1) Which is not an aggregate function?

A. SUM()
B. COUNT()
C. AVG()
D. MIN()
E. UPPER()
2) What is the output from Q1 when run on the table shown?

Q1: SELECT COUNT(*) FROM Harlem_Globetrotters
WHERE pos IN ('G', 'F')

Harlem_Globetrotters

<table>
<thead>
<tr>
<th>no</th>
<th>name</th>
<th>height</th>
<th>pos</th>
<th>school</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Lili &quot;Champ&quot; Thompson</td>
<td>1.75</td>
<td>G</td>
<td>Notre Dame State</td>
</tr>
<tr>
<td>2</td>
<td>Carlis &quot;Dizzy&quot; English</td>
<td>1.70</td>
<td>G</td>
<td>Cleveland State</td>
</tr>
<tr>
<td>3</td>
<td>Tay &quot;Firefly&quot; Fisher</td>
<td>1.75</td>
<td>G</td>
<td>Siena</td>
</tr>
<tr>
<td>6</td>
<td>Brianna &quot;Hoops&quot; Green</td>
<td>1.75</td>
<td>G</td>
<td>UT - El Paso</td>
</tr>
<tr>
<td>31</td>
<td>Donte &quot;Hammer&quot; Harrison</td>
<td>2.06</td>
<td>F</td>
<td>Hampton</td>
</tr>
<tr>
<td>15</td>
<td>Brittany &quot;Ice&quot; Hrynko</td>
<td>1.85</td>
<td>G</td>
<td>Montana</td>
</tr>
</tbody>
</table>

A. 1  
B. 5  
C. 6  
D. 0
3) What is the output from Q2 when run on the table shown?

Q2: SELECT MAX(height) FROM Harlem_Globetrotters 
    WHERE pos = 'G'

Harlem_Globetrotters

<table>
<thead>
<tr>
<th>no</th>
<th>name</th>
<th>height</th>
<th>pos</th>
<th>school</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Lili &quot;Champ&quot; Thompson</td>
<td>1.75</td>
<td>G</td>
<td>Notre Dame State</td>
</tr>
<tr>
<td>2</td>
<td>Carlis &quot;Dizzy&quot; English</td>
<td>1.70</td>
<td>G</td>
<td>Cleveland State</td>
</tr>
<tr>
<td>3</td>
<td>Tay &quot;Firefly&quot; Fisher</td>
<td>1.75</td>
<td>G</td>
<td>Siena</td>
</tr>
<tr>
<td>6</td>
<td>Brianna &quot;Hoops&quot; Green</td>
<td>1.75</td>
<td>G</td>
<td>UT - El Paso</td>
</tr>
<tr>
<td>31</td>
<td>Donte &quot;Hammer&quot; Harrison</td>
<td>2.06</td>
<td>F</td>
<td>Hampton</td>
</tr>
<tr>
<td>15</td>
<td>Brittany &quot;Ice&quot; Hrynko</td>
<td>1.85</td>
<td>G</td>
<td>Montana</td>
</tr>
</tbody>
</table>

A. 1.75
B. 1.85
C. 2.06
4) What is the output from Q3 when run on the table shown?

**Q3**: SELECT SUM(height) FROM Harlem_Globetrotters
WHERE pos = 'F'

<table>
<thead>
<tr>
<th>no</th>
<th>name</th>
<th>height</th>
<th>pos</th>
<th>school</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Lili &quot;Champ&quot; Thompson</td>
<td>1.75</td>
<td>G</td>
<td>Notre Dame State</td>
</tr>
<tr>
<td>2</td>
<td>Carlis &quot;Dizzy&quot; English</td>
<td>1.70</td>
<td>G</td>
<td>Cleveland State</td>
</tr>
<tr>
<td>3</td>
<td>Tay &quot;Firefly&quot; Fisher</td>
<td>1.75</td>
<td>G</td>
<td>Siena</td>
</tr>
<tr>
<td>6</td>
<td>Brianna &quot;Hoops&quot; Green</td>
<td>1.75</td>
<td>G</td>
<td>UT - El Paso</td>
</tr>
<tr>
<td>31</td>
<td>Donte &quot;Hammer&quot; Harrison</td>
<td>2.06</td>
<td>F</td>
<td>Hampton</td>
</tr>
<tr>
<td>15</td>
<td>Brittany &quot;Ice&quot; Hrynko</td>
<td>1.85</td>
<td>G</td>
<td>Montana</td>
</tr>
</tbody>
</table>

A. 1.75  
B. 1.85  
C. 2.06
5) What is the output from Q4 when run on the table shown?

Q4: SELECT COUNT(DISTINCT height) FROM Harlem_Globetrotters WHERE pos NOT IN ('F', 'G')

<table>
<thead>
<tr>
<th>no</th>
<th>name</th>
<th>height</th>
<th>pos</th>
<th>school</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Lili &quot;Champ&quot; Thompson</td>
<td>1.75</td>
<td>G</td>
<td>Notre Dame State</td>
</tr>
<tr>
<td>2</td>
<td>Carlis &quot;Dizzy&quot; English</td>
<td>1.70</td>
<td>G</td>
<td>Cleveland State</td>
</tr>
<tr>
<td>3</td>
<td>Tay &quot;Firefly&quot; Fisher</td>
<td>1.75</td>
<td>G</td>
<td>Siena</td>
</tr>
<tr>
<td>6</td>
<td>Brianna &quot;Hoops&quot; Green</td>
<td>1.75</td>
<td>G</td>
<td>UT - El Paso</td>
</tr>
<tr>
<td>31</td>
<td>Donte &quot;Hammer&quot; Harrison</td>
<td>2.06</td>
<td>F</td>
<td>Hampton</td>
</tr>
<tr>
<td>15</td>
<td>Brittany &quot;Ice&quot; Hrynko</td>
<td>1.85</td>
<td>G</td>
<td>Montana</td>
</tr>
</tbody>
</table>

A. 0
B. 1
C. 3
D. 4
## Latest College ERD

### Student_Beam
- **PK**: sid
- **String**: string
- **fname**: string
- **lname**: string
- **dob**: date
- **status**: string

### Teacher_Beam
- **PK**: tid
- **fname**: string
- **lname**: string
- **dept**: string

### Takes_Beam
- **PK, FK**: sid
- **cid**: string
- **grade**: string

### Teaches
- **PK, FK**: tid
- **cid**: string

### Class
- **PK**: cid
- **cno**: string
- **cname**: string
- **credits**: integer
Syntax of Global Aggregate Queries

```
SELECT <aggregate function>[, <aggregate function>] 
FROM <single table> 
[JOIN <single table> ON <join condition>] 
[WHERE <boolean condition>]  
ORDER BY <field(s) to sort on>
```
Syntax of Group By Queries

SELECT <unaggregated field(s)>
FROM <single table>
[JOIN <single table>
ON <join condition>]
[WHERE <boolean condition>]
GROUP BY <unaggregated field(s)>
First Question

What are the years of birth and status of all students?

Student_Beam(sid, fname, lname, dob, status)
Class(cid, cno, cname, credits)
Teacher_Beam(tid, fname, lname, dept)
Takes_Beam(sid, cid, grade)
Teaches(tid, cid)
Syntax of Aggregate Queries with Groups

```
SELECT <unaggregated field(s)>, <aggregate function(s)>
FROM <single table>
[JOIN <single table> ON <join condition>]
[WHERE <boolean condition>]
GROUP BY <unaggregated field(s)>
[HAVING <boolean condition>]
[ORDER BY <field(s) to sort on>]
```
How `COUNT()` works

1) SELECT `COUNT(*)`
   FROM `Employee`

2) SELECT `COUNT(emp_dept)`
   FROM `Employee`

3) SELECT `COUNT(DISTINCT emp_dept)`
   FROM `Employee`
Second Question

How many students are taking each class?

Student_Beam(sid, fname, lname, dob, status)
Class(cid, cno, cname, credits)
Teacher_Beam(tid, fname, lname, dept)
Takes_Beam(sid, cid, grade)
Teaches(tid, cid)
Third Question

*For each class with at least two students, how many students are taking such a class?*

- `Student_Beam(sid, fname, lname, dob, status)`
- `Class(cid, cno, cname, credits)`
- `Teacher_Beam(tid, fname, lname, dept)`
- `Takes_Beam(sid, cid, grade)`
- `Teaches(tid, cid)`
For each class with at least two students, how many students are taking such a class?

Student_Beam(sid, fname, lname, dob, status)
Class(cid, cno, cname, credits)
Teacher_Beam(tid, fname, lname, dept)
Takes_Beam(sid, cid, grade)
Teaches(tid, cid)

Does this query require a **HAVING** clause?
A. Yes
B. No
Database Views

- Return a table of results from a SQL query
- Saved in the database as named query
- Defined by `CREATE VIEW` statement

Employee(`empid`, `fname`, `lname`, `job`, `level`, `start_date`, `curr_salary`, `dob`, `ssn`, `emergency_contact`)

CREATE VIEW Manager_View AS
SELECT `empid`, `fname`, `lname`, `job`, `level`, `start_date`, `curr_salary`
FROM Employee
WHERE level NOT IN ('Executive', 'CEO')
ORDER BY `empid`

SELECT `empid`, `fname`, `lname` FROM Manager_View
WHERE start_date < '2020-01-01'
AND `job` = 'Data Engineer'
Example Views

CREATE VIEW Executive_View AS
    SELECT empid, fname, lname, job, level, start_date, curr_salary
    FROM Employee
    WHERE level != 'CEO'
    ORDER BY empid

SELECT empid, fname, lname
FROM Executive_View
WHERE curr_salary > 300000
AND level = 'Director'

CREATE VIEW Manager_View AS
    SELECT empid, fname, lname, job, level, start_date, curr_salary
    FROM Executive_View
    WHERE level != 'Executive'
    ORDER BY empid

SELECT empid, fname, lname
FROM Manager_View
WHERE start_date < '2020-01-01'
AND role = 'Data Engineer'
Case Study: Coronavirus (COVID-19)

**Jupyter Notebook:**
- Walk through ingestion pipeline
- Walk through modeling pipeline
- Walk through analysis queries
- Create BQ views

**Data Studio:**
- Create data sources in Data Studio
- Connect data sources to BQ views
- Create charts and reports in Data Studio
Milestone 7

Appendix: Additional Practice Problems
Fourth Question

For each student who is at least 19-years old and is earning more than 2 class credits, how many total class credits are such students earning?

Student(sid, fname, lname, dob)
Class(cno, cname, credits)
Teacher(tid, fname, lname, dept)
Takes(sid, cno, grade)
Teaches(tid, cno)
iClicker Question

For each student who is 19-years old or above and is earning at least 3 class credits, how many total class credits are such students earning?

Student(sid,.fname, lname, dob)
Class(cno, cname, credits)
Teacher(tid, fname, lname, dept)
Takes(sid, cno, grade)
Teaches(tid, cno)

Does this query require a WHERE clause?
A. Yes   B. No
Fifth Question

*Who takes exactly 3 classes?*

Show the answer as a sorted list of sids.

Student(*sid*, fname, lname, dob)
Class(*cno*, cname, credits)
Teacher(*tid*, fname, lname, dept)
Takes(*sid*, *cno*, grade)
Teaches(*tid*, *cno*)
**iClicker Question**

*Who takes exactly 3 classes?*

Show the answer as a sorted list of sids.

Does this query contain an aggregate function in the `SELECT` clause?
A. Yes  B. No