

JOBSHEET

Basis Data Lanjut



Agna Putra Prawira (2341720065)

Class 2I / 01

STUDY PROGRAM D-IV TEKNIK INFORMATIKA
INFORMATION TECHNOLOGY MAJOR
POLITEKNIK NEGERI MALANG

Jl. Soekarno Hatta No. 9 Malang 65144

1.

```
-- Check the data
SELECT empid, firstname, mgrid FROM HR.Employees;
GO

-- How can we change the mgrid of certain employee?? Use UPDATE
UPDATE HR.Employees SET mgrid = 7 WHERE empid = 8;
GO

ALTER PROCEDURE HR.ChangeMgr (@EmployeeId AS INT, @NewMgrID AS INT)
BEGIN
    -- Make sure that the @EmployeeID exists
    DECLARE @EmpName AS VARCHAR(50) = (SELECT firstname FROM HR.Employees WHERE empid = @EmployeeId)
    -- Also, make sure that the @NewMgrID exists
    DECLARE @MgrName AS VARCHAR(50) = (SELECT firstname FROM HR.Employees WHERE empid = @NewMgrID)
    IF @EmpName IS NULL OR @MgrName IS NULL
    BEGIN
        RAISERROR('Employee or Manager does not exist', 16, 1)
        RETURN
    END
    UPDATE HR.Employees SET mgrid = @NewMgrID WHERE empid = @EmployeeId
END
```

95 %

Results Messages

	empid	firstname	mgrid
1	1	Sara	NULL
2	2	Don	1
3	3	Judy	2
4	4	Yael	3
5	5	Sven	2
6	6	Paul	5
7	7	Russell	5
8	8	Maria	3
9	9	Zoya	5

3.

-- BRANCHING

-- Is a facility that enables us to execute or not execute any code based on certain conditions

-- In T-SQL we use the IF..ELSE keyword.

-- Example: Display whether a product is an EXPENSIVE or CHEAP one based on comparison with AVG value of all price

-- We are using the Production.Products table

	productid	productname	supplierid	categoryid	unitprice	discontinued
1	1	Product HHYDP	1	1	18.00	0
2	2	Product RECZE	1	1	19.00	0
3	3	Product IMEHJ	1	2	10.00	0
4	4	Product KSBRM	2	2	22.00	0
5	5	Product EPEIM	2	2	21.35	1
6	6	Product VALIV	3	2	25.00	0
7	7	Product HMLNI	3	7	30.00	0
8	8	Product WVJFP	3	2	40.00	0
9	9	Product AOZBW	4	6	97.00	1
10	10	Product YHXGE	4	8	31.00	0

	Price	Average Price
1	30	28.8663

5.
 - ROUTINE
 - Is element of control of flow that enables us to execute a particular block that processes inputs to become
 - certain result or goal by calling its name.
 - ROUTINE - FUNCTION
 - Is a routine that always have a return value
 - Example: A function to determine that a particular product id is expensive or CHEAP just like the previous case.

Messages

```
Msg 2714, Level 16, State 3, Procedure fn_check_price, Line 1 [Batch Start Line 62]
There is already an object named 'fn_check_price' in the database.

Completion time: 2024-12-10T23:32:43.9027875+07:00
```

6.


```
-- Call the function:
SELECT Production.fn_check_price(7);
-- You can also call it like this:
SELECT
    productid,
    productname,
    unitprice,
    Production.fn_check_price(productid)
FROM
```

95 %

Results **Messages**

	(No column name)
1	EXPENSIVE

```
SELECT
    productid,
    productname,
    unitprice,
    Production.fn_check_price(productid)
FROM
    Production.Products;
```

-- ROUTINE - PROCEDURE
 -- Procedure is a routine that doesnt have any return value.
 -- In SQL Server it is known as STORED PROCEDURE.
 -- Although stored procedure cannot return value, it can however
 -- Example: Create a stored procedure to check whether the value
 -- tables is empty or not. If it is empty, replace it

95 %

Results **Messages**

	productid	productname	unitprice	(No column name)
1	1	Product HHYDP	18.00	CHEAP
2	2	Product RECZE	19.00	CHEAP
3	3	Product IMEHJ	10.00	CHEAP
4	4	Product KSBRM	22.00	CHEAP
5	5	Product EPEIM	21.35	CHEAP
6	6	Product VAIIV	25.00	CHEAP
7	7	Product HMLNI	30.00	EXPENSIVE
8	8	Product WJFJP	40.00	EXPENSIVE
9	9	Product AOZBW	97.00	EXPENSIVE
10	10	Product YHXGE	31.00	EXPENSIVE
11	11	Product QMVUN	21.00	CHEAP
12	12	Product OSFNS	38.00	EXPENSIVE
13	13	Product POXFU	6.00	CHEAP
14	14	Product PWCJB	23.25	CHEAP
15	15	Product KSZOI	15.50	CHEAP
16	16	Product PAFRH	17.45	CHEAP
17	17	Product BLKX	38.00	EXPENSIVE

- 7.

8. -- ROUTINE - PROCEDURE

-- Procedure is a routine that doesn't have any return value.

-- In SQL Server it is known as STORED PROCEDURE.

-- Although stored procedure cannot return value, it can however change the data in our database.

-- Example: Create a stored procedure to check whether the value of 'shipregion' column in Sales.Orders

-- tables is empty or not. If it is empty, replace it with 'Not Available'

```
SELECT * FROM Sales.Orders;  
GO  
  
CREATE PROCEDURE Sales.sp_check_shipregion (@OrderId AS INT) AS  
BEGIN  
    -- Get the value of the shipregion for the given orderId  
    DECLARE @ShipRegion AS NVARCHAR(50) = (SELECT shipregion FROM Sales.Orders WHERE orderId = @OrderId);  
    SELECT @ShipRegion;  
    -- We check, if the value IS NULL  
    IF @ShipRegion IS NULL  
        -- We change it to 'NOT AVAILABLE'  
        UPDATE Sales.Orders SET shipregion = 'NOT AVAILABLE' WHERE orderId = @OrderId;  
END
```

orderid	custid	empid	orderdate	requireddate	shippeddate	shipperid	freight	shipname	shipaddress	shipcity	shipregion	shippostalcode	shipcountry
10248	85	5	2006-07-04 00:00:00.000	2006-08-01 00:00:00.000	2006-07-16 00:00:00.000	3	32.38	Ship to 85-B	6789 rue de l'Abbaye	Reims	NULL	10345	France
10249	79	6	2006-07-05 00:00:00.000	2006-08-16 00:00:00.000	2006-07-10 00:00:00.000	1	11.61	Ship to 79-C	Luisenstr. 9012	Münster	NULL	10326	Germany
10250	34	4	2006-07-08 00:00:00.000	2006-08-05 00:00:00.000	2006-07-12 00:00:00.000	2	65.83	Destination SQKXA	Rua do Paço, 7890	Rio de Janeiro	RJ	10195	Brazil
10251	84	3	2006-07-08 00:00:00.000	2006-08-05 00:00:00.000	2006-07-15 00:00:00.000	1	41.34	Ship to 84-A	3456, rue du Commerce	Lyon	NULL	10342	France
10252	76	4	2006-07-09 00:00:00.000	2006-08-06 00:00:00.000	2006-07-11 00:00:00.000	2	51.30	Ship to 76-B	Boulevard Tirou, 9012	Charleroi	NULL	10318	Belgium
10253	34	3	2006-07-10 00:00:00.000	2006-07-24 00:00:00.000	2006-07-16 00:00:00.000	2	58.17	Destination JPAIY	Rua do Paço, 8901	Rio de Janeiro	RJ	10196	Brazil
10254	14	5	2006-07-11 00:00:00.000	2006-08-08 00:00:00.000	2006-07-23 00:00:00.000	2	22.98	Destination YUIRD	Hauptstr. 1234	Bern	NULL	10139	Switzerland
10255	68	9	2006-07-12 00:00:00.000	2006-08-09 00:00:00.000	2006-07-15 00:00:00.000	3	148.33	Ship to 68-A	Starenweg 6789	Göteborg	NULL	10294	Switzerland
10256	88	3	2006-07-15 00:00:00.000	2006-08-12 00:00:00.000	2006-07-17 00:00:00.000	2	13.97	Ship to 88-B	Rua do Mercado, 5678	Resende	SP	10354	Brazil
10257	35	4	2006-07-16 00:00:00.000	2006-08-13 00:00:00.000	2006-07-22 00:00:00.000	3	81.91	Destination JYDLM	Carerra1234 con Ave. Carlos Soublette #9-35	San Cristóbal	Táchira	10199	Venezuela
10258	20	1	2006-07-17 00:00:00.000	2006-08-14 00:00:00.000	2006-07-23 00:00:00.000	1	140.51	Destination RYDMF	Kirchgasse 9012	Graz	NULL	10157	Austria
10259	13	4	2006-07-18 00:00:00.000	2006-08-15 00:00:00.000	2006-07-25 00:00:00.000	3	3.25	Destination LGGCH	Sierras de Granada 9012	México D.F.	NULL	10137	Mexico
10260	56	4	2006-07-19 00:00:00.000	2006-08-16 00:00:00.000	2006-07-29 00:00:00.000	1	55.09	Ship to 56-A	Mehlhheimerstr. 0123	Köln	NULL	10258	Germany
10261	61	4	2006-07-19 00:00:00.000	2006-08-16 00:00:00.000	2006-07-30 00:00:00.000	2	3.05	Ship to 61-B	Rua da Panificadora, 6789	Rio de Janeiro	RJ	10274	Brazil
10262	65	8	2006-07-22 00:00:00.000	2006-08-19 00:00:00.000	2006-07-25 00:00:00.000	3	48.29	Ship to 65-B	8901 Milan Dr	Albuquerque	NM	10286	USA
10263	20	9	2006-07-23 00:00:00.000	2006-08-20 00:00:00.000	2006-07-31 00:00:00.000	3	146.06	Destination FFXXT	Kirchgasse 0123	Graz	NULL	10158	Austria
10264	54	6	2006-07-24 00:00:00.000	2006-08-21 00:00:00.000	2006-08-23 00:00:00.000	3	3.67	Destination VFFBN	Kirchgasse 9012	Graz	NULL	10157	Austria

Query executed successfully. LAPTOP-DL9EITU3\MSSQLSERVER... LAPTOP-DL9EITU3\user (65) TSQ2012 00:00:00 830 rows

9. -- Calling procedure

EXECUTE Sales.sp_check_shipregion @OrderId=10248;

SELECT * FROM Sales.Orders;

orderid	custid	empid	orderdate	requireddate	shippeddate	shipperid	freight	shipname	shipaddress	shipcity	shipregion	shippostalcode	shipcountry
10248	85	5	2006-07-04 00:00:00.000	2006-08-01 00:00:00.000	2006-07-16 00:00:00.000	3	32.38	Ship to 85-B	6789 rue de l'Abbaye	Reims	NOT AVAILABLE	10345	France
10249	79	6	2006-07-05 00:00:00.000	2006-08-16 00:00:00.000	2006-07-10 00:00:00.000	1	11.61	Ship to 79-C	Luisenstr. 9012	Münster	NULL	10326	Germany
10250	34	4	2006-07-08 00:00:00.000	2006-08-05 00:00:00.000	2006-07-12 00:00:00.000	2	65.83	Destination SQKXA	Rua do Paço, 7890	Rio de Janeiro	RJ	10195	Brazil
10251	84	3	2006-07-08 00:00:00.000	2006-08-05 00:00:00.000	2006-07-15 00:00:00.000	1	41.34	Ship to 84-A	3456, rue du Commerce	Lyon	NULL	10342	France
10252	76	4	2006-07-09 00:00:00.000	2006-08-06 00:00:00.000	2006-07-11 00:00:00.000	2	51.30	Ship to 76-B	Boulevard Tirou, 9012	Charleroi	NULL	10318	Belgium
10253	34	3	2006-07-10 00:00:00.000	2006-07-24 00:00:00.000	2006-07-16 00:00:00.000	2	58.17	Destination JPAIY	Rua do Paço, 8901	Rio de Janeiro	RJ	10196	Brazil
10254	14	5	2006-07-11 00:00:00.000	2006-08-08 00:00:00.000	2006-07-23 00:00:00.000	2	22.98	Destination YUIRD	Hauptstr. 1234	Bern	NULL	10139	Switzerland
10255	68	9	2006-07-12 00:00:00.000	2006-08-09 00:00:00.000	2006-07-15 00:00:00.000	3	148.33	Ship to 68-A	Starenweg 6789	Göteborg	NULL	10294	Switzerland
10256	88	3	2006-07-15 00:00:00.000	2006-08-12 00:00:00.000	2006-07-17 00:00:00.000	2	13.97	Ship to 88-B	Rua do Mercado, 5678	Resende	SP	10354	Brazil
10257	35	4	2006-07-16 00:00:00.000	2006-08-13 00:00:00.000	2006-07-22 00:00:00.000	3	81.91	Destination JYDLM	Carerra1234 con Ave. ...	San Cristóbal	Táchira	10199	Venezuela
10258	20	1	2006-07-17 00:00:00.000	2006-08-14 00:00:00.000	2006-07-23 00:00:00.000	1	140.51	Destination RYDMF	Kirchgasse 9012	Graz	NULL	10157	Austria

Query executed successfully. LAPTOP-DL9EITU3\MSSQLSERVER... LAPTOP-DL9EITU3\user (65) TSQ2012 00:00:00 831 rows