

Database Programming

Exercise 1

Write a SP add_musical_style to add a new musical style, given the stylename of the new musical style (= parameter).

First check if the stylename doesn't already exist. If so throw an exception.

Use an output parameter to pass the result back to the user: 1 if the insert succeeded and 0 if the insert didn't succeed

Use the following testcode

```
-- Testcode 1: add Dance music => no problem
begin try
    begin transaction
        DECLARE @result tinyint
        EXEC add_musical_style 'Dance music', @result
        print 'Dance music inserted or not: ' + str(@result)
        select * from Musical_Styles
        rollback;
    end try
begin catch
    DECLARE @e int;
    SET @e = ERROR_NUMBER();
    PRINT N'Error Procedure = ' + ERROR_PROCEDURE();
    PRINT N'Error Message = ' + ERROR_MESSAGE();

end catch

-- Testcode 2: add Classical => Error because Classical already exists

begin try
    begin transaction
        DECLARE @result tinyint
        EXEC add_musical_style 'Classical', @result
        print 'Classical inserted or not: ' + str(@result)
        select * from Musical_Styles
        rollback
    end try
begin catch
    DECLARE @e int;
    SET @e = ERROR_NUMBER();
    PRINT N'Error Procedure = ' + ERROR_PROCEDURE();
    PRINT N'Error Message = ' + ERROR_MESSAGE();

end catch
```

Exercise 2

Write a SP delete_musical_style to delete a new musical style, given the stylename of the new musical style (= parameter).

First check if the stylename exists. If not so throw an error.

Then check if there are Entertainers with this musical style. If so, throw an error

Then check if there are Customers with this musical style. If so, throw an error

Delete the musical style

Use an output parameter to pass the result back to the user: 1 if the delete succeeded and 0 if the delete didn't succeed

Write testcode to try to delete the following styles: Chamber Music, Elvis, Variety

Exercise 3

Write a SP add_musical_preference to add a new musical preference given a customerid, a stylename and a preferenceSeq (= parameters)

Check if the customer exists. If not, throw an error

Check if the stylename exists. If not, throw an error

Check the preferenceSeq: if the customer already has a stylename with this preferenceSeq: this stylename and the subsequent styles get preferenceSeq + 1

Use an output parameter to pass the result back to the user: 1 if the insert succeeded and 0 if the insert didn't succeed

Write testcode for the following cases:

- for customer 10005 add Variety with PreferenceSeq = 3
- for customer 10007 add Variety with PreferenceSeq = 1

Exercise 4

Write a SP add_entertainer_style to add a new Entertainer_Style, given the entertainerid, the stylename and the stylestrength (= parameters)

Check if the given entertainer exists. If not, throw an error

Check if the given stylename exists. If not, throw an error

Check if the entertainer not already has this style. If so, throw an error

Check if the given stylestrength is a value between 1 and 3

If the given stylestrength is already taken for this entertainerid: the other stylestrengths are getting a new value = old value + 1. Make sure there are no more than 3 stylestrengths per entertainer

Use an output parameter to pass the result back to the user: 1 if the insert succeeded and 0 if the insert didn't succeed

Write testcode for the following cases:

- Add entertainer with entertainerid = 1020 + stylename = chamber music + stylestrength = 1
- Add entertainer with entertainerid = 1003 + stylename = disco + stylestrength = 3
- Add entertainer with entertainerid = 1005 + stylename = Jazz + stylestrength = 1
- Add entertainer with entertainerid = 1005 + stylename = Jazz + stylestrength = 4
- Add entertainer with entertainerid = 1005 + stylename = Folk + stylestrength = 2
- Add entertainer with entertainerid = 1003 + stylename = Folk + stylestrength = 3

Exercise 5

List all customers per city, sorted in alphabetical order.

The image below shows only a part of the result.



Messages

- Auburn
 - Elizabeth Hallmark
- Bellevue
 - Liz Smith
 - Sarah Thompson
 - Joyce Smith
 - Louise Johnson
 - Mark Davison
- Kirkland
 - Zachary Johnson
 - Darren Davidson
 - Ben Clothier
- Redmond
 - Tom Wickerath
 - Kerry Patterson

Exercise 6

List all artists per Entertainer_Styles, sorted from cheaper to more expensive.

If there is an Entertainer_Style for which there are no artists, it should not appear in the overview

Also calculate the average price at the end for each Entertainer_Style

The image below shows only a part of the result.

Messages

60's Music

- JV & the Deep Six 275 EUR
- Country Feeling 280 EUR

--> Average price = 278 EUR

70's Music

- Saturday Revue 250 EUR

--> Average price = 250 EUR

Chamber Music

- Julia Schnebly 90 EUR
- Katherine Ehrlich 145 EUR

--> Average price = 118 EUR

Classic Rock & Roll

- JV & the Deep Six 275 EUR

--> Average price = 275 EUR

Classical

- Susan McLain 75 EUR
- Julia Schnebly 90 EUR
- Katherine Ehrlich 145 EUR

Exercise 7

Give per Member the number of Entertainers a Member is part of and the list of the Entertainers of which a Member is a part of

Notice the difference between e.g. Robert Brown is part of 2 Entertainers and Steve Davidson is part of 1 Entertainer

The image below shows only a part of the result.

Messages	
Robert Brown is part of	2 Entertainers
- Coldwater Cattle Company	
- Country Feeling	
Janice Davidson is part of	2 Entertainers
- Carol Peacock Trio	
- Modern Dance	
Jeffrey Davidson is part of	2 Entertainers
- JV & the Deep Six	
- Modern Dance	
Steve Davidson is part of	1 Entertainer
- Modern Dance	
Susan Davidson is part of	2 Entertainers
- Saturday Revue	
- Susan McLain	
Gary Hallmark is part of	2 Entertainers
- Country Feeling	
- JV & the Deep Six	
Michael Hernandez is part of	3 Entertainers
- Jazz Persuasion	
- Modern Dance	
- Topazz	
George Johnson is part of	2 Entertainers
- Caroline Coie Quartet	
- Country Feeling	
Julia Johnson is part of	1 Entertainer
- Julia Schnebly	
Megan Johnson is part of	1 Entertainer
- Coldwater Cattle Company	
Angel Kennedy is part of	1 Entertainer
- Jazz Persuasion	

Exercise 8

List the Engagements per Agent and what the Agent has earned per year

To add the the small sentence 'Total salary for 2015 is 35510' (for example), you have to add an IF statement after `FETCH NEXT FROM cursor_2 INTO ...`

The image below shows only a part of the result.

Messages									
1 William Thompson Yearly salary = 35000									
- year =	2015								
* EngagementNumber =	13	StartDate = 17 Sep 2015	ContractPrice =	770	CustomerID =	10003	EntertainerID =	1006	Commission = EUR 31
* EngagementNumber =	14	StartDate = 24 Sep 2015	ContractPrice =	2750	CustomerID =	10001	EntertainerID =	1008	Commission = EUR 110
* EngagementNumber =	15	StartDate = 24 Sep 2015	ContractPrice =	770	CustomerID =	10007	EntertainerID =	1013	Commission = EUR 31
* EngagementNumber =	21	StartDate = 30 Sep 2015	ContractPrice =	1490	CustomerID =	10005	EntertainerID =	1003	Commission = EUR 60
* EngagementNumber =	42	StartDate = 30 Oct 2015	ContractPrice =	2150	CustomerID =	10002	EntertainerID =	1013	Commission = EUR 86
* EngagementNumber =	45	StartDate = 21 Oct 2015	ContractPrice =	530	CustomerID =	10015	EntertainerID =	1012	Commission = EUR 21
* EngagementNumber =	48	StartDate = 05 Nov 2015	ContractPrice =	950	CustomerID =	10002	EntertainerID =	1007	Commission = EUR 38
* EngagementNumber =	71	StartDate = 22 Dec 2015	ContractPrice =	1670	CustomerID =	10002	EntertainerID =	1003	Commission = EUR 67
* EngagementNumber =	68	StartDate = 24 Dec 2015	ContractPrice =	1670	CustomerID =	10009	EntertainerID =	1005	Commission = EUR 67
Total salary for	2015 is	35510							
- year =	2016								
* EngagementNumber =	74	StartDate = 01 Jan 2016	ContractPrice =	590	CustomerID =	10004	EntertainerID =	1005	Commission = EUR 24
* EngagementNumber =	111	StartDate = 13 Feb 2016	ContractPrice =	185	CustomerID =	10012	EntertainerID =	1004	Commission = EUR 7
* EngagementNumber =	118	StartDate = 19 Feb 2016	ContractPrice =	350	CustomerID =	10014	EntertainerID =	1010	Commission = EUR 14
* EngagementNumber =	114	StartDate = 19 Feb 2016	ContractPrice =	1550	CustomerID =	10005	EntertainerID =	1002	Commission = EUR 62
* EngagementNumber =	124	StartDate = 23 Feb 2016	ContractPrice =	1850	CustomerID =	10006	EntertainerID =	1008	Commission = EUR 74
* EngagementNumber =	123	StartDate = 23 Feb 2016	ContractPrice =	770	CustomerID =	10013	EntertainerID =	1001	Commission = EUR 31
* EngagementNumber =	131	StartDate = 03 Mar 2016	ContractPrice =	1850	CustomerID =	10014	EntertainerID =	1003	Commission = EUR 74
Total salary for	2016 is	35266							
2 Scott Johnson Yearly salary = 27000									
- year =	2015								
* EngagementNumber =	9	StartDate = 18 Sep 2015	ContractPrice =	1370	CustomerID =	10010	EntertainerID =	1010	Commission = EUR 55
* EngagementNumber =	56	StartDate = 01 Dec 2015	ContractPrice =	770	CustomerID =	10001	EntertainerID =	1002	Commission = EUR 31
* EngagementNumber =	42	StartDate = 09 Dec 2015	ContractPrice =	500	CustomerID =	10003	EntertainerID =	1005	Commission = EUR 20
Total salary for	2015 is	27106							
- year =	2016								
* EngagementNumber =	83	StartDate = 06 Jan 2016	ContractPrice =	650	CustomerID =	10010	EntertainerID =	1006	Commission = EUR 26
* EngagementNumber =	96	StartDate = 30 Jan 2016	ContractPrice =	2930	CustomerID =	10012	EntertainerID =	1010	Commission = EUR 117
* EngagementNumber =	119	StartDate = 19 Feb 2016	ContractPrice =	500	CustomerID =	10012	EntertainerID =	1004	Commission = EUR 20
Total salary for	2016 is	27163							

Exercise 9

If there is an update of the Musical_Preferences table where the PreferenceSeq is adjusted, it must be checked whether the new PreferenceSeq is an allowed value (between 1 and 3)

If not, the transaction must be rolled back and an error should be thrown

In the testcode below, CustomerID = 10001 and StyleID = 10 get's PreferenceSeq = 10

```
begin try
    begin transaction

        UPDATE Musical_Preferences
        SET PreferenceSeq = 10
        WHERE CustomerID = 10001 and StyleID = 10

        print 'Update PreferenceSeq = 10 for CustomerID = 10001 and styleID = 10'

        select * from Musical_Preferences
        WHERE CustomerID = 10001


        rollback;
end try
begin catch
    DECLARE @e int;
    SET @e = ERROR_NUMBER();
    PRINT N'Error Procedure = ' + ERROR_PROCEDURE();
    PRINT N'Error Message = ' + ERROR_MESSAGE();
end catch
```

Exercise 10


If a record is removed from the Musical_Preferences table for a specific customer (for example CustomerID = 10007 + StyleID = 4 + PreferenceSeq = 2), then the PreferenceSeq (for example with value 3) of the subsequent records should be decreased by 1

Write testcode for the following examples


CustomerID	StyleID	PreferenceSeq	CustomerID	StyleID	PreferenceSeq
10007	4	2	10007	8	1
10007	8	1	10007	19	2
10007	19	3			



CustomerID	StyleID	PreferenceSeq	CustomerID	StyleID	PreferenceSeq
10007	4	2	10007	4	1
10007	8	1	10007	19	2
10007	19	3			



CustomerID	StyleID	PreferenceSeq	CustomerID	StyleID	PreferenceSeq
10007	4	2	10007	4	2
10007	8	1	10007	19	1
10007	19	3			




Exercise 11

If a record is inserted into the Musical_Preferences table for a specific customer (for example CustomerID = 10007 + StyleID = 5 + PreferenceSeq = 2), then the PreferenceSeq (for example with value 2 and 3) of the subsequent records should be increased by 1. All records with PreferenceSeq > 3 should be removed.


Is it useful to check whether the new combination of CustomerID and StyleID already exists in the table?

Write testcode for the following examples


CustomerID	StyleID	PreferenceSeq	CustomerID	StyleID	PreferenceSeq
10007	4	2	10007	4	3
10007	8	1	10007	8	1
10007	19	3	10007	19	3
10007	5	2	10007	5	2



CustomerID	StyleID	PreferenceSeq	CustomerID	StyleID	PreferenceSeq
10007	4	2	10007	4	2
10007	8	1	10007	8	1



10007	19	3	10007	19	3
10007	5	3	10007	5	3

CustomerID	StyleID	PreferenceSeq		CustomerID	StyleID	PreferenceSeq
10007	4	2		10007	4	3
10007	8	1		10007	8	2
10007	19	3		10007	19	3
10007	5	1		10007	5	1

Exercise 12

When a Member is added to an Entertainer, then it must be ensured that the EntPricePerDay is adjusted accordingly so that the price per member remains the same.

E.g.: Entertainer 1002 now has 2 members and the EntPricePerDay is 120 EUR. If an extra member is added, the EntPricePerDay should become 180 EUR.