣΙΑ ΔΙΑΧΕΙΡΙΣΗΣ

Με τη συγχρηματοδότηση της Ελλάδας και της Ευρωπαϊκής Ένωσης



Άδειες Χρήσης

- Το παρόν εκπαιδευτικό υλικό υπόκειται σε άδειες χρήσης Creative Commons.
- Για εκπαιδευτικό υλικό, όπως εικόνες, που υπόκειται σε άλλου τύπου άδειας χρήσης, η άδεια χρήσης αναφέρεται ρητώς.



Χρηματοδότηση

- Το παρόν εκπαιδευτικό υλικό έχει αναπτυχθεί στα πλαίσια του εκπαιδευτικού έργου του διδάσκοντα.
- Το έργο «Ανοικτά Ακαδημαϊκά Μαθήματα στο Πανεπιστήμιο Μακεδονίας» έχει χρηματοδοτήσει μόνο τη αναδιαμόρφωση του εκπαιδευτικού υλικού.
- Το έργο υλοποιείται στο πλαίσιο του Επιχειρησιακού Προγράμματος «Εκπαίδευση και Δια Βίου Μάθηση» και συγχρηματοδοτείται από την Ευρωπαϊκή Ένωση (Ευρωπαϊκό Κοινωνικό Ταμείο) και από εθνικούς πόρους.



Ευρωπαϊκή Ένωση
Ευρωπαϊκό Κοινωνικό Ταμείο



ΕΠΙΧΕΙΡΗΣΙΑΚΟ ΠΡΟΓΡΑΜΜΑ
ΕΚΠΑΙΔΕΥΣΗ ΚΑΙ ΔΙΑ ΒΙΟΥ ΜΑΘΗΣΗ
επένδυση στην κοινωνία της γνώσης
ΥΠΟΥΡΓΕΙΟ ΠΑΙΔΕΙΑΣ & ΘΡΗΣΚΕΥΜΑΤΩΝ, ΠΟΛΙΤΙΣΜΟΥ & ΑΘΛΗΤΙΣΜΟΥ
ΕΙΔΙΚΗ ΥΠΗΡΕΣΙΑ ΔΙΑΧΕΙΡΙΣΗΣ

Με τη συγχρηματοδότηση της Ελλάδας και της Ευρωπαϊκής Ένωσης



ΕΣΠΑ
2007-2013
πρόγραμμα για την ανάπτυξη
ΕΥΡΩΠΑΪΚΟ ΚΟΙΝΩΝΙΚΟ ΤΑΜΕΙΟ

Τύποι δεδομένων

Για τα πεδία ορισμού των γνωρισμάτων (ατομικοί τύποι) έχουμε:

- **Αριθμητικοί**
 - Ακέρατοι (tinyint, smallint, mediumint, int, bigint)
 - Σταθερού μήκους (numeric, decimal)
 - Κινητής υποδιαστολής (float, double)
- **Ημερομηνίες και Ώρες** (data, datetime, timestamp, time, year)
- **Αλφαριθμητικές** (char, varchar, text, enum, binary, set)

Αριθμητικοί τύποι δεδομένων

Ακέραιοι

- TINYINT (1 byte) : Signed [-128, 127] – Unsigned [0, 255]
- SMALLINT (2 bytes) : S [-32768, 32767] – U [0, 65535]
- MEDIUMINT (3 bytes) : S [-8388608, 8388607] – U [0, 16777215]
- INT (4 bytes) : Signed [-2147483648, 2147483647,] – Unsigned [0, 4294967295]
- BIGINT (8 bytes) : Signed [-9223372036854775808, 9223372036854775807] – Unsigned [0, 18446744073709551615]

Σταθερού μήκους

- DECIMAL (ακριβής τύπος): DECIMAL(5, 2) -> 5 ακέραια ψηφία και 2 δεκαδικά

Κινητής Υποδιαστολής

- FLOAT, REAL (προσεγγιστικός τύπος): FLOAT(5,2)-> 5 ψηφία συνολικά από τα οποία το πολύ 2 μπορεί να είναι δεκαδικά
- DOUBLE PRECISION (8 bytes)

Τύποι Ημερομηνίας και Ώρας

- **DATE**: μόνο ημερομηνία χωρίς ώρα
 - Μορφή: 'YYYY-MM-DD'
 - Εύρος: '1000-01-01' - '9999-12-31'.
- **DATETIME**: ημερομηνία και ώρα
 - Μορφή: 'YYYY-MM-DD HH:MM:SS'
 - Εύρος: '1000-01-01 00:00:00' – '9999-12-31 23:59:59'
- **TIMESTAMP**:
 - 'YYYYMMDDHHMMSS'
- **TIME**: μόνο ώρα (HH:MM:SS) Προσέξτε στο format!
- **YEAR**: έτος, YEAR(4) ή YEAR(2)

Αλφαριθμητικοί Τύποι Δεδομένων

- **CHAR:** δηλώνεται μαζί με ένα μέγιστο μήκος, 0 έως 255, π.χ., CHAR(30), σταθερό μέγεθος, γέμισμα με κενά για τις επιπλέον θέσεις (δεν ανακτώνται συνήθως)
- **VARCHAR:** δηλώνεται μαζί με ένα μέγιστο μήκος, 0 έως 65,535, μεταβλητό μέγεθος, 1-2 byte πρόθεμα καθορίζει το μήκος του δεδομένου
- **TEXT:** δεν απαιτεί καθορισμό μέγιστου μεγέθους
- **ENUM:** ένα αλφαριθμητικό αντικείμενο του οποίου η τιμή επιλέγεται από μια λίστα από επιτρεπόμενες τιμές που απαριθμούνται κατά τον ορισμό του γνωρίσματος στη δημιουργία της σχέσης, π.χ. size ENUM('x-small', 'small', 'medium', 'large', 'x-large')

Δημιουργία πινάκων

- Σχήμα πίνακα

CD(code, title, year)

- Εντολή DDL της SQL: CREATE TABLE

**CREATE TABLE table_name(attr1 TYPE1, attr2 TYPE2,...,attrN
TYPE N), CONSTRAINTS, REFERENCES) OPTIONS;**

**CREATE TABLE CD (
code integer,
title varchar(200),
year integer)**

Τροποποίηση σχήματος πινάκων

- Εντολή DDL της SQL: ALTER TABLE

ALTER TABLE CD ADD COLUMN bought DATE

ALTER TABLE CD DROP COLUMN year

ALTER TABLE CD MODIFY COLUMN year DATE

- Εντολή DDL της SQL: DROP TABLE

DROP TABLE CD

Παραδείγματα Τροποποίησης

```
CREATE TABLE Test1(a INT, b CHAR(20));
```

- ALTER TABLE Test1 ADD COLUMN c CHAR(10);
- ALTER TABLE Test1 DROP COLUMN b;
- ALTER TABLE Test1 MODIFY COLUMN a TINYINT;
- ALTER TABLE Test1 RENAME COLUMN c TO b1;
- ALTER TABLE Test1 RENAME TO Test2;

Από το λογικό στο φυσικό (1)

CD(code, title, year)

```
CREATE TABLE CD (  
  code integer primary key,  
  title varchar(200) NOT NULL,  
  year integer)
```

Από το λογικό στο φυσικό (2)

HAS(cdcode, songcode, pos)

Ξένα κλειδιά: cdcode, songcode

Σύνθετο κλειδί: (cdcode, songcode)

```
CREATE TABLE HAS (  
  cdcode integer NOT NULL,  
  songcode integer NOT NULL,  
  pos integer NOT NULL,  
  PRIMARY KEY (cdcode, songcode),  
  FOREIGN KEY (cdcode) REFERENCES CD(code),  
  FOREIGN KEY (songcode) REFERENCES SONG(code))
```

Από το λογικό στο φυσικό (3)

Εναλλακτικά:

```
CREATE TABLE HAS (  
  cdcode integer NOT NULL,  
  songcode integer NOT NULL,  
  pos integer NOT NULL,  
  PRIMARY KEY (cdcode, songcode),  
  CONSTRAINT fk_1 FOREIGN KEY (cdcode)  
    REFERENCES CD(code),  
  CONSTRAINT fk_2 FOREIGN KEY (songcode)  
    REFERENCES SONG(code))
```

Από το λογικό στο φυσικό (4)

Επίσης:

```
ALTER TABLE table_name ADD CONSTRAINT ...  
ALTER TABLE table_name ADD FOREIGN KEY ...  
ALTER TABLE table_name DROP CONSTRAINT ...  
ALTER TABLE table_name DROP FOREIGN KEY ...
```

Για σιγουριά δείτε εγχειρίδιο συγκεκριμένου ΣΔΒΔ

Από το λογικό στο φυσικό (5)

CD(code, title, year)

```
CREATE TABLE CD (  
  code integer,  
  title varchar(200) NOT NULL,  
  year integer);
```

```
ALTER TABLE CD ADD CONSTRAINT pkc PRIMARY KEY (code);
```

```
ALTER TABLE CD DROP pkc;
```

Από το λογικό στο φυσικό (6)

- Περιορισμός συμμετοχής:

CD(cdcode, ..., companycode)

και companycode δεν μπορεί να πάρει τιμή NULL

```
CREATE TABLE CD (  
  cdcode integer PRIMARY KEY,  
  ...  
  companycode integer NOT NULL,  
  FOREIGN KEY companycode  
  REFERENCES COMPANY(code))
```


Από το λογικό στο φυσικό (7)

- Αδύναμο σύνολο οντοτήτων

CD(code, ...)

COPY(code, num, condition)

```
CREATE TABLE COPY (  
  code integer NOT NULL,  
  num integer NOT NULL,  
  condition char(3),  
  PRIMARY KEY (code, num),  
  FOREIGN KEY code REFERENCES CD(code))
```

Περιορισμοί Ακεραιότητας (1)

- Πεδίου ορισμού
- NULL
- UNIQUE
- Κυρίου κλειδιού (PRIMARY KEY)

```
CREATE TABLE STUDENT (  
  AM char(10) NOT NULL PRIMARY KEY,  
  AT char(8) NOT NULL UNIQUE,  
  name varchar(100))
```

Περιορισμοί Ακεραιότητας (2)

- Αναφορική (FOREIGN KEY)
 - Τί συμβαίνει σε περίπτωση διαγραφής/τροποποίησης του κύριου κλειδιού στο οποίο αναφέρεται το ξένο κλειδί;
 - ON DELETE ή ON UPDATE
 - NO ACTION
 - CASCADE
 - SET NULL/SET DEFAULT

Αναφορική ακεραιότητα

```
CREATE TABLE HAS (  
  cdcode integer NOT NULL,  
  songcode integer NOT NULL,  
  pos integer NOT NULL,  
  PRIMARY KEY (cdcode, songcode),  
  FOREIGN KEY (cdcode) REFERENCES CD(code)  
    ON DELETE CASCADE ON UPDATE CASCADE  
  FOREIGN KEY (songcode) REFERENCES SONG(code)  
    ON DELETE CASCADE ON UPDATE CASCADE)
```

Εντολές DML της SQL

```
INSERT INTO CD VALUES (1, 'Adele 21')
```

```
UPDATE CD  
SET title='Adele 22'  
WHERE code=1
```

```
DELETE FROM CD  
WHERE code=1
```

Εισαγωγή Δεδομένων

```
INSERT INTO table_name(attr1, attr2,...attrN) VALUES (v1, v2,...,vN);
```

```
CREATE TABLE Employee(ssn CHAR(15) NOT NULL, name CHAR(20),  
lot INT, PRIMARY KEY(ssn));
```

```
INSERT INTO Employee VALUES ('A12413', 'John', 4);
```

```
INSERT INTO Employee VALUES ('A12563', 'Mary', 17);
```

```
INSERT INTO Employee(ssn, lot, name) VALUES ('B3246', 8, 'Tom');
```

```
INSERT INTO Employee VALUES ('A12582', 'Alice', NULL);
```

```
INSERT INTO Employee(ssn, name) VALUES ('B32467', 'Ben');
```

Προσοχή στους περιορισμούς αναφορικής ακεραιότητας!

Διαγραφή Δεδομένων

`DELETE FROM table_name WHERE condition;`

`DELETE FROM Employee WHERE name='John';`

- Διαγράφει όλες τις πλειάδες που στο πεδίο name έχουν την τιμή 'John'

`CREATE TABLE Test1(a INT PRIMARY KEY, b CHAR(20));`

`DELETE FROM Test1 WHERE a>50;`

Τι γίνεται με τους περιορισμούς αναφορικής ακεραιότητας;

Τροποποίηση Δεδομένων

```
UPDATE table_name SET attr1=expr1, attr2 =expr2,... WHERE  
cond;
```

```
CREATE TABLE Test1(a INT PRIMARY KEY, b CHAR(20));
```

```
UPDATE Test1 SET a=a+100 WHERE a<100;
```

Τι γίνεται με τους περιορισμούς αναφορικής ακεραιότητας;