

JPL-Caltech Virtual Summer School

# Big Data Analytics

September 2 – 12, 2014

Matthew J. Graham (Caltech)  
Advanced SQL

last time

---

- | select, joins, aggregate functions

- | create, keys

- | show and describe

- | insert, load data

- | update

- | delete

ALTER TABLE *tableName* ...

ALTER TABLE star ADD COLUMN bmag double AFTER vmag

ALTER TABLE star DROP COLUMN bmag

CREATE VIEW *viewName* AS ...

```
CREATE VIEW region1View AS
  SELECT * FROM star WHERE ra BETWEEN 150 AND 170
    AND dec BETWEEN -10 AND 10
```

```
SELECT id FROM region1View WHERE vmag < 10
```

```
CREATE VIEW region2View AS
  SELECT * FROM star s, stellarTypes t WHERE s.stellarType = t.id
    AND ra BETWEEN 150 AND 170 AND dec BETWEEN -10 AND 10
```

```
SELECT id FROM regionView2 WHERE vmag < 10 and stellarType LIKE 'A%'
```

CREATE INDEX *indexName* ON *tableName*(*columns*)

CREATE INDEX vmagIndex ON star(vmag)

- A clustered index is one in which the ordering of data entries is the same as the ordering of data records
- Only one clustered index per table but multiple unclustered indexes
- Typically implemented as B+ trees but alternate types such as bitmap index for high frequency repeated data

## stored procedures

---

```
CREATE PROCEDURE procedureName @param1 type, ...  
AS ...
```

```
CREATE PROCEDURE findNearestNeighbour @starName varchar(20) AS  
BEGIN  
    DECLARE @ra, @dec float  
    DECLARE @name varchar(20)  
    SELECT @ra = ra, @dec = dec FROM star WHERE name LIKE @starName  
    SELECT name FROM getNearestNeighbour(@ra, @dec)  
END  
  
EXEC findNearestNeighbour 'Sirius'
```

```
DECLARE cursorName CURSOR FOR SELECT ...  
OPEN cursorName  
FETCH cursorName INTO ...  
CLOSE cursorName
```

- | A cursor is a control structure for successive traversal of records in a result set

- | Slowest way of accessing data



## cursors example

---

For each row in the result set, update the relevant stellar model

```
DECLARE @name varchar(20)
DECLARE @mag float
DECLARE starCursor CURSOR FOR
    SELECT name, AVG(vmag) FROM star
    GROUP BY stellarType
OPEN starCursor
    FETCH starCursor INTO @name, @mag
    EXEC updateStellarModel @name, @mag / CALL updateStellarModel(@name, @mag)
CLOSE starCursor
```



# triggers

---

```
CREATE TRIGGER triggerName ON  
tableName ...
```

A trigger is procedural code that is automatically executed in response to certain events on a particular table:

- INSERT
- UPDATE
- DELETE



```
CREATE TRIGGER starTrigger ON star FOR UPDATE AS  
    IF @@ROWCOUNT = 0 RETURN  
    IF UPDATE (vmag) EXEC refreshModels  
GO
```

*matthew graham*

---

## Python:

```
import MySQLdb
Con = MySQLdb.connect(host="127.0.0.1", port=1234, user="mjpg",
    passwd="mjpg", db="test")
Cursor = Con.cursor()
sql = "SELECT * FROM star"
Cursor.execute(sql)
Results = Cursor.fetchall()
...
Con.close()
```