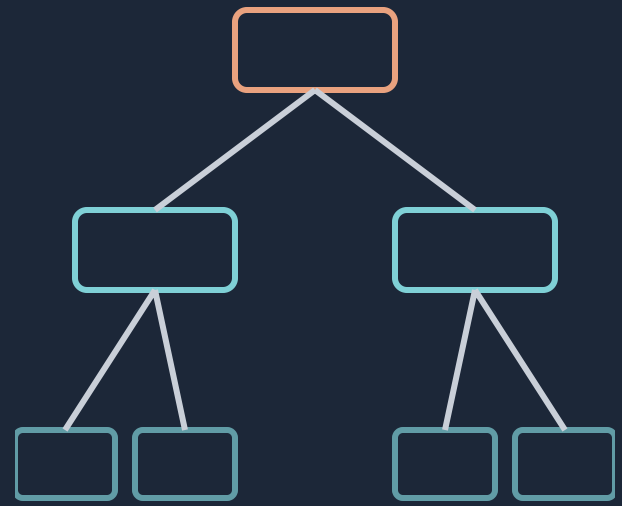
МӘЛІМЕТТЕР ҚОРЫ · ДӘРІС 10

Индекстеу

МҚ объектілерін қосу ерекшеліктері



Оқытушы: Айдана Карибаева

01

Индекс дегеніміз не?

Индекс — деректерді **іздеуді жылдамдату** үшін МҚ қолданатын арнайы құрылым. Кітаптың соңындағы **көрсеткішке** ұқсас.

> Индексті деректердің өзіне әсер етпей **қосуға** немесе **жоюға** болады.

Кітаптағы көрсеткіш

Java	`12, 45, 230`
JDBC	`310`
SQL	`305, 312`

Барлық беттерді парақтаудың орнына — бірден керек бетке.

02

Индекстің бағасы

Жылдамдайды

`SELECT … WHERE` `JOIN` `ORDER BY`

индекссіз — бүкіл кестені қарау

индекспен — бірден табу

Баяулатады

`INSERT` `UPDATE` `DELETE`

Әр өзгерісте индексті де **жаңарту** керек; индекс қосымша **орын** алады.

03 Индексті құру және жою

Қарапайым индекс

```
CREATE INDEX idx_lastname
  ON Persons (LastName);
```

Бірнеше баған бойынша

```
CREATE INDEX idx_pname
  ON Persons (LastName, FirstName);
```

Бірегей индекс — қайталауға жол бермейді

```
CREATE UNIQUE INDEX uq_customers_email
  ON Customers (Email);
```

SQL Server

```
DROP INDEX table_name.index_name;
```

MySQL

```
ALTER TABLE table_name
  DROP INDEX index_name;
```

> Синтаксис МҚБЖ түріне байланысты **өзгеше** болуы мүмкін.

04 Кластерлі және кластерленбеген индекс

Кластерлі (CLUSTERED)

ID ↑	АТЫ
1001	Әлі
1002	Бота
1003	Дана
1004	Ерлан

- кесте жолдары **индекс бойынша сұрыпталады**
- кестеде тек **біреу** ғана болады
- үлкен кестелерді оқуға тиімді

Кластерленбеген

индекс

АТЫ	→
Әлі	#2
Бота	#4
Дана	#3
Ерлан	#1

кесте (үйме)

#	ID	АТЫ
1	1004	Ерлан
2	1001	Әлі
3	1003	Дана
4	1002	Бота

- кестеден **бөлек** сақталады
- әр жазбада жолға көрсеткіш — **жол локаторы**
- бір кестеде бірнеше болуы мүмкін

05

Индекс мысалдары

```
CREATE CLUSTERED INDEX IX_CustomerRecord_YearlyIncome
  ON CustomerRecord ([Yearly Income] ASC);
```

→ «Yearly Income» бағаны бойынша өсу ретімен сұрыпталады

```
CREATE CLUSTERED INDEX ix_clust_employee_id
  ON Employees (EmployeeId, Email);
```

```
CREATE INDEX ix_cars_employee_id
  ON Cars (EmployeeId);
-- жылдамдайды:
SELECT * FROM Cars WHERE EmployeeId = 1;
```

```
CREATE INDEX ix_cars_e_c_o_ids
  ON Cars (EmployeeId, CarId, OwnerId);
```

Құрама индекс шарттар **сол ретпен** берілгенде пайдалы.

06

Сұрыпталған индекс және бірегейлік

Кему ретіндегі индекс

```
CREATE INDEX ix_scoreboard_score
  ON scoreboard (score DESC);

SELECT * FROM scoreboard
ORDER BY score DESC;
```

МҚБЖ **қосымша сұрыптамайды** — индексті сол ретпен оқиды.

Бірегей индекспен жаңарту

```
UPDATE Customers
SET Email = "richard0123@example.com"
WHERE id = 1;
```

Мұндай email бар болса — **қате**. MySQL-де балама: `ON DUPLICATE KEY`.

07

Индексті өшіру және қайта құру

Жою

```
DROP INDEX ix_cars_employee_id
  ON Cars;
```

Өшіру

```
ALTER INDEX ix_cars_employee_id
  ON Cars DISABLE;
```

метадеректер мен статистика сақталады

Қайта құру

```
ALTER INDEX ix_cars_employee_id
  ON Cars REBUILD;
```

08 Индексті қашан қолданбау керек?

× **шағын** кестелерде — әсері байқалмайды

× деректер **жиі қосылатын / өзгеретін** кестелерде

× **жиі өңделетін** бағандарда

× **NULL** мәндері көп бағандарда

✓ Индекс **WHERE**, **JOIN**, **ORDER BY** ішінде жиі қолданылатын бағандарда ең тиімді.

09 Қорытынды

ӘРЕКЕТ	КОМАНДА
Индекс құру	`CREATE INDEX i ON t (c)`
Бірегей	`CREATE UNIQUE INDEX …`
Кластерлі	`CREATE CLUSTERED INDEX …`
Сұрыптау реті	`(c ASC) / (c DESC)`
Өшіру / қайта құру	`ALTER INDEX i ON t DISABLE | REBUILD`
Жою	`DROP INDEX i ON t`