DinoDiscover SQL: Aggregation & Visualization of Fossil Data

Tara Cool
Boston University Metropolitan College
CS 669: Database Design & Implementation for Business
Professor Warren Mansur
Group 1 (Pamela Farr, Facilitator)
May 30, 2023

Section One – Aggregating Data

1. *Creating Table Structure and Data* – In this step, we will create table structures, define columns with appropriate data types, and establish constraints. Additionally, we will populate these tables with data, including an additional dinosaur discovery of our choice. The provided data is in a flattened representation, but we will insert it relationally into the schema using foreign keys.

Location	Dig Name	Dig Cost	Dinosaur Common Name	Weight (in pounds)	Paleontologist
Stonesfield	Great British Dig	$8,000	Megalosaurus	3000	William Buckland
Stonesfield	Great British Dig	$8,000	Apatosaurus	4000	William Buckland
Stonesfield	Great British Dig	$8,000	Triceratops	4500	William Buckland
Stonesfield	Great British Dig	$8,000	Stegosaurus	3500	William Buckland
Utah	Parowan Dinosaur Tracks	$10,000	Parasaurolophus	6000	John Ostrom
Utah	Parowan Dinosaur Tracks	$10,000	Tyrannosaurus Rex	5000	John Ostrom
Utah	Parowan Dinosaur Tracks	$10,000	Velociraptor	7000	John Ostrom
Arizona	Dynamic Desert Dig	$3,500	Tyrannosaurus Rex	6000	John Ostrom
Stonesfield	Mission Jurassic Dig		Spinosaurus	8000	Henry Osborn
Stonesfield	Mission Jurassic Dig		Diplodocus	9000	Henry Osborn
Stonesfield	Ancient Site Dig	$5,500	Tyrannosaurus Rex	7500	Henry Osborn
Stonesfield	Ancient Site Dig	$5,500	Velociraptor	6500	Henry Osborn

Note that the Dig Cost for “Mission Jurassic Dig” is null (has no value).

The information for the additional dinosaur discovery was added to the table above; denoted in blue font.

```sql
CREATE TABLE Location (
location_id DECIMAL(12) NOT NULL PRIMARY KEY,
location_name VARCHAR(64) NOT NULL);

CREATE TABLE Dig_site (
dig_site_id DECIMAL(12) NOT NULL PRIMARY KEY,
location_id DECIMAL(12) NOT NULL,
dig_name VARCHAR(32) NOT NULL,
dig_cost DECIMAL(8,2) NULL,
FOREIGN KEY (location_id) REFERENCES Location(location_id));

CREATE TABLE Paleontologist (
paleontologist_id DECIMAL(12) NOT NULL PRIMARY KEY,
first_name VARCHAR(32) NOT NULL,
last_name VARCHAR(32) NOT NULL);

CREATE TABLE Dinosaur_discovery (
dinosaur_discovery_id DECIMAL(12) NOT NULL PRIMARY KEY,
dig_site_id DECIMAL(12) NOT NULL,
paleontologist_id DECIMAL(12) NOT NULL,
common_name VARCHAR(64) NOT NULL,
fossil_weight DECIMAL(6) NOT NULL,
FOREIGN KEY (paleontologist_id) REFERENCES Paleontologist(paleontologist_id));
```

100 %

Messages

```
Commands completed successfully.
```

```sql
INSERT INTO Location (location_id, location_name)
VALUES (101, 'Stonesfield');
INSERT INTO Location (location_id, location_name)
VALUES (102, 'Utah');
INSERT INTO Location (location_id, location_name)
VALUES (103, 'Arizona');

INSERT INTO Dig_site (dig_site_id, location_id, dig_name, dig_cost)
VALUES (11, 101, 'Great British Dig', 8000);
INSERT INTO Dig_site (dig_site_id, location_id, dig_name, dig_cost)
VALUES (22, 102, 'Parowan Dinosaur Tracks', 10000);
INSERT INTO Dig_site (dig_site_id, location_id, dig_name, dig_cost)
VALUES (33, 103, 'Dynamic Desert Dig', 3500);
INSERT INTO Dig_site (dig_site_id, location_id, dig_name, dig_cost)
VALUES (44, 101, 'Mission Jurassic Dig', NULL);
INSERT INTO Dig_site (dig_site_id, location_id, dig_name, dig_cost)
VALUES (55, 101, 'Ancient Site Dig', 5500);

INSERT INTO Paleontologist (paleontologist_id, first_name, last_name)
VALUES (100, 'William', 'Buckland');
INSERT INTO Paleontologist (paleontologist_id, first_name, last_name)
VALUES (200, 'John', 'Ostrom');
INSERT INTO Paleontologist (paleontologist_id, first_name, last_name)
VALUES (300, 'Henry', 'Osborn');
```

```
53 INSERT INTO Dinosaur_discovery (dinosaur_discovery_id, dig_site_id, paleontologist_id, common_name, fossil_weight)
54 VALUES (1, 11, 100, 'Megalosaurus', 3000);
55 INSERT INTO Dinosaur_discovery (dinosaur_discovery_id, dig_site_id, paleontologist_id, common_name, fossil_weight)
56 VALUES (2, 11, 100, 'Apatosaurus', 4000);
57 INSERT INTO Dinosaur_discovery (dinosaur_discovery_id, dig_site_id, paleontologist_id, common_name, fossil_weight)
58 VALUES (3, 11, 100, 'Triceratops', 4500);
59 INSERT INTO Dinosaur_discovery (dinosaur_discovery_id, dig_site_id, paleontologist_id, common_name, fossil_weight)
60 VALUES (4, 11, 100, 'Stegosaurus', 3500);
61 INSERT INTO Dinosaur_discovery (dinosaur_discovery_id, dig_site_id, paleontologist_id, common_name, fossil_weight)
62 VALUES (5, 22, 200, 'Parasaurolophus', 6000);
63 INSERT INTO Dinosaur_discovery (dinosaur_discovery_id, dig_site_id, paleontologist_id, common_name, fossil_weight)
64 VALUES (6, 22, 200, 'Tyrannosaurus Rex', 5000);
65 INSERT INTO Dinosaur_discovery (dinosaur_discovery_id, dig_site_id, paleontologist_id, common_name, fossil_weight)
66 VALUES (7, 22, 200, 'Velociraptor', 7000);
67 INSERT INTO Dinosaur_discovery (dinosaur_discovery_id, dig_site_id, paleontologist_id, common_name, fossil_weight)
68 VALUES (8, 33, 200, 'Tyrannosaurus Rex', 6000);
69 INSERT INTO Dinosaur_discovery (dinosaur_discovery_id, dig_site_id, paleontologist_id, common_name, fossil_weight)
70 VALUES (9, 44, 300, 'Spinosaurus', 8000);
71 INSERT INTO Dinosaur_discovery (dinosaur_discovery_id, dig_site_id, paleontologist_id, common_name, fossil_weight)
72 VALUES (10, 44, 300, 'Diplodocus', 9000);
73 INSERT INTO Dinosaur_discovery (dinosaur_discovery_id, dig_site_id, paleontologist_id, common_name, fossil_weight)
74 VALUES (11, 55, 300, 'Tyrannosaurus Rex', 7500);
75 INSERT INTO Dinosaur_discovery (dinosaur_discovery_id, dig_site_id, paleontologist_id, common_name, fossil_weight)
76 VALUES (12, 55, 300, 'Velociraptor', 6500);
```

100 %

Messages

```
(1 row affected)
(1 row affected)
(1 row affected)
(1 row affected)
(1 row affected)
(1 row affected)
(1 row affected)
(1 row affected)
(1 row affected)
(1 row affected)
(1 row affected)
(1 row affected)
```

2. *Counting Matches* – A museum wants to know how many dinosaur discoveries weigh at least 4,200 pounds. We will write a single SQL query to fulfill this request.

```
80 SELECT COUNT(dinosaur_discovery_id) AS nr_heavy_fossils
81 FROM Dinosaur_discovery
82 WHERE fossil_weight >= 4200;
```

100 %

Results Messages

	nr_heavy_fossils
1	9

3. *Determining Highest and Lowest* – The same museum needs to know the cost of the most expensive and least expensive dinosaur digs. Thus, this section addresses the cost of the most expensive and least expensive dinosaur digs. We will also delve into how

the SQL processor handles dig costs for the "Mission Jurassic Dig," where the cost is null.

```
86  SELECT FORMAT(MIN(dig_cost), '$.00') AS least_expensive,
87  FORMAT(MAX(dig_cost), '$.00') AS most_expensive
88  FROM Dig_site;
```

100 %

Results | Messages

	least_expensive	most_expensive
1	$3500.00	$10000.00

The SQL processor treated the dig costs for the “Mission Jurassic Dig” differently than the other cost values, as there is no ‘dig_cost’ value for the “Mission Jurassic Dig”(s). In other words, the dig costs for the “Mission Jurassic Dig” are ‘NULL’; and therefore, not interpreted as numeric values. For example, SQL Server recognizes that these values are unknown/not available (i.e., ‘NULL’); rather than assuming that the dig costs are equal to $0 (e.g., in that case, the “Mission Jurassic Dig” would be considered the least expensive).

4. *Grouping Aggregate Results* – We will write a single SQL query to provide the dig site names and costs along with the number of dinosaur discoveries at each site. This information is essential for a museum considering its own paleontological expedition.

```
92  SELECT dig_name, FORMAT(dig_cost, 'C') AS dig_cost, COUNT(dinosaur_discovery_id) AS nr_dinosaur_discoveries
93  FROM Dinosaur_discovery
94  JOIN Dig_site ON Dig_site.dig_site_id = Dinosaur_discovery.dig_site_id
95  GROUP BY Dig_site.dig_site_id, dig_name, dig_cost;
```

100 %

Results | Messages

	dig_name	dig_cost	nr_dinosaur_discoveries
1	Great British Dig	$8,000.00	4
2	Parowan Dinosaur Tracks	$10,000.00	3
3	Dynamic Desert Dig	$3,500.00	1
4	Mission Jurassic Dig	NULL	2
5	Ancient Site Dig	$5,500.00	2

5. *Limiting Results by Aggregation* – A paleontologist, looking to dig at a location ripe with discoveries, wants to search for locations with at least 6 dinosaur discoveries. In response to the paleontologist's request, we will create a single query to fulfill this requirement.

```
99  SELECT Location.location_name, COUNT(dinosaur_discovery_id) AS nr_dinosaur_discoveries
100 FROM Dinosaur_discovery
101 JOIN Dig_site ON Dig_site.dig_site_id = Dinosaur_discovery.dig_site_id
102 JOIN Location ON Location.location_id = Dig_site.location_id
103 GROUP BY Location.location_id, location_name
104 HAVING COUNT(dinosaur_discovery_id) >= 6;
```

100 %

Results | Messages

	location_name	nr_dinosaur_discoveries
1	Stonesfield	8

6. *Adding Up Values* – A museum needs to know which digs (not locations) had at least 15,000 pounds of discovered dinosaur remains. A single query will provide this information along with relevant columns.

```
108 SELECT dig_name, SUM(fossil_weight) AS total_fossil_weight
109 FROM Dinosaur_discovery
110 JOIN Dig_site ON Dig_site.dig_site_id = Dinosaur_discovery.dig_site_id
111 GROUP BY Dig_site.dig_site_id, dig_name
112 HAVING SUM(fossil_weight) >= 15000;
```

100 %

Results | Messages

	dig_name	total_fossil_weight
1	Great British Dig	15000
2	Parowan Dinosaur Tracks	18000
3	Mission Jurassic Dig	17000

7. *Integrating Aggregation with Other Constructs* – A research institution has requested the names of all paleontologists and the number of digs they participated in at the "Stonesfield" location. We will order this list from most to least, with the paleontologist who discovered the most Stonesfield dinosaurs at the top.

```
117 SELECT first_name + ' ' + last_name AS paleontologist_name, COUNT(Dinosaur_discovery.dig_site_id) AS nr_digs
118 FROM Dinosaur_discovery
119 JOIN Dig_site ON Dig_site.dig_site_id = Dinosaur_discovery.dig_site_id
120 JOIN Location ON Location.location_id = Dig_site.location_id AND
121 Location.location_name = 'Stonesfield'
122 RIGHT JOIN Paleontologist ON Paleontologist.paleontologist_id = Dinosaur_discovery.paleontologist_id
123 GROUP BY Paleontologist.paleontologist_id, first_name, last_name
124 ORDER BY nr_digs DESC;
```

100 %

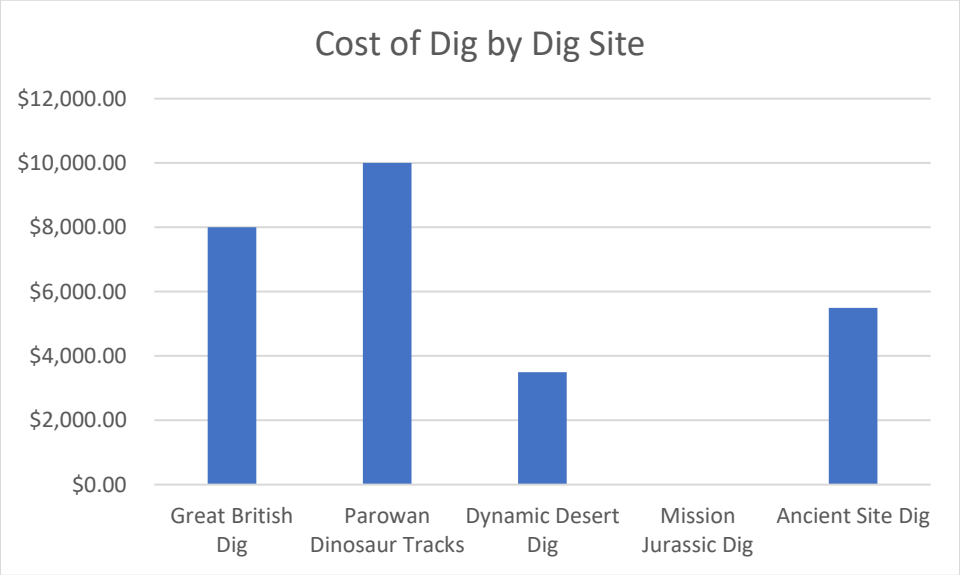
Results | Messages

	paleontologist_name	nr_digs
1	William Buckland	4
2	Henry Osborn	4
3	John Ostrom	0

Section Two –Data Visualization

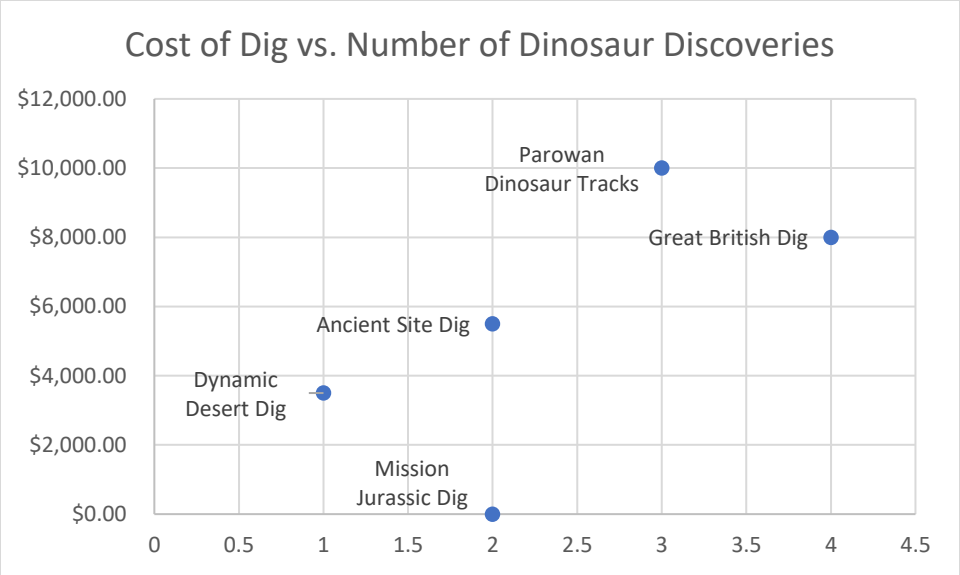
8. *Visualizing Data with One or Two Measures*

a. *Creating a Bar Chart* – We will use SQL results obtained in Step 4 to create a bar chart that represents the cost of digs at various sites. This visualization will highlight differences in dig costs, including the "Mission Jurassic Dig," where the cost is null.



This bar chart gives a visual depiction of the cost of the digs performed at each of the dig sites. According to the bar chart, the “Parowan Dinosaur Tracks” dig site had the highest dig cost ($10,000) and the “Dynamic Desert Dig” dig site had the lowest dig cost ($3,500); while the dig costs for the “Great British Dig” and “Ancient Site Dig” dig sites were somewhere in between. It is also important to note that the cost for the “Mission Jurassic Dig” dig site is given as $0; though we know from the data that this value is unknown (i.e., NULL, which is not equal to 0).

b. *Creating a scatterplot* – In this part, we will craft a scatterplot with dig costs on one axis and the number of dinosaurs found on the other axis. Each dig name will be labeled for clarity. The scatterplot will provide insights into the relationship between dig costs and the number of discoveries made at each site.



The scatterplot is a visual representation of the relationship between the cost of the dig and the number of dinosaur discoveries made according to the dig site name. In particular, the cost of the dig is denoted on the y-axis and the number of digs is depicted on the x-axis; allowing the end user to compare the two fields for each dig site. For example, the "Dynamic Desert Dig" dig site has the lowest number of dinosaur discoveries (i.e., 1 discovery) and the second lowest cost (i.e., $3,500); while the "Mission Jurassic Dig" dig site has the lowest cost (i.e., $0, even though we know from the database/SQL queries that this value is NULL) and the second lowest number of dinosaur discoveries (i.e., 2 discoveries). In a similar manner, the "Great British Dig" dig site has the highest number of dinosaur discoveries (i.e., 4 discoveries) and the second highest cost (i.e., $8,000); while the "Parowan Dinosaur Tracks" dig site has the highest cost (i.e., $10,000) and the second highest number of dinosaur discoveries (i.e., 3 discoveries). Moreover, the dig cost for the "Ancient Site Dig" dig site lies somewhere in the middle of the dig costs for the other dig sites; while the number of discoveries made here is equal to that of the "Mission Jurassic Dig" dig site (i.e., 2 discoveries).

9. *Another Data Visualization* – In this section, we will create a visualization of our choice based on data in the Dinosaur schema. The visualization will tell a useful story about the data, and we will explain why we chose this particular chart or visualization.

SQL query used to retrieve data for Step 9: *Listing the total weight of the dinosaur fossil remains found at each dig site and the number of dinosaur discoveries made; ordered by the sum of the fossil weight in each location; from the greatest to the least.*

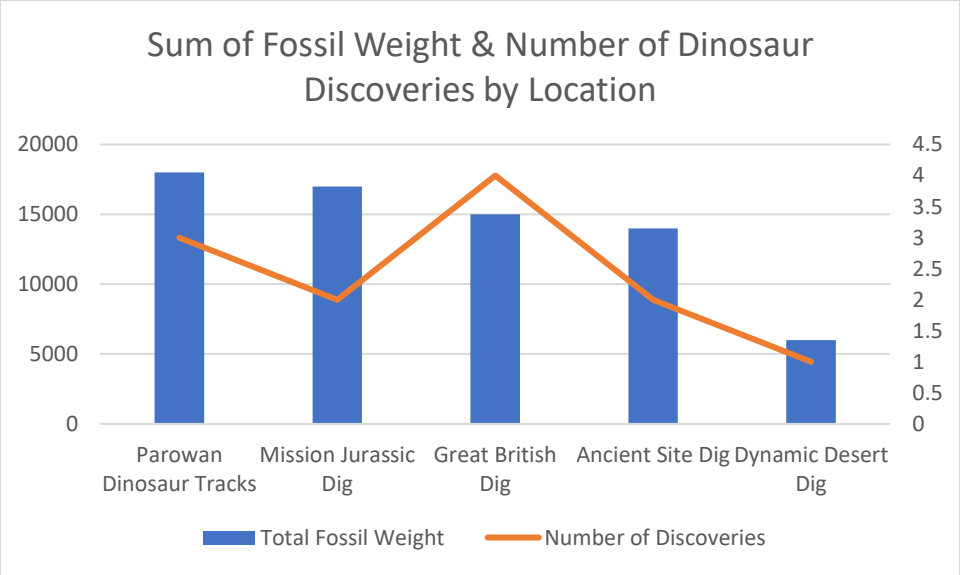
```
135 SELECT dig_name, SUM(fossil_weight) AS total_fossil_weight, COUNT(dinosaur_discovery_id) AS nr_dinosaur_discoveries
136 FROM Dinosaur_discovery
137 JOIN Dig_site ON Dig_site.dig_site_id = Dinosaur_discovery.dig_site_id
138 GROUP BY Dig_site.dig_site_id, dig_name
139 ORDER BY total_fossil_weight DESC;
```

100 %

Results | Messages

	dig_name	total_fossil_weight	nr_dinosaur_discoveries
1	Parowan Dinosaur Tracks	18000	3
2	Mission Jurassic Dig	17000	2
3	Great British Dig	15000	4
4	Ancient Site Dig	14000	2
5	Dynamic Desert Dig	6000	1

Visualization: ***Bar Chart – Line Chart Combination***



The visualization depicted above uses a combination of a bar chart and a line chart to represent multiple fields for each class of the attribute "Dig_site"; in which the sum of or total fossil weight and the number of discoveries are both denoted on the y-axis, with total fossil weight on the left axis and the number of discoveries on the right. Moreover, the bar chart represents the total fossil weight for each of the different dig sites

(i.e., corresponding to the left-hand scale of the y-axis), while the number of discoveries made at each location is represented by individual data points connected by a line (i.e., a line chart, corresponding to the right-hand scale of the y-axis).

The locations and/or the corresponding bars are listed in descending order by total fossil weight; suggesting that the remains found at the "Parowan Dinosaur Tracks" dig site weighed the most, while the remains at the "Dynamic Desert Dig" dig site weighed the least. From the visualization, we can also see that the "Dynamic Desert Dig" dig site had the least number of dinosaur discoveries, as well; and potentially conclude that this site was not as successful as the others in terms of their findings/discoveries. Also, the remains found at the "Great British Dig" dig site weighed the most, but the number of dinosaur discoveries made at this dig site was in between those of the other dig sites (e.g., some might say it is "average", though this is not statistically computed).