Cognome \_\_\_\_\_\_\_\_\_\_\_\_\_\_ Nome \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Data \_\_\_\_/\_\_\_\_/\_\_\_\_\_

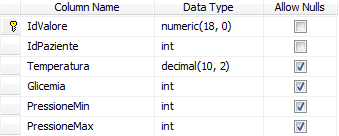
Esercitazione sulle UDF di MS SQL Server 2016

**“RepartoOspedale”**

# classe 5.ta G – a.s. 2017/2018

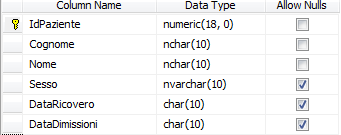
Si consideri il database *RepartoOspedale*  e le tabelle seguenti con i relativi contenuti.

Tabella ***Valori***

****

## TABELLE

Tabella ***Pazienti***



Si implementino le **UDF** sotto elencate. Per ciascuna UDF si fornisca un esempio di utilizzo.

1. Creare una UDF (*f\_sostituisci\_carattere\_slash*) che:
2. analizzi il contenuto di una stringa specificata secondo il formato ‘*gg/mm/aaaa’* (con *g,m,a* caratteri corrispondenti alle cifre numeriche cmprese tra 0🡪9 ) passata come **parametro** ritornando il **codice d’errore *-1*** qualora i caratteri *g,m,a* non fossero *numerici*
3. nel caso in cui **non** si verifichi la condizione d’**errore**, ritorni la stessa stringa privata di un eventuale carattere separatore collocato al posto del carattere ‘/’, sostituendolo con il carattere ‘/’ (es. ’27\_11:2016’ 🡪 ’27/11/2016’)

Scrivere la query che consenta di utilizzare l‘UDF *f\_sostituisci\_carattere\_slash* per sostituire (dove non ci siano altri errori), nella tabella ***Pazienti***, tutti i caratteri separatori diversi da ‘/’ con il carattere ‘/’ (campi ***DataRicovero*** e ***DataDimissioni***)

**Nota**: relativamente al punto a) si implementi solo una parte della soluzione, ma sufficiente per dare l’idea di come procedere. La soluzione completa deve anche prevedere la gestione di valori in input *null* ; in questo caso si ritorni il **codice d’errore *-2***

1. Creare una UDF (*f\_determina\_sesso*) che, applicando i seguenti criteri basati sull’analisi di una stringa passata come argomento, ritorni il caratter ‘**M**’ (*Maschio*) oppure ‘**F**’ (*Femmina*) ad indicare il sesso della persona, determinato dall’analisi della stringa che ne rappresenta il *nome*.

**Criteri** di riferimento per i nomi ***maschili***:

1. La stringa finisce con una tra le seguenti vocali: ‘o’,’i’,’e’ (es. *‘Marco’, ‘Gianni’, ‘Simone’*)
2. **Eccezioni** (da gestire): ‘*Andrea’, ‘Luca’*

**Nota**: tutti gli altri nomi devono intendersi ***femminili***

Scrivere la query che consenta di utilizzare l‘UDF *f\_determina\_sesso* per sostituire, nella tabella ***Pazienti***, per sostituire le eventuali **omissioni** del **sesso** (*null*) o presenza di valori non consentiti (diversi da ‘**M**’ o ‘**F**’)

con uno tra i valori consentiti (‘**M**’ o ‘**F**’) ritornato dall’esecuzione della UDF.

1. Risolvere, tramite l’applicazione di opportune UDF, la problematica del possibile inserimento errato dei valori di *pressione* nella tabella ***Valori***. In particolare si vuole ottenere lo **scambio automatico tramite query**, dei valori ***PressioneMin*** e ***PressioneMax,*** qualora questi valori fossero stati scambiati erroneamente in fase di data-entry (il minimo al posto del massimo e viceversa).

**SOLUZIONI**

**UDF 1.**

use RepartoOspedale

UPDATE Pazienti

SET DataRicovero=dbo.f\_sostituisci\_carattere\_slash(DataRicovero)

WHERE dbo.f\_sostituisci\_carattere\_slash(DataRicovero)<> '-1'

GO

CREATE FUNCTION dbo.f\_sostituisci\_carattere\_slash

(@data AS char(10))

RETURNS char(10)

AS

BEGIN

IF (

(

ASCII(Substring(@data,1,1))<>48 AND

ASCII(Substring(@data,1,1))<>49 AND

ASCII(Substring(@data,1,1))<>50

) OR

(

ASCII(Substring(@data,4,1))<>48 AND

ASCII(Substring(@data,2,1))<>49 AND ASCII(Substring(@data,2,1))<>50

) OR

(

ASCII(Substring(@data,4,1))<>48 AND

ASCII(Substring(@data,4,1))<>49 AND

ASCII(Substring(@data,4,1))<>50

)

)

BEGIN

RETURN '-1'

END

SET @data=Replace(@data,Substring(@data,3,1),'/')

SET @data=Replace(@data,Substring(@data,6,1),'/')

RETURN @data

END

go

**UDF 2**

CREATE FUNCTION dbo.f\_determina\_sesso (@nome AS nchar(10))

RETURNS nchar(10)

AS

BEGIN

IF (

Substring(@nome,len(@nome),1)<>'o' AND

Substring(@nome,len(@nome),1)<>'i' AND

@nome <>'Andrea' AND

@nome <>'Luca'

)

RETURN 'F'

ELSE

BEGIN

RETURN 'M'

END

RETURN 'E' -- Condizione mai raggiunta

END

go

use RepartoOspedale

UPDATE Pazienti

SET Sesso=dbo.f\_Determina\_Sesso(Nome)

WHERE Pazienti.Sesso NOT IN ('M', 'F') AND

dbo.f\_Determina\_Sesso(Nome) IN ('M', 'F')

GO

**UDF 3**

CREATE FUNCTION dbo.f\_max (@val1 AS int, @val2 AS int)

RETURNS int

AS

BEGIN

IF ( @val1 >= @val2 )

RETURN @val1

ELSE

RETURN @val2

RETURN 0 -- Condizione mai raggiunta

END

go

CREATE FUNCTION dbo.f\_min (@val1 AS int, @val2 AS int)

RETURNS int

AS

BEGIN

IF ( @val1 < @val2 )

RETURN @val1

ELSE

RETURN @val2

RETURN 0 -- Condizione mai raggiunta

END

go

use RepartoOspedale

UPDATE Valori

SET PressioneMin=dbo.f\_min(PressioneMin, PressioneMax),

PressioneMax=dbo.f\_max(PressioneMin, PressioneMax)

GO