CREATE DATABASE My\_BASE

GO

USE My\_BASE

Create SEQUENCE [dbo].[seq\_for\_city]

as int

start with 1

Increment by 1

Create SEQUENCE [dbo].[seq\_for\_developer]

as int

start with 1

Increment by 1

Create SEQUENCE [dbo].[seq\_for\_company]

as int

start with 1

Increment by 1

CREATE TABLE [dbo].[City](

[ID] int

,[Name] nvarchar(60)

)

CREATE TABLE [dbo].[Company](

[ID] int

,[Name] nvarchar(60)

)

CREATE TABLE [dbo].[Developer] (

[ID] int

,[Name] nvarchar(60)

,[Level] int

,[ExperienceInYears] int

,[Company] int

,[ActualCity] int

)

CREATE TABLE [dbo].[DeveloperLanguage] (

[ID] int IDENTITY(1,1)

,[Language] nvarchar (30)

,[Developer] int

)

CREATE TABLE [dbo].[DeveloperLanguageTag](

[ID] int IDENTITY(1,1)

,[DeveloperLanguage] int

,[IsWriter] int

,[Tag] nvarchar(20)

)

CREATE TABLE [dbo].[DBMS](

[ID] int IDENTITY(1,1) ,

[Name] nvarchar(50)

)

CREATE TABLE [dbo].[DeveloperDBMS](

[ID] int Identity(1,1)

,[Developer] int

,[Version] nvarchar(15)

,[DBMS] int

)

CREATE TABLE [dbo].[DeveloperDBMSTag](

[ID] int Identity(1,1)

,[DeveloperDBMS] int

,[Tag] nvarchar(20)

)

--drop PROC [dbo].[input\_developer]

CREATE PROC [dbo].[input\_developer]

@DATA NVARCHAR(max)

as

declare @Name nvarchar(50) = JSON\_VALUE(@DATA,'$.Name')

declare @Level int = JSON\_VALUE(@DATA,'$.Level')

declare @ExperienceInYears int = JSON\_VALUE(@DATA,'$.ExperienceInYears')

declare @City nvarchar(50) = JSON\_VALUE(@DATA,'$.City')

declare @Company nvarchar(100) = JSON\_VALUE(@DATA,'$.Company')

DECLARE @Developer\_ID int = next value for [dbo].[seq\_for\_developer]

declare @City\_id int

select @City\_id=ID from [dbo].[City] where Name = @City

declare @Company\_id int

select @Company\_id=ID from [dbo].[Company] where Name = @Company

Begin TRY

SET NOCOUNT ON

BEGIN TRAN

if @City\_id is null

begin

SEt @City\_id = next value for [dbo].[seq\_for\_city]

insert [dbo].[City]

select @City\_id, @City

end

if @Company\_id is null

begin

SEt @Company\_id = next value for [dbo].[seq\_for\_Company]

insert [dbo].[Company]

select @Company\_id, @Company

end

INSERT [dbo].[Developer]

select @Developer\_ID,@Name,@Level,@ExperienceInYears,@Company\_id,@City\_id

DECLARE @temp\_language\_tag TABLE ([Language] NVARCHAR(50), [ISWriter] int, [DeveloperLanguageTag] NVARCHAR(50))

INSERT @temp\_language\_tag

select json\_value(T.value,'$.Language') as Language,

json\_value(T.value,'$.IsWriter') as ISWriter,

B.value as DeveloperLanguageTag

from OPENJSON(JSON\_QUERY(@DATA,'$.DeveloperLanguage')) as T

cross apply OPENJSON(JSON\_QUERY(@DATA,'$.DeveloperLanguage['+CAST(T.[key] as nvarchar(2))+'].DeveloperLanguageTag')) as B

INSERT [dbo].[DeveloperLanguage]

SELECT distinct(Language),@Developer\_ID from @temp\_language\_tag

INSERT [dbo].[DeveloperLanguageTag]

select a.ID, b.ISWriter,b.DeveloperLanguageTag

from [DeveloperLanguage] as a

join @temp\_language\_tag as b

on a.Developer =@Developer\_ID and a.Language=b.Language

DECLARE @temp\_DBMS\_tag TABLE ([DBMS] NVARCHAR(50), Versions NVARCHAR(50),[Tag] NVARCHAR(50))

INSERT @temp\_DBMS\_tag

select json\_value(T.value,'$.DMBS') as DBMS,

B.value as Versions,

C.value as Tag

from OPENJSON(JSON\_QUERY(@DATA,'$.DeveloperDBMS')) as T

cross apply OPENJSON(JSON\_QUERY(@DATA,'$.DeveloperDBMS['+CAST(T.[key] as nvarchar(2))+'].Versions')) as B

cross apply OPENJSON(JSON\_QUERY(@DATA,'$.DeveloperDBMS['+CAST(T.[key] as nvarchar(2))+'].DevepolerDBMSTag')) as C

INSERT [dbo].[DBMS]

SELECT distinct(DBMS) from @temp\_DBMS\_tag

where [DBMS] not in (select [Name] from [dbo].[DBMS])

INSERT [dbo].[DeveloperDBMS]

select @Developer\_ID,A.Versions,B.ID

from @temp\_DBMS\_tag as A

join [dbo].[DBMS] as B

on B.[Name]=A.DBMS

INSERT [dbo].[DeveloperDBMSTag]

select A.ID,B.Tag

from [DeveloperDBMS] as A

join [dbo].[DBMS] as C

on A.DBMS=C.ID

join @temp\_DBMS\_tag as B

on A.Version=B.Versions

where A.Developer=@Developer\_ID

Commit

Set NOCOUNT OFF;

END TRY

BEGIN CATCH

Rollback TRANSACTION;

Set NOCOUNT OFF;

THROW;

END CATCH

DECLARE @Person\_1 NVARCHAR(4000)=

N'{

"Name":"Миша",

"Level":1,

"ExperienceInYears":2,

"City":"Казань",

"Company":"Maxima",

"DeveloperLanguage":[

{"Language":"C#",

"IsWriter":1,

"DeveloperLanguageTag":[".Net","ASP.Net"]

},

{"Language":"Python",

"IsWriter":1,

"DeveloperLanguageTag":["Numpy","Pandas"]

},

{"Language":"T-SQL",

"IsWriter":1,

"DeveloperLanguageTag":["index","lock"]

}

],

"DeveloperDBMS":[

{

"DMBS":"MS\_SQL",

"Versions":["2005","2008","2016"],

"DevepolerDBMSTag":["admin","OLTP"]

},

{

"DMBS":"My SQL",

"Versions":["5.0","6.0"],

"DevepolerDBMSTag":["admin"]

}

]

}'

DECLARE @Person\_2 NVARCHAR(4000)=

N'{

"Name":"Ильнур",

"Level":2,

"ExperienceInYears":2,

"City":"Москва",

"Company":"Yandex",

"DeveloperLanguage":[

{"Language":"C++",

"IsWriter":1,

"DeveloperLanguageTag":[".Net","ASP.Net"]

},

{"Language":"JavaScript",

"IsWriter":1,

"DeveloperLanguageTag":["NOde.js","Angular"]

},

{"Language":"T-SQL",

"IsWriter":1,

"DeveloperLanguageTag":["index","lock","view","procedure"]

}

],

"DeveloperDBMS":[

{

"DMBS":"MS\_SQL",

"Versions":["2005","2008","2016"],

"DevepolerDBMSTag":["optimization","OLTP"]

},

{

"DMBS":"My SQL",

"Versions":["5.0","5.5"],

"DevepolerDBMSTag":["admin","optimization"]

},

{

"DMBS":"PostgreSQL",

"Versions":["All"],

"DevepolerDBMSTag":["admin","optimization"]

}

]

}'

DECLARE @Person\_3 NVARCHAR(4000)=

N'{

"Name":"Максим",

"Level":3,

"ExperienceInYears":4,

"City":"Новосибирск",

"Company":"Sovcombank",

"DeveloperLanguage":[

{"Language":"C#",

"IsWriter":1,

"DeveloperLanguageTag":[".Net","ASP.Net"]

},

{"Language":"JavaScript",

"IsWriter":1,

"DeveloperLanguageTag":["NOde.js","Angular"]

},

{"Language":"T-SQL",

"IsWriter":1,

"DeveloperLanguageTag":["index","lock","view","procedure","function","triggers"]

}

],

"DeveloperDBMS":[

{

"DMBS":"MS\_SQL",

"Versions":["2005","2008","2016","2017","2019"],

"DevepolerDBMSTag":["optimization","OLTP"]

},

{

"DMBS":"DB2",

"Versions":["3.0","4.0"],

"DevepolerDBMSTag":["admin","optimization"]

},

{

"DMBS":"PostgreSQL",

"Versions":["All"],

"DevepolerDBMSTag":["admin","optimization"]

},

{

"DMBS":"Oracle Database",

"Versions":["All"],

"DevepolerDBMSTag":["admin","optimization"]

}

]

}'

DECLARE @Person\_4 NVARCHAR(4000)=

N'{

"Name":"Наиль",

"Level":4,

"ExperienceInYears":3,

"City":"Казань",

"Company":"SkyEng",

"DeveloperLanguage":[

{"Language":"Python",

"IsWriter":1,

"DeveloperLanguageTag":["NumPy","Pandas"]

},

{"Language":"JavaScript",

"IsWriter":1,

"DeveloperLanguageTag":["GoogleJS","Node.js"]

},

{"Language":"T-SQL",

"IsWriter":1,

"DeveloperLanguageTag":["index","lock","view","procedure","trigger","fucntion","sequnce"]

}

],

"DeveloperDBMS":[

{

"DMBS":"MS\_SQL",

"Versions":["2005","2008","2012","2014","2016"],

"DevepolerDBMSTag":["optimization","OLTP","admin","lock"]

},

{

"DMBS":"My SQL",

"Versions":["5.0","5.5"],

"DevepolerDBMSTag":["admin","optimization"]

},

{

"DMBS":"PostgreSQL",

"Versions":["All"],

"DevepolerDBMSTag":["admin","optimization"]

}

]

}'

--exec input\_developer @Person\_1

--exec input\_developer @Person\_2

--exec input\_developer @Person\_3

--exec input\_developer @Person\_4

/\*

select \* from City

select \* from Company

select \* from DeveloperLanguage

select \* from DeveloperLanguageTag

select \* from Developer

select \* from DBMS

select \* from DeveloperDBMS

select \* from DeveloperDBMSTag

\*/

with Candidate as (

select d.Name as DeveloperName, d.[Level],

(

case

when exists (select \* from dbo.DeveloperLanguage as dl where dl.[Language] in ('C#', 'Java', 'Delphi')) then 1

else 0

end +

iif(ddbt.Id is null, 0, 1) +

iif(dlt.Id is null, 0, 1)

) as Advantage

--,ROW\_NUMBER() over (partition by d.Name order by [Level] + Advantage desc) as position

from dbo.Developer as d

inner join dbo.Company as c on c.Id = d.Company

inner join dbo.City as ct on ct.Id = d.ActualCity

inner join dbo.DeveloperDBMS as ddb on ddb.Developer = d.Id

inner join dbo.DeveloperLanguage as dl on dl.Developer = d.Id and dl.[Language] = 'T-SQL'

inner join dbo.DBMS as db on db.Id = ddb.DBMS

left join dbo.DeveloperDBMSTag as ddbt on ddbt.DeveloperDBMS = ddb.Id and ddbt.Tag = 'AlwaysOn'

left join dbo.DeveloperLanguageTag as dlt on dlt.DeveloperLanguage = dl.Id and dlt.IsWriter = 1 and dlt.Tag = 'MOT'

where

ct.Name = 'Казань' and c.Name != 'Maxima' and

db.Name = 'MS\_SQL' and ddb.[Version] in ('2008', '2012', '2014', '2016') and

(select count(\*) from dbo.DeveloperDBMSTag as ddbt where ddbt.DeveloperDBMS = ddb.Id and ddbt.Tag in ('OLTP', 'lock', 'transaction', 'optimization', 'admin')) >= 4 and

(

select count(\*)

from dbo.DeveloperLanguageTag as dlt

where dlt.DeveloperLanguage = dl.Id and dlt.IsWriter = 1 and dlt.Tag in ('index', 'procedure', 'function', 'trigger', 'view')

) > 3 and

d.ExperienceInYears >= 2

),

candidate\_with\_position as (

select \*

,ROW\_NUMBER() over (partition by DeveloperName order by [Level] + Advantage desc) as position

from Candidate

)

select DeveloperName from candidate\_with\_position where position=1