Regular Expression in SQL Server

Madhivanan, MVP (SQL Server)
madhivanan2001@gmail.com

Professional Association for SQL Server

Definition

Regular expressions provide better way of matching strings of text, such as particular characters, words, or patterns of characters.

Also referred as

* Regex
* Regexp

How to use Regular expression in SQL Server?

* **A bracket expression.**
* [abc] matches "a", "b", or "c". [a-z] specifies a range which matches any lowercase letter from "a" to "z". These forms can be mixed: [abcx-z] matches "a", "b", "c", "x", "y", or "z", as does [a-cx-z].
* [ABC] matches “A", “B", or “C". [A-Z] specifies a range which matches any uppercase letter from “A" to “Z". These forms can be mixed: [ABCX-Z] matches “A", “B", “C", “X", “Y", or “Z", as does [A-CX-Z].
* [0-9] matches any digit from 0 to 9
* [67] matches digit either 6 or 7
* [^g] matches any character other than g
* [^7-9] matches any character other than digits 7,8 and 9

Pseudo code

* **To check if data has first four characters as alphabets**

  where col like '[a-z][a-z][a-z][a-z]%‘

* **To check if data has first four characters as alphabets followed by two digits followed by alphabet S**

  where col like '[a-z][a-z][a-z][a-z][0-9][0-9][S]%'

* **To check if data ends with alphabet**

  where col like '%[a-z]'

Examples with sample data

* **To check if data has atleast a digit**

  where col like '%[0-9]%’

* **Only numerics**

  where data not like '%[^0-9]%‘

Example code

```
declare @sample_table table (data varchar(100) primary key)
insert into @sample_table (data)
select 'This is my new car' union all
select 'Be careful' union all
select 'This red car is very nice' union all
select 'carpets are there' union all
select 'Nothing' union all
select 'Ok. Thank you' union all
select 'My address is No 34, New Main Road, Chennai - 600045' union all
select 'Please call me at 9444572716' union all
select '9444572716'
```

Query

Select * from @sample_table

Result

This is my new car
Be careful
This red car is very nice
carpets are there
Nothing
Ok. Thank you
My address is No 34, New Main Road, Chennai - 600045
Please call me at 9444572716
9444572716

--Give data that has the word car

```
select * from @sample_table
where data like '%car%‘
```

Result

```
----------------------------------------------
This is my new car
Be careful
This red car is very nice
carpets are there
```

--Give data that has car immediately followed by atleast 4 characters

```
select * from @sample_table
where data like '%car[a-z][a-z][a-z][a-z]%‘
--or
select * from @sample_table
where data like '%car[a-za-za-za-z]%'
```

Result

```
----------------------------------------------------------
Be careful
carpets are there
```

--Get data that has atleast a digit

```
select * from @sample_table
where data like '%[0-9]%'
```

Result

```
-------------------------------------------------------------
My address is No 34, New Main Road, Chennai - 600045
Please call me at 9444572716
9444572716
```

--Get data that has not only digits but any characters too

```
select * from @sample_table
where data like '%[^0-9]%‘
```

Result

```
----------------------------------------------------------------------
This is my new car
Be careful
This red car is very nice
carpets are there
Nothing
Ok. Thank you
My address is No 34, New Main Road, Chennai - 600045
```

--Get data that has only digits

```
select * from @sample_table
where data not like '%[^0-9]%‘
```

Result

9444572716

More examples
Example1

Extract data that has number in the format
Ddd-ddd-dddd (d denotes a number from 0 to 9)

```
declare @sample_data table(data varchar(100))
insert into @sample_data(data)
select 'I am on the way. Note my number is 87-883-0114 and right now.....' as data union
    all
select 'Well done. Contact this number 345-245-9871 and let me know.....'

Select * from @sample_data
```

Result

I am on the way. Note my number is 87-883-0114 and right now.....
Well done. Contact this number 345-245-9871 and let me know.....

Query

```
select
      data
from
    @sample_data
where
      patindex('%[0-9][0-9][0-9][-][0-9][0-9][0-9][-][0-9][0-9][0-9][0-
    9]%',data)>0
```

Result

Well done. Contact this number 345-245-9871 and let me know.....

* **Extracting the contact number**

```
select
      substring(data,patindex('%[0-9][0-9][0-9][-][0-9][0-9][0-9][-][0-9][0-9][0-
    9][0-9]%',data),12) as ph_no
from
    @sample_data
where
      patindex('%[0-9][0-9][0-9][-][0-9][0-9][0-9][-][0-9][0-9][0-9][0-9]%',data)>0
```

Result

345-245-9871

Example 2

Extract amount prefixed by $

```
declare @sample_data table(data varchar(100))
insert into @sample_data(data)
select '9 Lemons cost 67 $99.99 on sale' as fruit union all
select '$5.99 Apples 877 are on sale' union all
select 'Where are the $65.99 lemons 7856' union all
select 'Oranges costs $99.5' union all
select ' and this costs 98.24'
```

Select * from @sample_data

Result

---

9 Lemons cost 67 $99.99 on sale
$5.99 Apples 877 are on sale
Where are the $65.99 lemons 7856
Oranges costs $99.5
and this costs 98.24

Get data that has $ as part of it

```
select
                        data,
                        substring(data,charindex('$',data),len(data)) as amount
from
                        @sample_data
where
                        data like '%$%'
```

Result

Data	amount
9 Lemons cost 67 $99.99 on sale	$99.99 on sale
$5.99 Apples 877 are on sale	$5.99 Apples 877 are on sale
Where are the $65.99 lemons 7856	$65.99 lemons 7856
Oranges costs $99.5	$99.5

Get the amount

```
select data,substring(amount,1,patindex('%[0-9][ ]%',amount+' ')) as amount from
(
    select
                    data,
                    substring(data,charindex('$',data),len(data)) as amount
    from
                    @sample_data
    where
                    data like '%$%'
) as t
```

Result

```
-------------------------------------------------------------------
9 Lemons cost 67 $99.99 on sale          $99.99
$5.99 Apples 877 are on sale             $5.99
Where are the $65.99 lemons 7856                    $65.99
Oranges costs $99.5                      $99.5
```

Benefits of Regular expression

* Better way of pattern matching
* Less coding
* Index usage which improves performance

Less Coding

--Give data that starts with four lowercase alphabets

Without using Regular expression

```
select * from @sample_table
where substring(data,1,1) in ('a','b','c',…..'z') And  substring(data,2,1) in
    ('a','b','c',…..'z')
And  substring(data,3,1) in ('a','b','c',…..'z') And  substring(data,4,1) in
    ('a','b','c',…..'z')
```

Result

This is my new car

This red car is very nice

carpets are there

Nothing

Please call me at 9444572716

Using Regular Expression

```
select * from @sample_table
where data like '[a-z] [a-z] [a-z] [a-z]%‘
```

Result

```
----------------------------------------------------
This is my new car
This red car is very nice
carpets are there
Nothing
Please call me at 9444572716
```

Index usage

Get data where first character is b, c or t

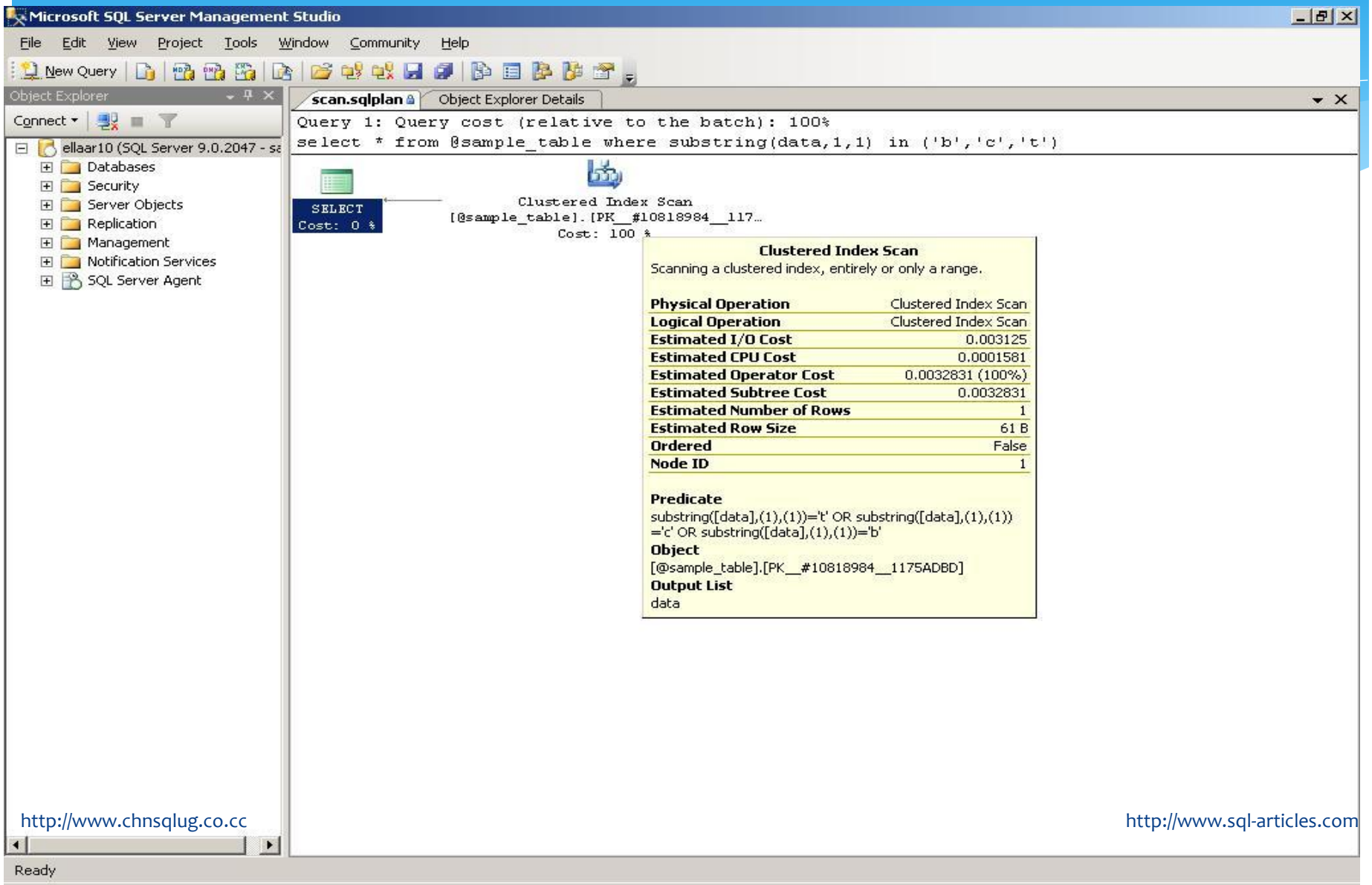
Usual method

```
select * from @sample_table
where substring(data,1,1) in ('b','c', 't')
```

Result

```
-------------------------------------------
This is my new car
Be careful
This red car is very nice
carpets are there
```

Table Scan



http://www.chnsqlug.co.cc

http://www.sql-articles.com

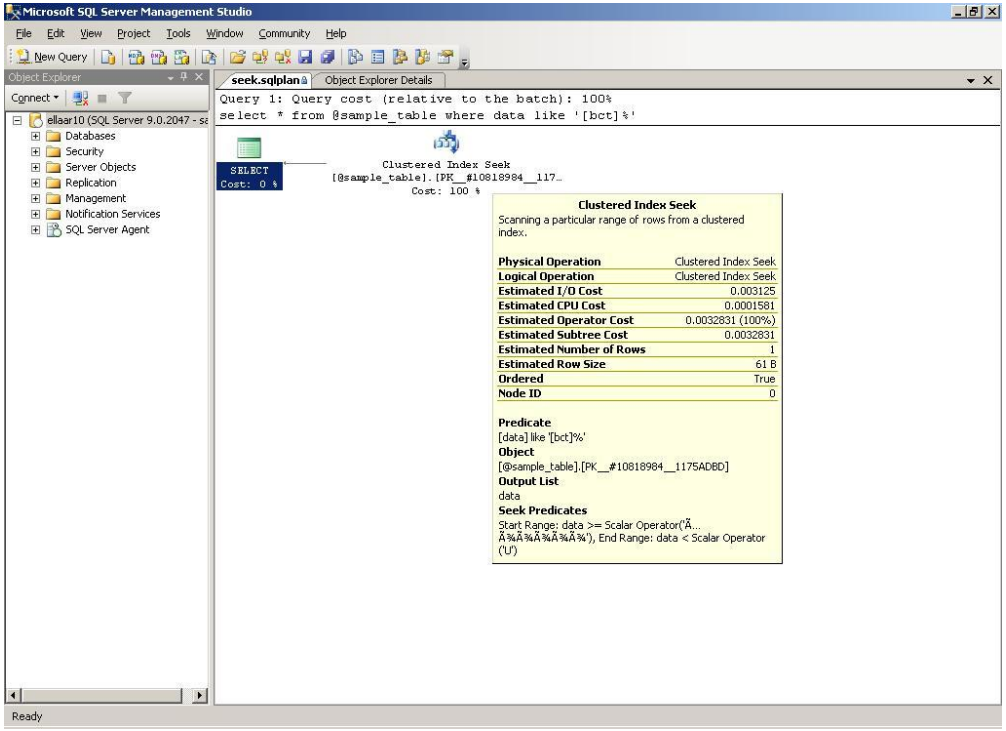
Regular expression

```
select * from @sample_data
where data like '[bct]%‘
```

Result

```
----------------------------------------------
This is my new car
Be careful
This red car is very nice
carpets are there
```

Table Seek


Microsoft SQL Server Management Studio
File Edit View Project Tools Window Community Help
New Query
Object Explorer
Connect
ellaar10 (SQL Server 9.0.2047 - sa
Databases
Security
Server Objects
Replication
Management
Notification Services
SQL Server Agent
seek.sqlplan
Object Explorer Details
Query 1: Query cost (relative to the batch): 100%
select * from @sample_table where data like '[bct]%'
SELECT
Cost: 0 %
Clustered Index Seek
[@sample_table].[PK__#10818984__117...
Cost: 100 %
Clustered Index Seek
Scanning a particular range of rows from a clustered index.
Physical Operation Clustered Index Seek
Logical Operation Clustered Index Seek
Estimated I/O Cost 0.003125
Estimated CPU Cost 0.0001581
Estimated Operator Cost 0.0032831 (100%)
Estimated Subtree Cost 0.0032831
Estimated Number of Rows 1
Estimated Row Size 61 B
Ordered True
Node ID 0
Predicate
[data] like '[bct]%'
Object
[@sample_table].[PK__#10818984__1175ADBD]
Output List
data
Seek Predicates
Start Range: data >= Scalar Operator('Â...
Â¾Â¾Â¾Â¾Â¾Â¾'), End Range: data < Scalar Operator
('Ù')
Ready

Thank You