Developers Blog

Core Programmers

Exam # 3: SQL Test

Total Marks: 100

Time: 1 hr 30 mins (No Extra)

Name :

Cell Phone :

Email :

----------2nd Half----------

[****Note:** Please follow the instructions list and don't write anything in the 2nd half]

Remotely	:	

Internet	:	

Instructions:

1. **Do not write anything except only name, cell phone and email in the first page.**
2. If you have internet facilities then you will not get any partial credits.
3. If you are using computer and having exam remotely. You can only use notepad2 in text file format to answer questions.
4. **[Correct]** tag means check for code correction only write that corrected line of code.
5. **[Text]** tag means code is not required and only describe in few sentence or word. Whatever feel straight, if a keyword describes it then that's it, you don't have to write further.
6. **[Diagram/Results]** show it in excel file with appropriate question number. Use only one to write all answers.
7. **[True/false]** tag means expects answer true or false.
8. **[Rewite]** tag means rewrite the code with correct syntax.
9. **[Code]** tag means sql code syntax should be in appropriate manner (100% correct syntax) with appropriate database name, if any mistakes in syntax number will be deducted.
10. Tag mixer **[Code] [Text] [Syntax]** means write codes, with appropriate syntax and describe in text.
11. **[Output]** write the outputs explicitly. If parameter is string "Hello" passed. If Parameter is int then 123 is passed.
12. **[Multiple]** means in mcq that there could be multiple answers and if 3 of those are the right answers and if you choose only 2 correct and one wrong then you will get (33.33 x 2)% - 33.33% = 33.33% on that question.
13. **[SQL #] # represents how many SQL statements required to answer that question.**
14. Wait for answer session to get details of the problems.

Comments about your exam:

1.
2.
3.
4.
5.

Start from here in your answer script or notepad2 and results in Excel file if asked.

1. Basics 40 Marks [No partial marks]
2. Advance 60 Marks [Depends but most cases no partial marks]
   When you are correcting or writing SQL it could be Oracle, MS SQL, MySQL or Microsoft Access form, whichever you follow just write it down. If you are choosing from MCQ in most case it supported by all language or else specific one will be written in the question.

Table : A

AID	AOID	AName	ALocation	BID
1	1	A1	Loc1	1
2	2	A 2	oc 2	*NULL*
3	2	A 3	oc 3	*NULL*
4	3	A4	oc4	1

Table : B

BID	BName	AID
1	B1	1
2	B2	2
3	B3	2
4	B4	1
5	B5	3

Select * from A ; -- means returns all rows of Table A
Select * from B ; -- means returns all rows of Table B

Example of Question in Details: If you have a table named 'Hello World' and assume there are several fields or columns in the table but I would like to select one column or field named 'How you Doing'. How can I do it? **[SQL 1]**

Same question can also be represented as below:

⇨ **Table "**Hello World**"**
Select column name "How you Doing" **[SQL 1] [Syntax]**

Basic 40 Marks

1. **Write query for Table "**Hello World**"** **Select column name "**How you Doing", "Name Date" **[SQL 1] [Syntax]**	1
2. Table "A" Select column name "AOID" is int type Search for AOID = "1","3" and ALocation = "loc1","oc2" with IN syntax **[SQL 1] [Syntax]**	2
3. Table "A" Select column name "ALocation" is varchar type Search for "loc2","loc4" values with IN syntax **[SQL 1] [Syntax]**	1
4. Table "A" Select as ALocation (varchar) + BID(int) **[SQL 1] [Syntax][MS SQL Server][Results]**	2
5. Table "A" Select as ALocation (varchar) + BID(int) **[SQL 1] [Syntax][Access]**	1
6. Table "A" Select as ALocation (varchar) + BID(int) **[SQL 1] [Syntax][MySQL][Results]**	2
7. Table "A" Select as ALocation (varchar) + BID(int) **[SQL 1] [Syntax][Oracle]**	1

8. Below SQL is a valid query in MS Access, MySQL, MS SQL Server and result might be differ from database to database but the concept of the result is same in all databases. — 2

Describe Table : C

Colu...	Data Type	A...
CID	int	☐
CN1	varchar(50)	☑
CN2	varchar(50)	☑

Table: C

CID	CN1	CN2
1	N1-V1	*NULL*
2	*NULL*	N2-V2
3	N1-V3	N2-V3
4	N1-V4	*NULL*
5	*NULL*	N2-V5

```
select CN1 + ' ' + CN2 AS Display from C ;
```

What would be the result? **[Results]**

9. What is the correct order of clauses in the select statement?[only one correct] — 2.5

1.select
2.order by
3.where
4.having
5.group by

A. 1,2,3,4,5
B. 1,3,5,4,2
C. 1,3,5,2,4
D. 1,3,2,5,4
E. 1,3,2,4,5
F. 1,5,2,3,4
G. 1,4,2,3,5
H. 1,4,3,2,5

10. Which one of those queries will give same result? Based on Table : A. **[Multiple]** — 2

AID	AOID	AName	ALocation	BID
1	1	A1	Loc1	1
2	2	A2	Loc2	*NULL*
3	2	A3	Loc3	*NULL*
4	3	A4	Loc4	1

```
SELECT * FROM A
WHERE BID = 1 AND
      ALocation IN ('loc1','loc2','loc3','loc4');
```

a. Query 1:

```
SELECT * FROM A
WHERE BID = 1 AND
      ALocation = 'loc3' OR
      ALocation = 'loc2' OR
      ALocation = 'loc1' OR
      ALocation = 'loc4';
```

b. Query 2:

```
SELECT * FROM A
WHERE BID = 1 AND
      ALocation = 'loc2' OR
      ALocation = 'loc3' OR
      ALocation = 'loc1' OR
      ALocation = 'loc4';
```

c. Query 3:

```
SELECT * FROM A
WHERE BID = 1 AND
      (ALocation = 'loc2' OR
      ALocation = 'loc3' OR
      ALocation = 'loc1' OR
      ALocation = 'loc4');
```

d. Query 4:

```
SELECT * FROM A
WHERE BID = 1 AND
      ALocation = 'loc2' AND
      ALocation = 'loc3' AND
      ALocation = 'loc1' AND
      ALocation = 'loc4';
```

11. Write a query based on Table A. BID can't be null and ALocation starts with "oc". **[SQL 1] [Syntax]**	1
12. On Table A, how to select items which are **not** started with "loc" in ALocation field. **[SQL 1] [Syntax]**	1
13. On Table A, how to select items which are started with "loc" in ALocation field. **[SQL 1] [Syntax]**	1
14. On Table A, how to select items which doesn't contains "oc" anywhere in ALocation field. **[SQL 1] [Syntax]**	1
15. What would be the BigO when you are searching against a primary key /clustered index?	1
16. What would be the BigO when you are searching against a non-primary key?	1
17. What would be the BigO when you are searching against a non-clustered index?	1
18. What would be the BigO when you are add a new item to a table which had 1 non-clustered index?	1
19. In terms of performance which one would be better as primary key? Number or varchar?	1
20. Let's say you have a table named T1 and a column named 'n1' and data type nchar(8). In that row there is value "Hello". How can I get it by simple sql query? **[SQL 1] [Syntax]**	1

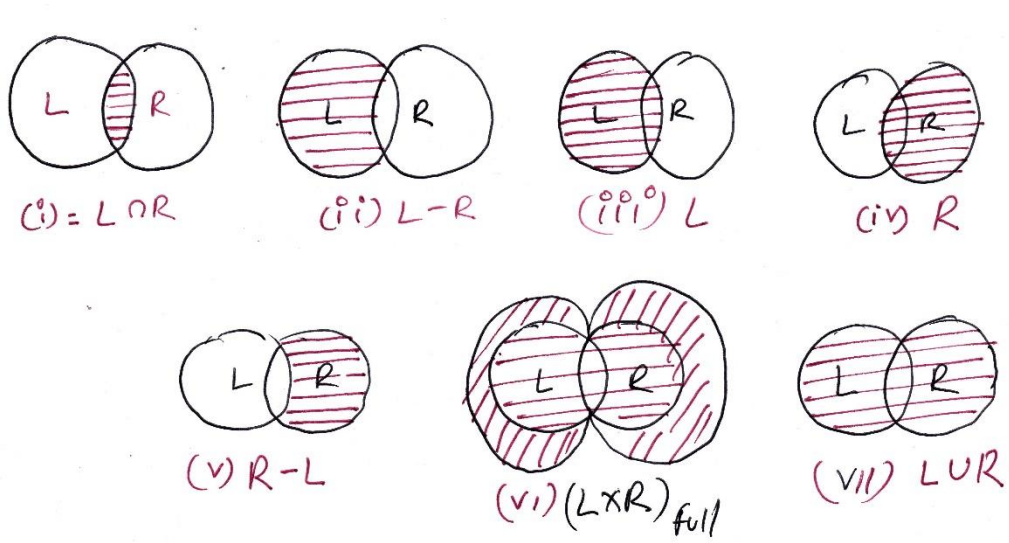
21. SQL :
`SELECT * FROM A`
`WHERE AID BETWEEN 1 AND 3;`
Choose which of those satisfies for this between queries? [MS SQL][Multiple]
a. `SELECT * FROM A WHERE AID >= 1 AND <= 3;`
b. `SELECT * FROM A WHERE AID > 1 AND < 3;`
c. `SELECT * FROM A WHERE AID > 1 AND < 3;`
d. `SELECT * FROM A WHERE AID >= 1 AND AID <= 3;`
e. `SELECT * FROM A WHERE AID >= 1 AND AID < 3;`
	3

f. `SELECT * FROM A WHERE AID > 1 AND AID < 3;` g. `SELECT * FROM A WHERE AID > 1 AND AID =< 3;` h. `It varies from database to database.`	
22. Table "Employee" Find BirthDate where dates are not between 12-Apr-1960 and 12-Apr-2010. **[SQL 1] [Syntax]**	1
23. How to write comments in SQL? Select * FROM Table; // How to write comments like this.	.5
24. Which of those SQL syntax are **incorrect**? **Only BASED on table A,B,C** from above examples. Check out the data sheet. a. `SELECT A.AID + B.BID FROM A,B;` b. `SELECT A.AID + B.BID FROM A,B WHERE A.AID + B.BID < 10;` c. `SELECT COS(A.AID * -1) - B.BID FROM A,B WHERE (A.AID + Sin(B.BID*10)) < 10 AND A.AID > 5;` d. `SELECT A.AID , str(A.BID) + A.ALocation FROM A;` e. `SELECT TOP 1 * FROM WHERE A.AID = B.ID;` f. `SELECT TOP 1 * FROM WHERE A.AID = B.ID ORDER BY B.BID;` g. `SELECT TOP 1 Count(*) FROM A WHERE A.AID = B.ID ORDER BY B.BID;` h. `SELECT TOP 1 Count(*) FROM A,C WHERE A.AID = C.BID ORDER BY C.BID;` i. `SELECT TOP 1 Count(*) FROM A,B WHERE A.AID = B.BID ORDER BY B.BID;` j. `SELECT TOP 100 Count(*) FROM A,B WHERE A.AID = B.BID ORDER BY B.BID;`	5
25. Write a query table name "PersonX" Select column DepartmentID, Average(salary), DepartmentName Where average salary is greater than 5000 Return only front 5 rows.	1
26. Write a based on Table A on page 2. Select unique values from AOID column Where AOID > 0 and AOID <-2 and AOID > -5 AND AOID + AOID = AOID ^ 2	2
27. SELECT substring index 2 to 4 from Alocation on A table (page 2) And add then concat with BID	1

Advance 60 Marks

1. Selects the terms which supports those diagram? — 4.5

A. Intersect B. Union C. Inner join	D. Join E. Left join F. Left outer join	G. Right join H. Right outer join I. L NOT R or R Not L J. FULL Outer Join



2. Sync queries with below results based on A,B tables data. **Hint** use pencil paper first. — 3.5

```
SELECT aTab.AID, btab.AID as BAID, btab.BID, AName, BName
     i. FROM a AS aTab FULL OUTER JOIN b bTab ON(aTab.AID = Btab.AID)
    ii. FROM a AS aTab LEFT OUTER JOIN b bTab ON(aTab.BID = Btab.BID)
   iii. FROM a AS aTab RIGHT OUTER JOIN b bTab ON(aTab.AID = Btab.AID)
    iv. FROM a AS aTab LEFT JOIN b bTab ON (aTab.BID = Btab.BID)
     v. FROM a AS aTab RIGHT JOIN b bTab ON(aTab.AID = Btab.AID)
    vi. FROM a AS aTab Inner JOIN b bTab ON(aTab.BID = Btab.BID)
   vii. FROM a AS aTab JOIN b bTab ON(aTab.BID = Btab.BID)
  viii. FROM a AS aTab , b AS bTab WHERE (aTab.BID = Btab.BID)
```

A Result 1

Results | Messages

	AID	BAID	BID	AName	BName
1	1	1	1	A1	B1
2	2	2	2	A2	B2
3	2	2	3	A2	B3
4	1	1	4	A1	B4
5	3	3	5	A3	B5

B Result 2

Results | Messages

	AID	BAID	BID	AName	BName
1	1	1	1	A1	B1
2	2	NULL	NULL	A2	NULL
3	3	NULL	NULL	A3	NULL
4	4	1	1	A4	B1

C Result 3

Results | Messages

	AID	BAID	BID	AName	BName
1	1	1	1	A1	B1
2	1	1	4	A1	B4
3	2	2	2	A2	B2
4	2	2	3	A2	B3
5	3	3	5	A3	B5
6	4	NULL	NULL	A4	NULL

D Result 4

Results | Messages

	AID	BAID	BID	AName	BName
1	1	1	1	A1	B1
2	4	1	1	A4	B1

3. Table "Employee" **[Partial Marks][MS SQL]** — 12

Name	Age
FIRSTNAME LastName (85)	85
ALIM ~~UL~~ Karim (21)	21
ALICE Wonderland (20.5)	20.50
MARKS Robinson(20)	20

Select FirstName LastName and age calculate from BirthDate Field and represent it as left example. Where age must >= 20. First name should be in upper case form and if contains 'UL' then it should be replaced with empty string. Two columns should be exactly displayed as left. And ages should also be sorted in descending order and rounded to two decimal places.

4. Books table

BookID	AuthorID	BookName	SubID	PublishedDate

Subject table

SubID	Subject

Author table

AuthorID	AuthorName

Rating table

BookID	RatingID	Rate

[Multiple can be answer or maybe only one]

Select SQL for those books which are written by "Dr. Ramesh ….." and has average rating of 2 or more and written in every subject. Sorted by rating in ascending order. Display BookID,BookName, Avgerage rating => DisplayRating.

a)
```
SELECT  R.BookID , B2.BookName , AVG(R.Rate) AS DisplayRating
FROM    dbo.Book B2 , dbo.Author A2 , dbo.Rating
WHERE   B2.AuthorID = A2.AuthorID AND R.BookID = B2.BookID AND
A2.AuthorName LIKE '*Dr. Ramesh%'
GROUP BY R.BookID
HAVING  AVG(R.Rate) >= 2
ORDER BY DisplayRating;
```

b)
```
SELECT  R.BookID , B2.BookName , AVG(R.Rate) AS DisplayRating
FROM    dbo.Book B2 , dbo.Author A2 , dbo.Rating R
WHERE   B2.AuthorID = A2.AuthorID AND R.BookID = B2.BookID AND
A2.AuthorName LIKE 'Dr. Ramesh%'
GROUP BY R.BookID , B2.BookName
HAVING  AVG(R.Rate) >= 2
ORDER BY AVG(R.Rate);
```

c)
```
SELECT  R.BookID , B2.BookName , AVG(R.Rate) AS DisplayRating
FROM    dbo.Book B2 , dbo.Author A2 , dbo.Rating R
WHERE   B2.AuthorID = A2.AuthorID AND R.BookID = B2.BookID AND
A2.AuthorName LIKE 'Dr. Rames_h%'
GROUP BY B2.BookName
HAVING  AVG(R.Rate) >= 2
ORDER BY DisplayRating;
```

d)
```
SELECT  R.BookID , B2.BookName , AVG(R.Rate)
FROM    dbo.Book B2 , dbo.Author A2 , dbo.Rating
WHERE   B2.AuthorID = A2.AuthorID AND R.BookID = B2.BookID AND
A2.AuthorName = 'Dr. Ramesh...'
GROUP BY B2.BookName
HAVING  AVG(R.Rate) <= 2
ORDER BY AVG(R.Rate);
```

e)
```
SELECT  R.BookID , B2.BookName , AVG(R.Rate) AS DisplayRating
FROM    dbo.Book B2 , dbo.Author A2 , dbo.Rating R
WHERE   B2.AuthorID = A2.AuthorID AND R.BookID = B2.BookID AND
A2.AuthorName = 'Dr. Ramesh...'
GROUP BY R.BookID
HAVING  AVG(R.Rate) >= 2
ORDER BY DisplayRating;
```

5

5. Books table

BookID	BookName	SubID	PublishedDate

Subject table

SubID	Subject

Author table

AuthorID	AuthorName	BookID

Rating table

BookID	RatingID	Rate

5

[Multiple can be answer or maybe only one]

Select SQL for those books which are in History or Engineering or Maths Language and has max rating >= 10 and written by more than two author. Display book name , author name , max rate and count of written books by author as Written. Finally sort those in descending order by Written Column.

a)
```
SELECT BookName, AuthorName, COUNT(A2.AuthorID) AS Written
FROM dbo.Book B2
JOIN dbo.Author A2 ON(B2.BookID = A2.BookID)
JOIN dbo.Rating R ON(B2.BookID > R.BookID)
JOIN dbo.Sub S ON(B2.SubID = S.SubID)
ORDER BY Written DESC
WHERE S.Subject IN ('History','Engineering','Maths')
GROUP BY A2.AuthorID,A2.AuthorName, B2.BookName
HAVING  COUNT(A2.AuthorID) > 2 AND MAX(R.Rate) >= 10
```

b)
```
SELECT BookName, AuthorName, MAX(Rate) ,COUNT(A2.AuthorID) AS Written
FROM dbo.Book B
JOIN dbo.Author A2 ON(B2.BookID >= A2.BookID)
JOIN dbo.Rating R ON(B2.BookID = R.BookID)
JOIN dbo.Sub S ON(B2.SubID = S.SubID)
WHERE S.Subject IN ('History','Engineering')  AND  COUNT(A2.AuthorID) >
3 AND MAX(R.Rate) >= 10
GROUP BY A2.AuthorID,A2.AuthorName
ORDER BY Written DESC
```

c)
```
SELECT BookName, AuthorName, MAX(Rate) ,COUNT(A2.AuthorID) AS Written
FROM dbo.Book B2
JOIN dbo.Author A2 ON(B2.BookID = A2.BookID)
JOIN dbo.Rating R ON(B2.BookID = R.BookID)
JOIN dbo.Sub S ON(B2.SubID = S.SubID)
WHERE S.Subject = 'History' OR 'Engineering' OR 'Maths' AND MAX(R.Rate)
>= 10
GROUP BY A2.AuthorID,A2.AuthorName, B2.BookName
ORDER BY Written DESC
```

d)
```
SELECT BookName, AuthorName, MAX(Rate) ,COUNT(A2.AuthorID) AS Written
FROM dbo.Book B2
LEFT JOIN dbo.Author A2 ON(B2.BookID = A2.BookID)
LEFT JOIN dbo.Rating R ON(B2.BookID = R.BookID)
LEFT JOIN dbo.Sub S ON(B2.SubID = S.SubID)
WHERE S.Subject IN ('History','Engineering','Maths')
GROUP BY A2.AuthorID,A2.AuthorName, B2.BookName
HAVING  COUNT(A2.AuthorID) > 2 AND MAX(R.Rate) >= 10
ORDER BY Written DESC
```

e)
```
SELECT BookName, AuthorName, MAX(Rate) ,COUNT(A2.AuthorID) AS Written
FROM dbo.Book B2
JOIN dbo.Author A2 ON(B2.BookID = A2.BookID)
JOIN dbo.Rating R ON(B2.BookID = R.BookID)
JOIN dbo.Sub S ON(B2.SubID = S.SubID)
```

```
WHERE S.Subject IN ('History','Engineering','Maths')
GROUP BY A2.AuthorID,A2.AuthorName, B2.BookName
HAVING  COUNT(A2.AuthorID) > 2 AND MAX(R.Rate) >= 10
ORDER BY Written DESC
```

6. What will be the result of below query? **[Correct][Syntax] [Results]** — 3
Page 2, Table A and B referred. **Try to be smart when writing results, if you have already answered or the answer is already on the question then just refer it. You don't have to write the results all over again.**

```
SELECT AID, btab.AID, AName, BName
FROM a AS aTab JOIN b bTab ON NOT(aTab.AID <> Btab.AID)
```

7. What will be the result of below query? **[Correct][Syntax][Results] [Partial]** — 2
Page 2, Table A and B referred.

```
SELECT aTab.AID, btab.AID, AName, BName
FROM a AS aTab JOIN b bTab ON (A.BID = B.BID)
```

8. What will be the result of below query? **[Correct][Syntax][Results] [Partial]** — 3
Page 2 , Table A and B referred.

```
SELECT aTab.AID, btab.AID, AName, BName
FROM a AS aTab JOIN b bTab ON (A.BID <> Btab.BID)
```

9. What will be the result of below query? **[Correct][Syntax][Results] [Partial]** — 2
Page 2 , Table A and B referred.

```
SELECT aTab.AID, btab.AID, AName, BName
FROM a AS aTab JOIN b bTab ON NOT (aTab.BID <> B.BID)
```

10. Books table **[Partial Marks] [MS SQL]** — 10

BookID	BookName	SubID	PublishedDate

Subject table

SubID	Subject

Author table

AuthorID	AuthorName	BookID

Rating table

BookID	RatingID	Rate

[Multiple can be answer or maybe only one]
Write a query to find books which are published only two months ago and haven't rated yet , have more than 4 authors who already have written at least one book in their past. Show book name, AuthorName , published date and finally rating count which should be zero.

11. SQL : Base on Table A and it's given data on page 2 , Hint use pencil paper test first. **[MS SQL]** — 7
Data is also included in the data sheet.

```
SELECT * FROM A WHERE ALocation IN
(SELECT TOP 2 ALocation FROM A WHERE AOID IN (SELECT AOID FROM A));
```

Choose which of will give exact results or equivalent queries? **[MS SQL Server][Multiple]**

A. `SELECT * FROM A WHERE ALocation IN (SELECT TOP 2 AOID FROM A);`
B. `SELECT * FROM A A1, A A2 WHERE AOID IN(Select TOP 2 AOID FROM A);`
C. `SELECT TOP 2 A1.* FROM A A1, A A2 WHERE A1.AOID = A2.AOID AND A1.AOID IN (Select TOP 2 AOID FROM A);`
D. `SELECT TOP 2 B.* FROM A WHERE AOID <= AND (SELECT TOP 2 AOID FROM A);`
E. `SELECT TOP 2 C.* FROM A WHERE AOID >= AND (SELECT TOP 2 AOID FROM A);`

F. `SELECT * FROM A WHERE AOID < ANY (SELECT TOP 2 AOID FROM A);`
G. `SELECT TOP 2 * FROM A WHERE AOID <= ANY (SELECT TOP 2 AOID FROM A);`
H. `SELECT TOP 2 A1.* FROM A A1 INNER JOIN A A2 ON (A1.AID = A2.AID AND 1 = ANY(SELECT 1 WHERE A2.AOID <= 2));`

12. Is it correct? If not then correct it? Show the exact result based on the code and **Table A** from page 2. **[MS SQL Only]** — 1+1+1

```
DECLARE @xp AS INT, @yp AS INT

SELECT @Xp = (SELECT TOP 1 AOID FROM A ORDER BY AOID DESC);
SELECT @Yp = (SELECT TOP 1 AOID FROM A ORDER BY AOID DESC);

SELECT
      CASE
            WHEN @xp = 1 THEN STR(@xp) + ' = Found One'
            WHEN @xp = @yp THEN STR(@xP) + ' = Found it'
            ELSE 'Not Found'
            END AS 'Result';

SELECT @Xp = (SELECT TOP 1 AOID FROM A WHERE AOID = 1 );
SELECT @Yp = (SELECT TOP 1 AOID FROM A ORDER BY AOID ASC);

SELECT
      CASE
            WHEN @xp = 1 THEN STR(@xp) + ' = Found One'
            WHEN @xp = @yp THEN STR(@xP) + ' = Found it'
            ELSE 'Not Found'
            END AS 'Result';

SELECT @Xp = (SELECT TOP 1 AOID FROM A WHERE AOID = 1 );
SELECT @Yp = (SELECT TOP 1 AOID FROM A ORDER BY AOID DESC);

SELECT
      CASE
            WHEN @xp = 1 THEN STR(@xp) + ' = Found One'
            WHEN @xp = @yp THEN STR(@xP) + ' = Found it'
            ELSE 'Not Found'
            END AS 'Result';
```