

Coding Challenge

PETPALS

TASKS:

1.

```
coding challenge 1_petpals x
Limit to 1000 rows
1  /*1 test*/
2  • CREATE DATABASE IF NOT EXISTS PetPalsDB;
3  • use petpalsdb;
```

2,3&4.

```
5  /*2,3,4 task*/
6  • CREATE TABLE Pets (
7      PetID INT PRIMARY KEY,
8      Name VARCHAR(255),
9      Age INT,
10     Breed VARCHAR(255),
11     Type VARCHAR(255),
12     AvailableForAdoption BIT
13 );
14
15 • CREATE TABLE Shelters (
16     ShelterID INT PRIMARY KEY,
17     Name VARCHAR(255),
18     Location VARCHAR(255)
19 );
20
21 • CREATE TABLE Donations (
22     DonationID INT PRIMARY KEY,
23     DonorName VARCHAR(255),
24     DonationType VARCHAR(255),
25     DonationAmount DECIMAL,
26     DonationItem VARCHAR(255),
27     DonationDate DATETIME
28 );
29
30 • CREATE TABLE AdoptionEvents (
31     EventID INT PRIMARY KEY,
32     EventName VARCHAR(255),
33     EventDate DATETIME,
34     Location VARCHAR(255)
35 );
36
37 • CREATE TABLE Participants (
38     ParticipantID INT PRIMARY KEY,
39     ParticipantName VARCHAR(255),
40     ParticipantType VARCHAR(255),
41     EventID INT REFERENCES AdoptionEvents(EventID)
42 );
```

Data Insertion:

```
coding challenge 1_petsqls
Limit to 1000 rows

44 INSERT INTO Pets (PetID, Name, Age, Breed, Type, AvailableForAdoption)
45 VALUES
46 (1, 'Sasha', 2, 'Indian Cat', 'Cat', 1),
47 (2, 'Boola', 3, 'Librarian', 'Dog', 1),
48 (3, 'Peera', 1, 'Persian Cat', 'Cat', 4),
49 (4, 'Raj', 4, 'Golden Retriever', 'Dog', 1)
50
51
52 INSERT INTO Shelters (ShelterID, Name, Location)
53 VALUES
54 (1, 'Happy Paws Shelter', 'Mumbai'),
55 (2, 'Compassionate Care Shelter', 'Delhi'),
56 (3, 'Loving Hearts Shelter', 'Bangalore')
57
58
59 INSERT INTO Donations (DonationID, DonorName, DonationType, DonationAmount, DonationTime, DonationDate)
60 VALUES
61 (1, 'Ananya', 'Cash', 500.00, NULL, '2023-03-01'),
62 (2, 'Rahul', 'Time', NULL, 'Pet Food', '2023-03-10'),
63 (3, 'Sneha', 'Crack', 300.00, NULL, '2023-03-20')
64
65
66 INSERT INTO AdoptionEvents (EventID, EventName, EventDate, Location)
67 VALUES
68 (1, 'Adoption Day', '2023-04-01', 'Jaipur'),
69 (2, 'Pet Fest', '2023-05-15', 'Mumbai')
70
71
72 INSERT INTO Participants (ParticipantID, ParticipantName, ParticipantType, EventID)
73 VALUES
74 (1, 'Happy Paws Shelter', 'Shelter', 1),
75 (2, 'Anita', 'Adopter', 1),
76 (3, 'Compassionate Care Shelter', 'Shelter', 2),
77 (4, 'Neha', 'Adopter', 2)
78
```

5.

The screenshot shows a database query editor with a SQL query in the main window. The query is as follows:

```
79
80 /* 5 task*/
81 * SELECT Name, Age, Breed, Type
82 FROM Pets
83 WHERE AvailableForAdoption = 1;
84
```

Below the query editor, the 'Result Grid' is displayed, showing the results of the query:

Name	Age	Breed	Type
Simba	2	Indian Cat	Cat
Whisk	2	Labrador	Dog
Raj	8	Golden Retriever	Dog

The 'Output' pane at the bottom shows the execution details:

#	Time	Action	Message	Duration / Fetch
1	21:53:32	SELECT Name, Age, Breed, Type FROM Pets WHERE AvailableForAdoption = 1 LIMIT 0: 1000	3 row(s) returned	0.000 sec / 0.000 sec

6.

The screenshot shows a database query editor with a SQL query in the main window. The query is as follows:

```
84 /*6 task*/
85 * SELECT ParticipantName
86 FROM Participants P
87 JOIN AdoptionEvents AE ON P.EventID = AE.EventID
88 WHERE AE.EventID = 1;
89
```

Below the query editor, the 'Result Grid' is displayed, showing the results of the query:

ParticipantName
Happy Paws Shelter
Arkita

The 'Output' pane at the bottom shows the execution details:

#	Time	Action	Message	Duration / Fetch
1	21:54:21	SELECT ParticipantName FROM Participants P JOIN AdoptionEvents AE ON P.EventID = AE.EventID WHERE...	2 row(s) returned	0.094 sec / 0.000 sec

7.

The screenshot shows a SQL script in SQL Studio. The script defines a procedure `updateShelterInfo` that takes three parameters: `shelterIDParam INT`, `newNameParam VARCHAR(255)`, and `newLocationParam VARCHAR(100)`. The procedure uses a `SELECT COUNT(*)` to check if the shelter exists. If it does, it updates the `name` and `location` columns. If it doesn't, it inserts a new row. The procedure is then called with the values 'New Shelter', 'Mumbai', and 'Loving Hearts Shelter'.

```

100 /***/
101 DELIMITER //
102 CREATE PROCEDURE updateShelterInfo( IN shelterIDParam INT, IN newNameParam VARCHAR(255), IN newLocationParam VARCHAR(100))
103 BEGIN
104     DECLARE shelterExists INT;
105     SELECT COUNT(*) INTO shelterExists FROM shelters WHERE shelterID = shelterIDParam;
106     IF shelterExists > 0 THEN
107         UPDATE shelters
108         SET name = newNameParam, location = newLocationParam
109         WHERE shelterID = shelterIDParam;
110     ELSE
111         INSERT INTO shelters (shelterID, name, location)
112         VALUES (shelterIDParam, newNameParam, newLocationParam);
113     END IF;
114 END //
115 DELIMITER ;
116 CALL updateShelterInfo(1, 'New Shelter', 'Mumbai');
117 SELECT * FROM shelters;
118

```

The Result Grid shows the following data:

ShelterID	Name	Location
1	New Shelter	Mumbai
2	Compassionate Care Shelter	Delft
3	Loving Hearts Shelter	Bangalore

The Output pane shows the following actions:

#	Time	Action	Message	Duration / Fetch
1	12:42:28	CREATE PROCEDURE updateShelterInfo(IN shelterIDParam INT, IN newNameParam VARCHAR(255), IN newLocationParam VARCHAR(100))	0 row(s) affected	0.103 sec
2	12:43:22	select * from shelters LIMIT 0, 1000	3 row(s) returned	0.125 sec / 0.000 sec
3	12:43:53	CALL updateShelterInfo(1, 'New Shelter', 'Mumbai')	1 row(s) affected	0.042 sec
4	12:44:05	select * from shelters LIMIT 0, 1000	3 row(s) returned	0.000 sec / 0.000 sec

8.

The screenshot shows a SQL query in SQL Studio that calculates the total donation amount for each shelter. The query uses a `LEFT JOIN` between the `Shelters` table and the `Donations` table, grouped by `S.Name`. The `IFNULL` function is used to handle shelters with no donations.

```

93 /*** task*/
94 SELECT S.Name AS ShelterName, IFNULL(SUM(DonationAmount), 0) AS TotalDonationAmount
95 FROM Shelters S
96 LEFT JOIN Donations D ON S.ShelterID = D.donationID
97 GROUP BY S.Name;
98

```

The Result Grid shows the following data:

ShelterName	TotalDonationAmount
Happy Paws Shelter	500
Compassionate Care Shelter	0
Loving Hearts Shelter	300

The Output pane shows the following action:

#	Time	Action	Message	Duration / Fetch
1	21:55:46	SELECT S.Name AS ShelterName, IFNULL(SUM(DonationAmount), 0) AS TotalDonationAmount FROM Shelters...	3 row(s) returned	0.405 sec / 0.000 sec

9.

The screenshot shows the SQL Server Enterprise Manager interface. The query editor at the top contains the following SQL code:

```
99
100 /*9 task*/
101 * SELECT Name, Age, Breed, Type
102 FROM Pets
103 WHERE AvailableForAdoption= '1';
104
```

Below the query editor, the 'Result Grid' displays the results of the query. The results are as follows:

Name	Age	Breed	Type
Simba	2	Jordan Cat	Cat
Rhola	3	Labrador	Dog
Kaj	4	Golden Retriever	Dog

The 'Output' pane at the bottom shows the execution details:

#	Time	Action	Message	Duration / Fetch
1	21:56:32	SELECT Name, Age, Breed, Type FROM Pets WHERE AvailableForAdoption= '1' LIMIT 0, 1000	3 row(s) returned	0.016 sec / 0.000 sec

10.

The screenshot shows the SQL Server Enterprise Manager interface. The query editor at the top contains the following SQL code:

```
105
106 /*10 task*/
107 * select donationdate ,IFNULL(SUM(DonationAmount), 0) AS TotalDonationAmount
108 from donations
109 group by(donationdate);
110
```

Below the query editor, the 'Result Grid' displays the results of the query. The results are as follows:

donationdate	TotalDonationAmount
2023-01-15 00:00:00	300
2023-02-20 00:00:00	0
2023-03-20 00:00:00	300

The 'Output' pane at the bottom shows the execution details:

#	Time	Action	Message	Duration / Fetch
1	21:57:00	select donationdate ,IFNULL(SUM(DonationAmount), 0) AS TotalDonationAmount from donations group by(donationdate)	3 row(s) returned	0.000 sec / 0.000 sec

11.

The screenshot shows the SQL Studio interface. The query editor contains the following SQL code:

```
111
112 /*11 task*/
113 * SELECT DISTINCT Breed
114 FROM Pets
115 WHERE (Age BETWEEN 1 AND 3) OR (Age > 5);
116
```

The Results Grid shows the following data:

Breed
Indian Cat
Labrador
Persian Cat

The Output pane shows the execution details:

#	Time	Action	Message	Duration / Fetch
1	21:57:40	SELECT DISTINCT Breed FROM Pets WHERE (Age BETWEEN 1 AND 3) OR (Age > 5) LIMIT 0, 1000	2 row(s) returned	0.000 sec / 0.000 sec

12.

The screenshot shows the SQL Studio interface. The query editor contains the following SQL code:

```
117 /*12 task*/
118 * SELECT P.Name AS PetName, P.Breed, P.Type, S.Name AS ShelterName
119 FROM Pets P
120 JOIN Shelters S ON P.petid = S.ShelterID
121 WHERE P.AvailableForAdoption = 1;
122
```

The Results Grid shows the following data:

PetName	Breed	Type	ShelterName
Simba	Indian Cat	Cat	Happy Paws Shelter
Whisk	Labrador	Dog	Compassionate Care Shelter

The Output pane shows the execution details:

#	Time	Action	Message	Duration / Fetch
1	21:58:22	SELECT P.Name AS PetName, P.Breed, P.Type, S.Name AS ShelterName FROM Pets P JOIN Shelters S ON P...	2 row(s) returned	0.000 sec / 0.000 sec

13.

The screenshot shows the SQL Server Enterprise Manager interface. The query editor contains the following SQL code:

```
123 /*13 task*/
124 * SELECT COUNT(*) AS TotalParticipants
125 FROM Participants P
126 JOIN AdoptionEvents AE ON P.EventID = AE.EventID
127 JOIN Shelters S ON AE.Location = 'jaigaon' AND P.ParticipantID = S.ShelterID
128 WHERE P.ParticipantType = 'Shelter';
129
```

The Results pane shows a single row with the value 1 for the column TotalParticipants.

The Output pane shows the following message:

```
1 21:59:02 SELECT COUNT(*) AS TotalParticipants FROM Participants P JOIN AdoptionEvents AE ON P.EventID = AE.Ex ... 1 row(s) returned
```

The Duration / Fetch is 0.000 sec / 0.000 sec.

14.

The screenshot shows the SQL Server Enterprise Manager interface. The query editor contains the following SQL code:

```
130
131 /*14 task*/
132 * SELECT DISTINCT Breed
133 FROM Pets
134 WHERE Age BETWEEN 1 AND 5;
135
136
```

The Results pane shows a list of breeds: Indian Cat, Labrador, Persian Cat, and Golden Retriever.

The Output pane shows the following message:

```
1 21:59:35 SELECT DISTINCT Breed FROM Pets WHERE Age BETWEEN 1 AND 5 LIMIT 0 1000
4 row(s) returned
```

The Duration / Fetch is 0.000 sec / 0.000 sec.

15.

The screenshot shows a SQL IDE interface with a query editor at the top containing the following SQL code:

```
137
138: /*15 task*/
139: SELECT Name, Age, Breed, Type
140: FROM Pets
141: WHERE AvailableForAdoption=1;
142
143
```

Below the editor is a 'Result Grid' showing the results of the query:

Name	Age	Breed	Type
Simba	2	Indian Cat	Cat
Whisker	3	Labrador	Dog
Raj	4	Golden Retriever	Dog

The bottom panel shows the 'Action Output' with the following message:

Time	Action	Message	Duration / Fetch
1 22:02:58	SELECT Name, Age, Breed, Type FROM Pets WHERE AvailableForAdoption=1 LIMIT 5, 1000	3 rows returned	0.016 sec / 0.000 sec

16.

The screenshot shows a SQL IDE interface with a query editor at the top containing the following SQL code:

```
142
143
144: /*16 task*/
145: /*No column to connect adoptions with petid and adoptions with userid*/
146
147
148
```

Below the editor is a 'Result Grid' showing the results of the query:

Name	Age	Breed	Type
Simba	2	Indian Cat	Cat
Whisker	3	Labrador	Dog
Raj	4	Golden Retriever	Dog

The bottom panel shows the 'Action Output' with the following message:

Time	Action	Message	Duration / Fetch
1 22:02:59	SELECT Name, Age, Breed, Type FROM Pets WHERE AvailableForAdoption=1 LIMIT 5, 1000	3 rows returned	0.000 sec / 0.000 sec

17.

SQL Server Enterprise Manager - SQL:Adventureworks

```

148
149 /*17 task*/
150 * SELECT S.Name AS ShelterName, COUNT(P.PetID) AS TotalAvailablePets
151 FROM Shelters S
152 LEFT JOIN Pets P ON S.ShelterID = P.PetID AND P.AvailableForAdoption = 1 GROUP BY S.Name;
153 /* there is not shelter id in table pet or viserversa so i connected shelterid with petid*/
154

```

Result Grid

ShelterName	TotalAvailablePets
Happy Paws Shelter	0
Compassionate Care Shelter	0
Living Hearts Shelter	0

Result 13

Output

Action Output

Time	Action	Message	Duration / Fetch
22:02:50	SELECT Name, Age, Breed, Type FROM Pets WHERE AvailableForAdoption=1 LIMIT 8, 1000	3 row(s) returned	0.000 sec / 0.000 sec
22:03:22	SELECT S.Name AS ShelterName, COUNT(P.PetID) AS TotalAvailablePets FROM Shelters S LEFT JOIN Pets ...	3 row(s) returned	0.000 sec / 0.000 sec

18.

SQL Server Enterprise Manager - SQL:Adventureworks

```

154
155 /*18 task*/
156 * SELECT A.PetID AS Pet1, B.PetID AS Pet2
157 FROM Pets A
158 JOIN Pets B ON A.petid = B.petID AND A.Breed = B.Breed AND A.PetID < B.PetID;
159 /* there is not shelter id in table pet or viserversa so i connected shelterid with petid*/
160

```

Result Grid

Pet1	Pet2
------	------

Result 14

Output

Action Output

Time	Action	Message	Duration / Fetch
22:03:58	SELECT A.PetID AS Pet1, B.PetID AS Pet2 FROM Pets A JOIN Pets B ON A.petid = B.petID AND A.Breed = B...	0 row(s) returned	0.016 sec / 0.000 sec

19.

SQL Server Enterprise Manager - SQL:Adventureworks

My Snippets

```

160
161 /*19 task*/
162 SELECT S.Name AS ShelterName, AE.EventName
163 FROM Shelters S
164 CROSS JOIN AdoptionEvents AE;
165
166

```

Result Grid

ShelterName	EventName
Happy Paws Shelter	Pet First
Happy Paws Shelter	Adoption Day
Compassionate Care Shelter	Pet First
Compassionate Care Shelter	Adoption Day
Loving Hearts Shelter	Pet First
Loving Hearts Shelter	Adoption Day

Result 13

Read Only Control Help Snippets

Output

Action Output

#	Time	Action	Message	Duration / Fetch
1	22:03:58	SELECT A.PetID AS Pet1, B.PetID AS Pet2 FROM Pets A JOIN Pets B ON A.petid = B.petid AND A.Breed = B...	0 rows returned	0.016 sec / 0.000 sec
2	22:04:26	SELECT S.Name AS ShelterName, AE.EventName FROM Shelters S CROSS JOIN AdoptionEvents AE LIMIT 0...	6 rows returned	0.000 sec / 0.000 sec

20.

SQL Server Enterprise Manager - SQL:Adventureworks

My Snippets

```

167
168 /*20 task*/
169 SELECT S.Name AS ShelterName, COUNT(A.PetID) AS AdoptedPetsCount
170 FROM Shelters S
171 JOIN Pets P ON S.ShelterID = P.ShelterID
172 JOIN AdoptionEvents A ON P.PetID = A.PetID
173 GROUP BY S.Name
174 ORDER BY AdoptedPetsCount DESC LIMIT 1;
175
176 /* not possible to connect shelter_id with pet table and adoptionevent with pet id there is not relation between them
177 the above can be the query also */

```

Result Grid

ShelterName	EventName
Happy Paws Shelter	Pet First
Happy Paws Shelter	Adoption Day
Compassionate Care Shelter	Pet First
Compassionate Care Shelter	Adoption Day
Loving Hearts Shelter	Pet First
Loving Hearts Shelter	Adoption Day

Result 13

Read Only Control Help Snippets

Output

Action Output

#	Time	Action	Message	Duration / Fetch
1	22:03:58	SELECT A.PetID AS Pet1, B.PetID AS Pet2 FROM Pets A JOIN Pets B ON A.petid = B.petid AND A.Breed = B...	0 rows returned	0.016 sec / 0.000 sec
2	22:04:26	SELECT S.Name AS ShelterName, AE.EventName FROM Shelters S CROSS JOIN AdoptionEvents AE LIMIT 0...	6 rows returned	0.000 sec / 0.000 sec