**Zadanie**

Przyjmujemy, że w bazie danych rejestrują się kolejne osoby, podając swoje imię. Automatycznie są rejestrowane także: data oraz dokładny czas rejestracji. Dane te stopniowo zapełniają następującą tabelę:

|  |  |
| --- | --- |
| **Imię** | **Data i godzina wpisu** |
| NULL | 20160101 20:36:12 |
| ADAM | 20160101 20:38:27 |
| Zbyszek | 20160101 20:39:02 |
| ADAM | 20160101 20:41:55 |
| ADAM | 20160101 20:42:31 |
| ADAM | 20160101 20:51:11 |
| ADAM | 20160101 20:51:24 |
| NULL | 20160101 20:52:01 |
| Zbyszek | 20160101 20:53:07 |
| Zbyszek | 20160101 20:53:44 |
| Renata | 20160101 20:54:52 |
| NULL | 20160101 20:54:57 |
| Zygmunt | 20160101 20:54:59 |
| NULL | 20160101 20:55:03 |
| NULL | 20160101 20:56:12 |
| Renata | 20160101 20:57:22 |
| Zygmunt | 20160101 20:58:31 |
| Zygmunt | 20160101 20:59:41 |
| Zygmunt | 20160101 21:02:02 |
| Zygmunt | 20160101 21:03:59 |
| ADAM | 20160101 21:42:31 |
| ADAM | 20160101 21:51:11 |
| ADAM | 20160101 21:51:24 |

Wykorzystując dane z tabeli rejestracji, należy utworzyć zapytanie generujące listę osób, które wpisały się kilka razy (więcej niż jeden raz) pod rząd. Osoby z wartością NULL w nazwie pomijamy. Należy wyświetlić:

• imię

• liczbę wystąpień w serii

• minimalną datę dodania w serii

• maksymalną datę dodania w serii

Poprawnym wynikiem zapytania powinna być tablica:

|  |  |  |  |
| --- | --- | --- | --- |
| Imię | Liczba wystąpień | Minimalna data | Maksymalna data |
| ADAM | 4 | 2016-01-01 20:41:55.000 | 2016-01-01 20:51:24.000 |
| Zbyszek | 2 | 2016-01-01 20:53:07.000 | 2016-01-01 20:53:44.000 |
| Zygmunt | 4 | 2016-01-01 20:58:31.000 | 2016-01-01 21:03:59.000 |
| ADAM | 3 | 2016-01-01 21:42:31.000 | 2016-01-01 21:51:24.000 |

**ROZWIĄZANIE**

-- Usunięcie tabeli testowej (o ile była wcześniej utworzona)

drop table if exists testTable;

-- Utworzenie tabeli na dane pochodzące z rejestracji

CREATE TABLE testTable (name nvarchar(50), createdDate datetime);

-- Zapełnienie tabeli testTable przykładowymi danymi

INSERT INTO testTable (name, createdDate)

VALUES (NULL,'20160101 20:36:12'),

('ADAM','20160101 20:38:27'),

('Zbyszek','20160101 20:39:02'),

('ADAM','20160101 20:41:55'),

('ADAM','20160101 20:42:31'),

('ADAM','20160101 20:51:11'),

('ADAM','20160101 20:51:24'),

(NULL,'20160101 20:52:01'),

('Zbyszek','20160101 20:53:07'),

('Zbyszek','20160101 20:53:44'),

('Renata','20160101 20:54:52'),

(NULL,'20160101 20:54:57'),

('Zygmunt','20160101 20:54:59'),

(NULL,'20160101 20:55:03'),

(NULL,'20160101 20:56:12'),

('Renata','20160101 20:57:22'),

('Zygmunt','20160101 20:58:31'),

('Zygmunt','20160101 20:59:41'),

('Zygmunt','20160101 21:02:02'),

('Zygmunt','20160101 21:03:59'),

('ADAM','20160101 21:42:31'),

('ADAM','20160101 21:51:11'),

('ADAM','20160101 21:51:24');

Go

-- Deklaracja zmiennej, która będzie przechowywać tabelę wynikową

DECLARE @Result TABLE

(

name NVARCHAR(50),

repeated INT,

minTime DATETIME,

maxTime DATETIME

);

-- Deklaracja innych zmiennych użytych w programie

DECLARE

@name AS NVARCHAR(50),

@prevName AS NVARCHAR(50),

@repeated AS INT,

@createdDate AS DATETIME,

@minTime AS DATETIME,

@prevMinTime AS DATETIME,

@maxTime AS DATETIME,

@prevMaxTime AS DATETIME;

-- Deklaracja i otwarcie kursora

DECLARE C CURSOR /\*FAST\_FORWARD\*/ FOR

SELECT name, createdDate

FROM testTable

where name is not null

ORDER BY createdDate;

OPEN C;

-- Zapis danych z kursora do zmiennych

FETCH NEXT FROM C INTO @name, @createdDate;

-- Ustawienie wartości początkowych

SET @repeated = 1;

SET @minTime = @createdDate;

SET @maxTime = @createdDate;

-- Pętla – pobieranie danych z tabeli wejściowej,

-- analizowanie ich i zapis do tabeli wynikowej

WHILE @@FETCH\_STATUS = 0

BEGIN

SET @prevName = @name;

SET @prevMinTime = @minTime;

SET @prevMaxTime = @maxTime;

FETCH NEXT FROM C INTO @name, @createdDate;

 IF @name <> @prevName

 BEGIN

IF @repeated > 1 BEGIN INSERT INTO @Result VALUES(@prevName, @repeated, @prevMinTime, @prevMaxTime) END;

 SET @repeated = 1;

 SET @minTime = @createdDate;

 SET @maxTime = @createdDate;

 END

 ELSE

 BEGIN

 SET @repeated = @repeated + 1;

 SET @maxTime = @createdDate;

 END

END;

INSERT INTO @Result VALUES(@prevName, @repeated-1, @prevMinTime, @prevMaxTime);

-- Operacje zamykające

CLOSE C;

DEALLOCATE C;

-- Wyświetlenie wyniku

select \* from @Result;