

8.3 -

studentwebmysql.ewu.edu ► eglushchenko

Structure SQL Search Query Export Import Operations

✓ Your SQL query has been executed successfully (Query took 0.0141 sec)

```
CREATE TABLE REGION(  
  RegionID CHAR( 1 ) NOT NULL ,  
  RegionName VARCHAR( 12 ) NOT NULL ,  
  PRIMARY KEY ( RegionID )  
);
```

Run SQL query/queries on database eglushchenko: ?

```
CREATE TABLE REGION  
(  
  RegionID CHAR(1) NOT NULL,  
  RegionName VARCHAR(12) NOT NULL,  
  PRIMARY KEY (RegionID)  
)
```

✓ Your SQL query has been executed successfully

```
INSERT INTO REGION  
VALUES (  
  'C', 'Chicagoland'  
); # 1 row(s) affected.  
INSERT INTO REGION  
VALUES (  
  'T', 'Tristate'  
); # 1 row(s) affected.
```

Run SQL query/queries on database eglushchenko: ?

```
INSERT INTO REGION VALUES ('C', 'chicagoland');# 1 row(s) affected.  
INSERT INTO REGION VALUES ('T', 'Tristate');# 1 row(s) affected.
```

Ephrem Glushchenko

Assignment 7

5/5/2019

MISC 485

✓ Showing rows 0 - 1 (~2¹ total, Query took 0.0006 sec)

```
SELECT *
FROM `region`
LIMIT 0 , 30
```

in mode and repeat headers after cells

Sort by key:

+ Options

		RegionID	RegionName
<input type="checkbox"/>		C	Chicagoland
<input type="checkbox"/>		T	Tristate

Check All / Uncheck All With selected:

in mode and repeat headers after cells

Query results operations

Print view Print view (with full texts) Export CREATE VIEW

✓ Your SQL query has been executed successfully (Query took 0.0337 sec)

```
CREATE TABLE STORE(
  StoreID CHAR( 2 ) NOT NULL ,
  StoreZip CHAR( 5 ) NOT NULL ,
  RegionID CHAR( 1 ) NULL ,
  PRIMARY KEY ( StoreID ) ,
  FOREIGN KEY ( RegionID ) REFERENCES REGION( RegionID )
);
```

Run SQL query/queries on database **eglushchenko**: ?

```
CREATE TABLE STORE
(
  StoreID CHAR(2) NOT NULL,
  StoreZip CHAR(5) NOT NULL,
  RegionID CHAR(1) NULL,
  PRIMARY KEY(StoreID),
  FOREIGN KEY(RegionID)
  REFERENCES REGION(RegionID)
)
```

Ephrem Glushchenko

Assignment 7

5/5/2019

MISC 485

```
✓ Your SQL query has been executed successfully

INSERT INTO STORE
VALUES (
  'S1', '60600', 'C'
); # 1 row(s) affected.
INSERT INTO STORE
VALUES (
  'S2', '60605', 'C'
); # 1 row(s) affected.
INSERT INTO STORE
VALUES (
  'S3', '35400', 'T'
```

Run SQL query/queries on database **eglushchenko**: ?

```
INSERT INTO STORE VALUES ('S1', '60600', 'C');# 1 row(s) affected.
INSERT INTO STORE VALUES ('S2', '60605', 'C');# 1 row(s) affected.
INSERT INTO STORE VALUES ('S3', '35400', 'T');# 1 row(s) affected.
```

```
✓ Showing rows 0 - 2 (~31 total, Query took 0.0008 sec)

SELECT *
FROM `store`
LIMIT 0 , 30
```

Show : 30 row(s) starting from record # 0
in horizontal mode and repeat headers after 100 cells
Sort by key: None

+ Options

			StoreID	StoreZip	RegionID
<input type="checkbox"/>			S1	60600	C
<input type="checkbox"/>			S2	60605	C
<input type="checkbox"/>			S3	35400	T

Check All / Uncheck All With selected:

Show : 30 row(s) starting from record # 0
in horizontal mode and repeat headers after 100 cells

```
✓ Your SQL query has been executed successfully ( Query took 0.0130 sec )

CREATE TABLE CUSTOMER(
  CustomerID CHAR( 7 ) NOT NULL ,
  CustomerName VARCHAR( 4 ) NOT NULL ,
  CustomerZip VARCHAR( 5 ) ,
  PRIMARY KEY ( CustomerID )
);
```

Run SQL query/queries on database **eglushchenko**: ?

```
CREATE TABLE CUSTOMER
(
  CustomerID CHAR(7) NOT NULL,
  CustomerName VARCHAR(4) NOT NULL,
  CustomerZip VARCHAR(5),
  PRIMARY KEY(CustomerID)
)
```

Ephrem Glushchenko

Assignment 7

5/5/2019

MISC 485

✓ Your SQL query has been executed successfully

```
INSERT INTO CUSTOMER
VALUES (
  '1-2-333', 'Tina', '60137'
); # 1 row(s) affected.
INSERT INTO CUSTOMER
VALUES (
  '2-3-444', 'Tony', '60611'
); # 1 row(s) affected.
INSERT INTO CUSTOMER
VALUES (
  '3-4-555', 'Pam', '35401'
);
```

Run SQL query/queries on database **eglushchenko**: ?

```
INSERT INTO CUSTOMER VALUES ('1-2-333', 'Tina', '60137');# 1 row(s) affected.
INSERT INTO CUSTOMER VALUES ('2-3-444', 'Tony', '60611');# 1 row(s) affected.
INSERT INTO CUSTOMER VALUES ('3-4-555', 'Pam', '35401');# 1 row(s) affected.
```

✓ Showing rows 0 - 2 (~3¹ total, Query took 0.0005 sec)

```
SELECT *
FROM 'customer'
LIMIT 0 , 30
```

Show : 30 row(s) starting from record # 0
in horizontal mode and repeat headers after 100 cells
Sort by key: None

+ Options

	CustomerID	CustomerName	CustomerZip
<input type="checkbox"/>	1-2-333	Tina	60137
<input type="checkbox"/>	2-3-444	Tony	60611
<input type="checkbox"/>	3-4-555	Pam	35401

Check All / Uncheck All With selected:

Show : 30 row(s) starting from record # 0
in horizontal mode and repeat headers after 100 cells

✓ Your SQL query has been executed successfully

```
CREATE TABLE VENDOR(
  VendorID CHAR( 2 ) NOT NULL ,
  VendorName VARCHAR( 15 ) NOT NULL ,
  PRIMARY KEY ( VendorID )
);
```

Run SQL query/queries on database **eglushchenko**

```
CREATE TABLE VENDOR
(
  VendorID CHAR(2) NOT NULL,
  VendorName VARCHAR(15) NOT NULL,
  PRIMARY KEY(VendorID)
)
```

Ephrem Glushchenko

Assignment 7

5/5/2019

MISC 485

```
✓ Your SQL query has been executed successfully

INSERT INTO VENDOR
VALUES (
  'PG', 'Pacifica Gear'
); # 1 row(s) affected.
INSERT INTO VENDOR
VALUES (
  'MK', 'Mountain King'
); # 1 row(s) affected.
```

Run SQL query/queries on database **eglushchenko**: ?

```
INSERT INTO VENDOR VALUES ('PG', 'Pacifica Gear');# 1 row(s) affected.
INSERT INTO VENDOR VALUES ('MK', 'Mountain King');# 1 row(s) affected.
```

```
✓ Showing rows 0 - 1 (~21 total, Query took 0.0009 sec)

SELECT *
FROM `vendor`
LIMIT 0 , 30
```

Show: 30 row(s) starting from record # 0
in horizontal mode and repeat headers after 100 cells
Sort by key: None

+ Options

	VendorID	VendorName
<input type="checkbox"/>	MK	Mountain King
<input type="checkbox"/>	PG	Pacifica Gear

Check All / Uncheck All With selected:

Show: 30 row(s) starting from record # 0
in horizontal mode and repeat headers after 100 cells

```
✓ Your SQL query has been executed successfully

CREATE TABLE CATEGORY(
  CategoryIDCHAR( 2 ) NOT NULL ,
  CategoryName VARCHAR( 10 ) ,
  PRIMARY KEY ( CategoryID )
);
```

Run SQL query/queries on database **eglushchenko**


```
CREATE TABLE CATEGORY
(
  CategoryID CHAR(2) NOT NULL,
  CategoryName VARCHAR(10),
  PRIMARY KEY(CategoryID)
)
```

Ephrem Glushchenko

Assignment 7

5/5/2019


MISC 485

 Your SQL query has been executed successfully

```
INSERT INTO CATEGORY
VALUES (
  'CP', 'Camping'
); # 1 row(s) affected.
INSERT INTO CATEGORY
VALUES (
  'FW', 'Footware'
); # 1 row(s) affected.
```

Run SQL query/queries on database **eglushchenko**: ?

```
INSERT INTO CATEGORY VALUES ('CP', 'Camping');# 1 row(s) affected.
INSERT INTO CATEGORY VALUES ('FW', 'Footware');# 1 row(s) affected.
```

 Showing rows 0 - 1 (~2¹ total, Query took 0.0008 sec)

```
SELECT *
FROM 'category'
LIMIT 0, 30
```





☐ Preview





Show: 30 row(s) starting from record # 0

in horizontal mode and repeat headers after 100 cells

Sort by key: None

+ Options

		CategoryID	CategoryName	
<input type="checkbox"/>			CP	Camping
<input type="checkbox"/>			FW	Footware

 Check All / Uncheck All With selected:   

Show: 30 row(s) starting from record # 0

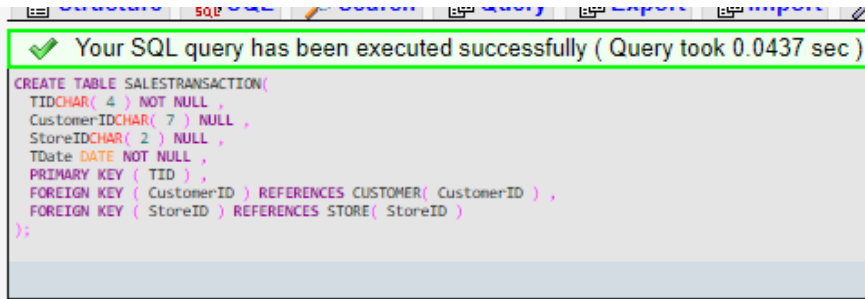
in horizontal mode and repeat headers after 100 cells

Ephrem Glushchenko

Assignment 7

5/5/2019

MISC 485



✓ Your SQL query has been executed successfully (Query took 0.0437 sec)

```
CREATE TABLE SALESTRANSACTION(  
  TIDCHAR( 4 ) NOT NULL  
  CustomerIDCHAR( 7 ) NULL ,  
  StoreIDCHAR( 2 ) NULL ,  
  TDate DATE NOT NULL ,  
  PRIMARY KEY ( TID ) ,  
  FOREIGN KEY ( CustomerID ) REFERENCES CUSTOMER( CustomerID ) ,  
  FOREIGN KEY ( StoreID ) REFERENCES STORE( StoreID )  
);
```

Run SQL query/queries on database **eglushchenko**: ?

```
CREATE TABLE SALESTRANSACTION  
(  
  TID CHAR(4) NOT NULL,  
  CustomerID CHAR(7) NULL,  
  StoreID CHAR(2) NULL,  
  TDate DATE NOT NULL,  
  PRIMARY KEY(TID),  
  FOREIGN KEY(CustomerID) REFERENCES CUSTOMER(CustomerID),  
  FOREIGN KEY(StoreID) REFERENCES STORE(storeID)  
)
```

INSERT INTO SALESTRANSACTION VALUES ('T111', '1-2-333', 'S1', 2013-01-01);

INSERT INTO SALESTRANSACTION VALUES ('T222', '2-3-444', 'S2', 2013-01-01);

INSERT INTO SALESTRANSACTION VALUES ('T333', '1-2-333', 'S3', 2013-01-02);

INSERT INTO SALESTRANSACTION VALUES ('T444', '3-4-555', 'S3', 2013-01-02);

INSERT INTO SALESTRANSACTION VALUES ('T555', '2-3-444', 'S3', 2013-01-02);

Ephrem Glushchenko

Assignment 7

5/5/2019

MISC 485

✓ Showing rows 0 - 4 (~5¹ total, Query took 0.0007 sec)

```
SELECT *
FROM 'salestransaction'
LIMIT 0, 30
```

Show : 30 row(s) starting from record # 0

in horizontal mode and repeat headers after 100 cells

Sort by key: None

+ Options

			TID	CustomerID	StoreID	TDate
<input type="checkbox"/>			T111	1-2-333	S1	2013-01-01
<input type="checkbox"/>			T222	2-3-444	S2	2013-01-01
<input type="checkbox"/>			T333	1-2-333	S3	2013-01-02
<input type="checkbox"/>			T444	3-4-555	S3	2013-01-02
<input type="checkbox"/>			T555	2-3-444	S3	2013-01-02

Check All / Uncheck All With selected:

Show : 30 row(s) starting from record # 0

in horizontal mode and repeat headers after 100 cells

Structure SQL Search Query Export Import

✓ Your SQL query has been executed successfully (Query took 0.0354 sec)

```
CREATE TABLE PRODUCT(
  ProductIDCHAR( 3 ) NOT NULL ,
  ProductName VARCHAR( 10 ) NOT NULL ,
  ProductPrice INT NOT NULL ,
  VendorIDCHAR( 2 ) NULL ,
  CategoryIDCHAR( 2 ) NULL ,
  PRIMARY KEY ( ProductID ) ,
  FOREIGN KEY ( VendorID ) REFERENCES VENDOR( VendorID ) ,
  FOREIGN KEY ( CategoryID ) REFERENCES CATEGORY( CategoryID )
);
```

Run SQL query/queries on database eglushchenko: ?

```
CREATE TABLE PRODUCT
(
  ProductID CHAR(3) NOT NULL,
  ProductName VARCHAR(10) NOT NULL,
  ProductPrice INT NOT NULL,
  VendorID CHAR(2) NULL,
  CategoryID CHAR(2) NULL,
  PRIMARY KEY(ProductID),
  FOREIGN KEY(VendorID) REFERENCES VENDOR(VendorID),
  FOREIGN KEY(CategoryID) REFERENCES CATEGORY(CategoryID)
)
```


Ephrem Glushchenko

Assignment 7

5/5/2019

MISC 485

✓ Your SQL query has been executed successfully

```
INSERT INTO PRODUCT
VALUES (
  '1X1', 'Zzz Bag', '100', 'PG', 'CP'
); # 1 row(s) affected.
INSERT INTO PRODUCT
VALUES (
  '2X2', 'Easy Boot', '70', 'MK', 'FW'
); # 1 row(s) affected.
INSERT INTO PRODUCT
VALUES (
  '3X3', 'Cosy Sock', '15', 'MK', 'FW'
```

Run SQL query/queries on database **eglushchenko**: ⓘ

```
INSERT INTO PRODUCT VALUES ('1X1', 'Zzz Bag', '100', 'PG', 'CP');# 1 row(s) affected.
INSERT INTO PRODUCT VALUES ('2X2', 'Easy Boot', '70', 'MK', 'FW');# 1 row(s) affected.
INSERT INTO PRODUCT VALUES ('3X3', 'Cosy Sock', '15', 'MK', 'FW');# 1 row(s) affected.
INSERT INTO PRODUCT VALUES ('4X4', 'Dura Boot', '90', 'PG', 'FW');# 1 row(s) affected.
INSERT INTO PRODUCT VALUES ('5X5', 'Tiny Tent', '150', 'MK', 'CP');# 1 row(s) affected.
INSERT INTO PRODUCT VALUES ('6X6', 'Biggy Tent', '250', 'MK', 'CP');# 1 row(s) affected.
```

✓ Showing rows 0 - 5 (~6¹ total, Query took 0.0005 sec)

```
SELECT *
FROM 'product'
LIMIT 0, 30
```

☐ Profiling [

Show : 30 row(s) starting from record # 0
in horizontal mode and repeat headers after 100 cells
Sort by key: None

+ Options

			ProductID	ProductName	ProductPrice	VendorID	CategoryID
<input type="checkbox"/>			1X1	Zzz Bag	100	PG	CP
<input type="checkbox"/>			2X2	Easy Boot	70	MK	FW
<input type="checkbox"/>			3X3	Cosy Sock	15	MK	FW
<input type="checkbox"/>			4X4	Dura Boot	90	PG	FW
<input type="checkbox"/>			5X5	Tiny Tent	150	MK	CP
<input type="checkbox"/>			6X6	Biggy Tent	250	MK	CP

⬆ Check All / Uncheck All With selected:

Show : 30 row(s) starting from record # 0
in horizontal mode and repeat headers after 100 cells

```
✓ Your SQL query has been executed successfully

CREATE TABLE SOLDVIA(
  ProductIDCHAR( 3 ) NULL ,
  TIDCHAR( 4 ) NULL ,
  NoOfItemsCHAR( 1 ) NOT NULL ,
  PRIMARY KEY ( ProductID, TID ) ,
  FOREIGN KEY ( ProductID ) REFERENCES PRODUCT( ProductID ) ,
  FOREIGN KEY ( TID ) REFERENCES SALESTRANSACTION( TID )
);
```

Run SQL query/queries on database [eglushchenko](#)

```
CREATE TABLE SOLDVIA
(
  ProductID CHAR(3) NULL,
  TID CHAR(4) NULL,
  NoOfItems CHAR(1) NOT NULL,
  PRIMARY KEY(ProductID, TID),
  FOREIGN KEY(ProductID) REFERENCES PRODUCT(ProductID),
  FOREIGN KEY(TID) REFERENCES SALESTRANSACTION(TID)
)
```

```
✓ Your SQL query has been executed successfully

INSERT INTO SOLDVIA
VALUES (
  '1X1', 'T111', '1'
); # 1 row(s) affected.
INSERT INTO SOLDVIA
VALUES (
  '2X2', 'T222', '1'
); # 1 row(s) affected.
INSERT INTO SOLDVIA
VALUES (
  '3X3', 'T333', '5'
```

Run SQL query/queries on database [eglushchenko](#): ?

```
INSERT INTO SOLDVIA VALUES ('1X1', 'T111', '1');# 1 row(s) affected.
INSERT INTO SOLDVIA VALUES ('2X2', 'T222', '1');# 1 row(s) affected.
INSERT INTO SOLDVIA VALUES ('3X3', 'T333', '5');# 1 row(s) affected.
INSERT INTO SOLDVIA VALUES ('1X1', 'T333', '1');# 1 row(s) affected.
INSERT INTO SOLDVIA VALUES ('4X4', 'T444', '1');# 1 row(s) affected.
INSERT INTO SOLDVIA VALUES ('2X2', 'T444', '2');# 1 row(s) affected.
INSERT INTO SOLDVIA VALUES ('4X4', 'T555', '4');# 1 row(s) affected.
INSERT INTO SOLDVIA VALUES ('5X5', 'T555', '2');# 1 row(s) affected.
```

Ephrem Glushchenko

Assignment 7

5/5/2019

MISC 485

✓ Showing rows 0 - 8 (~9¹ total, Query took 0.0009 sec)

```
SELECT *  
FROM `soldvia`  
LIMIT 0 , 30
```

Show : 30 row(s) starting from record # 0

in horizontal mode and repeat headers after 100 cells

Sort by key: None

+ Options

			ProductID	TID	NoOfItems
<input type="checkbox"/>			1X1	T111	1
<input type="checkbox"/>			1X1	T333	1
<input type="checkbox"/>			2X2	T222	1
<input type="checkbox"/>			2X2	T444	2
<input type="checkbox"/>			3X3	T333	5
<input type="checkbox"/>			4X4	T444	1
<input type="checkbox"/>			4X4	T555	4
<input type="checkbox"/>			5X5	T555	2
<input type="checkbox"/>			6X6	T555	1

↑

Check All / Uncheck All With selected:

Show : 30 row(s) starting from record # 0

in horizontal mode and repeat headers after 100 cells

8.4 –

Ephrem Glushchenko

Assignment 7

5/5/2019

MISC 485

✓ Your SQL query has been executed successfully (Query took 0.0134 sec)

```
CREATE TABLE CALENDAR(  
  CalendarKey VARCHAR( 10 ) NOT NULL ,  
  FullDate DATE NOT NULL ,  
  DayOfWeek CHAR( 2 ) NOT NULL ,  
  DayOfMonth CHAR( 2 ) NOT NULL ,  
  MONTH VARCHAR( 15 ) NOT NULL ,  
  Quarter VARCHAR( 10 ) NOT NULL ,  
  YEAR CHAR( 4 ) ,  
  PRIMARY KEY ( CalendarKey )  
);
```

Run SQL query/queries on database [eglushchenko](#): ?

```
CREATE TABLE CALENDAR  
(  
  CalendarKey VARCHAR(10) NOT NULL,  
  FullDate DATE NOT NULL,  
  DayOfWeek CHAR(2) NOT NULL,  
  DayOfMonth CHAR(2) NOT NULL,  
  Month VARCHAR(15) NOT NULL,  
  Quarter VARCHAR(10) NOT NULL,  
  Year CHAR(4),  
  PRIMARY KEY(CalendarKey)  
)
```

✓ Your SQL query has been executed suc

```
CREATE TABLE STORE(  
  StoreKey VARCHAR( 10 ) NOT NULL ,  
  StoreID VARCHAR( 10 ) NOT NULL ,  
  StoreZip VARCHAR( 10 ) NOT NULL ,  
  StoreRegionName VARCHAR( 20 ) NOT NULL ,  
  PRIMARY KEY ( StoreKey )  
);
```

Run SQL query/queries on database [eglu](#)

```
CREATE TABLE STORE  
(  
  StoreKey VARCHAR(10) NOT NULL,  
  StoreID VARCHAR(10) NOT NULL,  
  StoreZip VARCHAR(10) NOT NULL,  
  StoreRegionName VARCHAR(20) NOT NULL,  
  PRIMARY KEY(StoreKey)  
)
```

Ephrem Glushchenko

Assignment 7

5/5/2019

MISC 485

✓ Your SQL query has been executed successfully (C

```
CREATE TABLE PRODUCT(  
  ProductKey CHAR( 7 ) NOT NULL ,  
  ProductID VARCHAR( 10 ) NOT NULL ,  
  ProductName VARCHAR( 15 ) NOT NULL ,  
  ProductPrice INT NOT NULL ,  
  ProductVendorName VARCHAR( 10 ) NOT NULL ,  
  ProductCategoryName VARCHAR( 10 ) NOT NULL ,  
  PRIMARY KEY ( ProductKey )  
);
```

Run SQL query/queries on database [eglushchenko](#):

```
CREATE TABLE PRODUCT  
(  
  ProductKey CHAR(7) NOT NULL,  
  ProductID VARCHAR(10) NOT NULL,  
  ProductName VARCHAR(15) NOT NULL,  
  ProductPrice INT NOT NULL,  
  ProductVendorName VARCHAR(10) NOT NULL,  
  ProductCategoryName VARCHAR(10) NOT NULL,  
  PRIMARY KEY(ProductKey)  
)
```

✓ Your SQL query has been execut

```
CREATE TABLE CUSTOMER(  
  CustomerKey CHAR( 7 ) NOT NULL ,  
  CustomerID VARCHAR( 10 ) NOT NULL ,  
  CustomerName VARCHAR( 15 ) NOT NULL ,  
  CustomerZip CHAR( 5 ) ,  
  PRIMARY KEY ( CustomerKey )  
);
```

Run SQL query/queries on databas

```
CREATE TABLE CUSTOMER  
(  
  CustomerKey CHAR(7) NOT NULL,  
  CustomerID VARCHAR(10) NOT NULL,  
  CustomerName VARCHAR(15) NOT NULL,  
  CustomerZip CHAR(5),  
  PRIMARY KEY(CustomerKey)  
)
```

Ephrem Glushchenko

Assignment 7

5/5/2019

MISC 485

✓ Your SQL query has been executed successfully (Query took 0.0

```
CREATE TABLE SALES_Fact_table(  
  CalendarKey CHAR( 7 ) NULL ,  
  StoreKey CHAR( 5 ) NULL ,  
  ProductKey CHAR( 5 ) NULL ,  
  CustomerKey CHAR( 5 ) NULL ,  
  DollarsSold INT NOT NULL ,  
  UnitsSold INT NOT NULL ,  
  FOREIGN KEY ( CalendarKey ) REFERENCES CALENDAR( CalendarKey ) ,  
  FOREIGN KEY ( StoreKey ) REFERENCES STORE( StoreKey ) ,  
  FOREIGN KEY ( ProductKey ) REFERENCES PRODUCT( ProductKey ) ,  
  FOREIGN KEY ( CustomerKey ) REFERENCES CUSTOMER( CustomerKey )
```

Run SQL query/queries on database eglushchenko: ⓘ

```
CREATE TABLE SALES_Fact_table  
(  
  CalendarKey CHAR(7) NULL,  
  StoreKey CHAR(5) NULL,  
  ProductKey CHAR(5) NULL,  
  CustomerKey CHAR(5) NULL,  
  DollarsSold INT NOT NULL,  
  UnitsSold INT NOT NULL,  
  FOREIGN KEY(CalendarKey) REFERENCES CALENDAR(CalendarKey),  
  FOREIGN KEY(StoreKey) REFERENCES STORE(StoreKey),  
  FOREIGN KEY(ProductKey) REFERENCES PRODUCT(ProductKey),  
  FOREIGN KEY(CustomerKey) REFERENCES CUSTOMER(CustomerKey)  
)
```