

2. Schreibe eine gespeicherte Prozedur, welche Route, Bahnhof, Ankunfszeit und Abfahrtszeit als Parameter kriegt. Falls sich der Bahnhof schon auf dieser Route befindet, dann werden die Ankunfszeit und Abfahrtszeit entsprechend geändert. Ansonsten, wird der neue Bahnhof zu der Route eingefügt.

Gebe ein Beispiel wo die Prozedur ausgeführt wird.

CREATE PROCEDURE changeTimes(@routeId int, @bahnhofId int, @departure time, @arrival time)

AS

BEGIN

IF EXISTS(SELECT bl.BahnhofeListeId

FROM BahnhofeListe bl

WHERE bl.BahnhofId = @bahnhofId AND bl.RouteId = @routeId)

BEGIN

UPDATE BahnhofeListe

SET Abfahrtszeit = @departure, Ankunftszeit = @arrival

WHERE BahnhofId = @bahnhofId AND RouteId = @routeId;

END

ELSE

BEGIN

INSERT INTO BahnhofeListe (RouteId, BahnhofId, Ankunftszeit, Abfahrtszeit) VALUES (@routeId, @bahnhofId, @departure, @arrival);

END

END

exec changeTimes 1, 1, '10:00', '12:00'

3. Schreibe eine benutzerdefinierte Funktion, die alle Bahnhöfe auflistet, die mehr als ein Zug haben an einem bestimmten Zeitpunkt (gegeben als Parameter). (z.B. um 10:30)

Gebe ein Beispiel wo die Funktion ausgeführt wird.

create function getBahnhofe(@zeitPunkt time)

returns @bahnhofe2 table (name nvarchar(50)) as

begin

insert into @bahnhofe2

select B.Name

from Bahnhof B

where B.IdBahnhof in (

select RP.IdBahnhof

from RoutenPlan RP

where RP.Ankunftszeit <= @zeitPunkt

and RP.Abfahrtszeit >= @zeitPunkt

group by RP.IdBahnhof

having count(\*) >= 2

)

return

End

4. Erstelle einen View/Sicht mit den Namen der Routen, welche die wenigsten Bahnhöfe enthalten und, die, zusätztlich, nicht mehr als 5 Bahnhöfe enthalten.

**#1 Methode**

CREATE VIEW RouteView AS

SELECT R.RouteID, R.Name

FROM Routen R

INNER JOIN RoutenBahnhofen RB on R.RouteID=RB.RouteID

GROUP BY R.RouteID

HAVING COUNT(RB.BahnhofID) <= 5

INTERSECT

SELECT R.RouteId, R.Name

FROM Routen R

INNER JOIN RoutenBahnhofen RB on R.RouteID=RB.RouteID

GROUP BY R.RouteID

HAVING count(RB.BahnhofID) <= ALL (

SELECTCount(RB2.BahnhofID)

FROM Routen R2

INNER JOIN RoutenBahnhofen RB2 on R2.RouteID=RB2.RouteID

GROUP BY R2.RouteID

)

Having nrGari1 <= ALL ( nrGari0, nrGari1, nrGari2,….,nrGariN)

Count(..) < celalte count(\*)

SELECT \* FROM RouteView

**#2 Methode**

CREATE VIEW RouteView AS

SELECT TOP 1 WITH TIES R.RouteID, R.Name

FROM Routen R

INNER JOIN RoutenBahnhofen RB on R.RouteID=RB.RouteID

GROUP BY R.RouteID

HAVING COUNT(RB.BahnhofID) <= 5

ORDER BY COUNT(RB.BahnhofID) ASC