REPORT #3

# Problem

Optimization of queries/procedures

We should chose few queries/procedures and try to optimize them.

# Query 1

declare @iterator int = 1;

declare @tempSNAME table (tempIndex int identity(1,1), sName nvarchar(15))

declare @tempSNUMBEROFSTUDENTS table (tempIndex int identity(1,1), sNumberOfStudents int)

declare @tempSTOWNID table (tempIndex int identity(1,1), sTownId int)

while @iterator <> 1001

begin

insert @tempSNAME

exec spdCreateRandomString 20

insert @tempSNUMBEROFSTUDENTS

exec spdCreateRandomInt

insert @tempSTOWNID

select top 1 townID from Towns

order by newid()

set @iterator = @iterator + 1

end

insert into Schools(schoolName, schoolNumberOfStudents, townID)

select t.sName, t2.sNumberOfStudents, t3.sTownId

from @tempSNAME t

inner join @tempSNUMBEROFSTUDENTS t2 on t2.tempIndex = t.tempIndex

inner join @tempSTOWNID t3 on t3.tempIndex = t.tempIndex

select \* from Schools

GO

Ideas to optimize it

*Problem:* There are 3 temporary tables

*Solution:* use variables, or normal tables

During my research I found this information:

In 99% of cases, table variables reside in memory, hence it is a lot faster. Temporary tables reside in the TempDb database. So operating on temporary tables require inter database communication and hence will be slower.

*Problem 2:* Aliases

*Solution:* remove them and use full names

It shouldn’t mean much but it’s a way to optimize it too

# Query 2

create procedure spdCreateRandomEmail

as

begin

declare @mailName nvarchar(8)

declare @hostName nvarchar(3)

declare @mailEnding nvarchar(2)

declare @char1 nvarchar(1)

declare @char2 nvarchar(1)

declare @mailLength int

declare @mailHostLenght int

declare @mailEndingLenght int

set @mailName = ''

set @mailEnding = ''

set @hostName = ''

set @char1 = '@'

set @char2 = '.'

set @mailLength = 8

set @mailHostLenght = 3

set @mailEndingLenght = 2

while @mailLength <> 0

begin

select @mailName = @mailName + char(round(rand() \* 26, 1) + 97)

set @mailLength = @mailLength - 1

end

while @mailHostLenght <> 0

begin

select @hostName = @hostName + char(round(rand() \* 26, 1) + 97)

set @mailHostLenght = @mailHostLenght - 1

end

while @mailEndingLenght <> 0

begin

select @mailEnding = @mailEnding + char(round(rand() \* 26, 1) + 97)

set @mailEndingLenght = @mailEndingLenght - 1

end

select @mailName + @char1 + @hostName + @char2 + @mailEnding

end

go

Ideas to optimize it

*Problem:* there are 3 loops and a lot of variables

*Solution:* we can remove 2 of those loops and set constant email values after @ character.

It would remove 2 loops, rand() calls and some variables while still providing randomized email address

# Query 3

select \* from Schools

join Towns on Towns.townID = Schools.townID

where townPopulation < 100

and numberOfStudents % 2 = 0

Shorter one

Ideas to optimize it

*Problem:*  select \* from

*Solution:* select SchoolID, schoolName, numberofstudents, Schools.townID, townName, townPopulation, townMayor

It is better practice to write select *all column names* than select \*

select SchoolID, schoolName, numberofstudents, Schools.townID, townName, townPopulation, townMayor from Schools

join Towns on Towns.townID =Schools.townID

where townPopulation < 100

and numberOfStudents % 2 = 0

# Query 4

select numberOfStudents from schools where 0 < (select count(\*) from Towns where townPopulation < 1000)

Ideas to optimize it

*Problem:* using count(\*) and 0< for existence checking

*Solution:* use exists and select \*

select numberOfStudents from schools where exists (select \* from towns where townPopulation < 1000)

# Query 5

select \* from Schools where numberOfStudents < 3000 or townID > 987

Ideas to optimize it

*Problem:* using or on two different columns

*Solution:* use union

select \* from Schools where numberOfStudents < 3000

union

select\* from Schools where townID > 987