CREATE RULE Val\_Gender

AS @col IN ('М', 'Ж')

go

CREATE RULE Val\_Size

AS @col IN ('38', '40-42', '44-46', '48-50', '52-54', '56-58', '60-62')

go

CREATE RULE Val\_ShoeSize

AS @col IN (48, 50, 51, 52, 38, 39, 40, 41, 42, 43, 44, 45, 46, 47)

go

CREATE RULE Val\_Unit

AS @col IN ('шт.', 'пара')

go

CREATE TABLE CANCELLATION

(

ID\_Cancellation int IDENTITY ( 1,1 ) ,

ID\_Issued integer NOT NULL ,

Date\_Cancellation datetime NOT NULL

)

go

ALTER TABLE CANCELLATION

ADD CONSTRAINT XPKCANCELLATION PRIMARY KEY CLUSTERED (ID\_Cancellation ASC)

go

CREATE TABLE EMPL\_POSITION

(

Name\_Empl\_Position varchar(60) NOT NULL ,

Id\_Position int IDENTITY ( 1,1 ) ,

ID\_Manufactory int NOT NULL

)

go

ALTER TABLE EMPL\_POSITION

ADD CONSTRAINT XPKEMPL\_POSITION PRIMARY KEY CLUSTERED (Id\_Position ASC)

go

CREATE TABLE ISSUED

(

ID\_Issued int IDENTITY ( 1,1 ) ,

ID\_Revenue integer NOT NULL ,

ID\_person integer NOT NULL ,

Cancellation bit NOT NULL ,

Date\_Issued datetime NOT NULL

)

go

ALTER TABLE ISSUED

ADD CONSTRAINT XPKISSUED PRIMARY KEY CLUSTERED (ID\_Issued ASC)

go

ALTER TABLE ISSUED

ADD CONSTRAINT XAK1ISSUED UNIQUE (ID\_Revenue ASC)

go

CREATE TABLE MANUFACTORY

(

Name\_manufactory varchar(60) NOT NULL ,

ID\_Manufactory int IDENTITY ( 1,1 )

)

go

ALTER TABLE MANUFACTORY

ADD CONSTRAINT XPKMANUFACTORY PRIMARY KEY CLUSTERED (ID\_Manufactory ASC)

go

ALTER TABLE MANUFACTORY

ADD CONSTRAINT XAK1MANUFACTORY UNIQUE (Name\_manufactory ASC)

go

CREATE TABLE NORM

(

ID\_Norm int IDENTITY ( 1,1 ) ,

Amount integer NULL ,

ID\_Workwear int NOT NULL ,

Id\_Position int NOT NULL

)

go

ALTER TABLE NORM

ADD CONSTRAINT XPKNORM PRIMARY KEY CLUSTERED (ID\_Norm ASC)

go

CREATE TABLE PERSON

(

ID\_person int IDENTITY ( 1,1 ) ,

Surname varchar(60) NOT NULL ,

Name varchar(20) NOT NULL ,

Patronymic varchar(30) NOT NULL ,

Sex varchar(20) NOT NULL ,

Height integer NOT NULL ,

Shoe\_size char(18) NOT NULL ,

Size\_headdress integer NOT NULL ,

Size\_glove char(18) NOT NULL ,

Clothing\_size varchar(20) NOT NULL ,

Id\_Position int NOT NULL

)

go

ALTER TABLE PERSON

ADD CONSTRAINT XPKPERSON PRIMARY KEY CLUSTERED (ID\_person ASC)

go

CREATE TABLE REVENUE

(

ID\_Revenue int IDENTITY ( 1,1 ) ,

Issued bit NOT NULL ,

Date\_revenue datetime NOT NULL ,

Clothing\_size varchar(20) NULL ,

Shoe\_size integer NULL ,

Size\_headdress integer NULL ,

Size\_glove integer NULL ,

ID\_Workwear int NOT NULL

)

go

ALTER TABLE REVENUE

ADD CONSTRAINT XPKREVENUE PRIMARY KEY CLUSTERED (ID\_Revenue ASC)

go

CREATE TABLE WORKWEAR\_DIRECTORY

(

Name\_Workwear varchar(40) NOT NULL ,

Category varchar(20) NOT NULL ,

Description varchar(200) NULL ,

Time\_of\_Wear integer NOT NULL ,

Unit varchar(20) NOT NULL ,

ID\_Workwear int IDENTITY ( 1,1 )

)

go

ALTER TABLE WORKWEAR\_DIRECTORY

ADD CONSTRAINT XPKWORKWEAR\_DIRECTORY PRIMARY KEY CLUSTERED (ID\_Workwear ASC)

go

ALTER TABLE CANCELLATION

ADD CONSTRAINT R\_12 FOREIGN KEY (ID\_Issued) REFERENCES ISSUED(ID\_Issued)

go

ALTER TABLE EMPL\_POSITION

ADD CONSTRAINT R\_15 FOREIGN KEY (ID\_Manufactory) REFERENCES MANUFACTORY(ID\_Manufactory)

go

ALTER TABLE ISSUED

ADD CONSTRAINT Default\_Value\_451\_525612567

DEFAULT 0 FOR Cancellation

go

ALTER TABLE ISSUED

ADD CONSTRAINT R\_14 FOREIGN KEY (ID\_Revenue) REFERENCES REVENUE(ID\_Revenue)

go

ALTER TABLE ISSUED

ADD CONSTRAINT R\_11 FOREIGN KEY (ID\_person) REFERENCES PERSON(ID\_person)

go

ALTER TABLE NORM

ADD CONSTRAINT R\_8 FOREIGN KEY (ID\_Workwear) REFERENCES WORKWEAR\_DIRECTORY(ID\_Workwear)

go

ALTER TABLE NORM

ADD CONSTRAINT R\_13 FOREIGN KEY (Id\_Position) REFERENCES EMPL\_POSITION(Id\_Position)

go

exec sp\_bindrule 'Val\_Gender', 'PERSON.Sex'

go

exec sp\_bindrule 'Val\_ShoeSize', 'PERSON.Shoe\_size'

go

exec sp\_bindrule 'Val\_Size', 'PERSON.Clothing\_size'

go

ALTER TABLE PERSON

ADD CONSTRAINT R\_10 FOREIGN KEY (Id\_Position) REFERENCES EMPL\_POSITION(Id\_Position)

go

exec sp\_bindrule 'Val\_Size', 'REVENUE.Clothing\_size'

go

exec sp\_bindrule 'Val\_ShoeSize', 'REVENUE.Shoe\_size'

go

ALTER TABLE REVENUE

ADD CONSTRAINT Default\_Value\_449\_1552948548

DEFAULT 0 FOR Issued

go

ALTER TABLE REVENUE

ADD CONSTRAINT R\_9 FOREIGN KEY (ID\_Workwear) REFERENCES WORKWEAR\_DIRECTORY(ID\_Workwear)

go

exec sp\_bindrule 'Val\_Unit', 'WORKWEAR\_DIRECTORY.Unit'

go

CREATE TRIGGER Insert\_Issued ON ISSUED

FOR INSERT

AS

DECLARE

@ID\_Revenue as int

BEGIN

SELECT @ID\_Revenue = ID\_Revenue

FROM inserted

UPDATE REVENUE

SET Issued = 'true'

WHERE ID\_Revenue = @ID\_Revenue

END

CREATE TRIGGER Insert\_Cancellation ON CANCELLATION

FOR INSERT

AS

DECLARE

@ID\_Issued as int

BEGIN

SELECT @ID\_Issued = ID\_Issued

FROM inserted

UPDATE ISSUED

SET Cancellation = 'true'

WHERE ID\_Issued = @ID\_Issued

END

CREATE TRIGGER tI\_CANCELLATION ON CANCELLATION FOR INSERT AS

/\* ERwin Builtin Trigger \*/

/\* INSERT trigger on CANCELLATION \*/

BEGIN

DECLARE @NUMROWS int,

@nullcnt int,

@validcnt int,

@errno int,

@errmsg varchar(255)

SELECT @NUMROWS = @@rowcount

/\* ERwin Builtin Trigger \*/

/\* ISSUED CANCELLATION on child insert restrict \*/

/\* ERWIN\_RELATION:CHECKSUM="00014ac6", PARENT\_OWNER="", PARENT\_TABLE="ISSUED"

CHILD\_OWNER="", CHILD\_TABLE="CANCELLATION"

P2C\_VERB\_PHRASE="", C2P\_VERB\_PHRASE="",

FK\_CONSTRAINT="R\_12", FK\_COLUMNS="ID\_Issued" \*/

IF

/\* %ChildFK(" OR",UPDATE) \*/

UPDATE(ID\_Issued)

BEGIN

SELECT @nullcnt = 0

SELECT @validcnt = count(\*)

FROM inserted,ISSUED

WHERE

/\* %JoinFKPK(inserted,ISSUED) \*/

inserted.ID\_Issued = ISSUED.ID\_Issued

/\* %NotnullFK(inserted," IS NULL","select @nullcnt = count(\*) from inserted where"," and") \*/

IF @validcnt + @nullcnt != @NUMROWS

BEGIN

SELECT @errno = 30002,

@errmsg = 'Cannot insert CANCELLATION because ISSUED does not exist.'

GOTO ERROR

END

END

/\* ERwin Builtin Trigger \*/

RETURN

ERROR:

raiserror @errno @errmsg

rollback transaction

END

go

CREATE TRIGGER tU\_CANCELLATION ON CANCELLATION FOR UPDATE AS

/\* ERwin Builtin Trigger \*/

/\* UPDATE trigger on CANCELLATION \*/

BEGIN

DECLARE @NUMROWS int,

@nullcnt int,

@validcnt int,

@insID\_Cancellation int,

@errno int,

@errmsg varchar(255)

SELECT @NUMROWS = @@rowcount

/\* ERwin Builtin Trigger \*/

/\* ISSUED CANCELLATION on child update restrict \*/

/\* ERWIN\_RELATION:CHECKSUM="000148ca", PARENT\_OWNER="", PARENT\_TABLE="ISSUED"

CHILD\_OWNER="", CHILD\_TABLE="CANCELLATION"

P2C\_VERB\_PHRASE="", C2P\_VERB\_PHRASE="",

FK\_CONSTRAINT="R\_12", FK\_COLUMNS="ID\_Issued" \*/

IF

/\* %ChildFK(" OR",UPDATE) \*/

UPDATE(ID\_Issued)

BEGIN

SELECT @nullcnt = 0

SELECT @validcnt = count(\*)

FROM inserted,ISSUED

WHERE

/\* %JoinFKPK(inserted,ISSUED) \*/

inserted.ID\_Issued = ISSUED.ID\_Issued

/\* %NotnullFK(inserted," IS NULL","select @nullcnt = count(\*) from inserted where"," AND") \*/

IF @validcnt + @nullcnt != @NUMROWS

BEGIN

SELECT @errno = 30007,

@errmsg = 'Cannot update CANCELLATION because ISSUED does not exist.'

GOTO ERROR

END

END

/\* ERwin Builtin Trigger \*/

RETURN

ERROR:

raiserror @errno @errmsg

rollback transaction

END

go

CREATE TRIGGER tI\_EMPL\_POSITION ON EMPL\_POSITION FOR INSERT AS

/\* ERwin Builtin Trigger \*/

/\* INSERT trigger on EMPL\_POSITION \*/

BEGIN

DECLARE @NUMROWS int,

@nullcnt int,

@validcnt int,

@errno int,

@errmsg varchar(255)

SELECT @NUMROWS = @@rowcount

/\* ERwin Builtin Trigger \*/

/\* MANUFACTORY EMPL\_POSITION on child insert restrict \*/

/\* ERWIN\_RELATION:CHECKSUM="0001550e", PARENT\_OWNER="", PARENT\_TABLE="MANUFACTORY"

CHILD\_OWNER="", CHILD\_TABLE="EMPL\_POSITION"

P2C\_VERB\_PHRASE="", C2P\_VERB\_PHRASE="",

FK\_CONSTRAINT="R\_15", FK\_COLUMNS="ID\_Manufactory" \*/

IF

/\* %ChildFK(" OR",UPDATE) \*/

UPDATE(ID\_Manufactory)

BEGIN

SELECT @nullcnt = 0

SELECT @validcnt = count(\*)

FROM inserted,MANUFACTORY

WHERE

/\* %JoinFKPK(inserted,MANUFACTORY) \*/

inserted.ID\_Manufactory = MANUFACTORY.ID\_Manufactory

/\* %NotnullFK(inserted," IS NULL","select @nullcnt = count(\*) from inserted where"," and") \*/

IF @validcnt + @nullcnt != @NUMROWS

BEGIN

SELECT @errno = 30002,

@errmsg = 'Cannot insert EMPL\_POSITION because MANUFACTORY does not exist.'

GOTO ERROR

END

END

/\* ERwin Builtin Trigger \*/

RETURN

ERROR:

raiserror @errno @errmsg

rollback transaction

END

go

CREATE TRIGGER tD\_EMPL\_POSITION ON EMPL\_POSITION FOR DELETE AS

/\* ERwin Builtin Trigger \*/

/\* DELETE trigger on EMPL\_POSITION \*/

BEGIN

DECLARE @errno int,

@errmsg varchar(255)

/\* ERwin Builtin Trigger \*/

/\* EMPL\_POSITION PERSON on parent delete restrict \*/

/\* ERWIN\_RELATION:CHECKSUM="0001d6c9", PARENT\_OWNER="", PARENT\_TABLE="EMPL\_POSITION"

CHILD\_OWNER="", CHILD\_TABLE="PERSON"

P2C\_VERB\_PHRASE="", C2P\_VERB\_PHRASE="",

FK\_CONSTRAINT="R\_10", FK\_COLUMNS="Id\_Position" \*/

IF EXISTS (

SELECT \* FROM deleted,PERSON

WHERE

/\* %JoinFKPK(PERSON,deleted," = "," AND") \*/

PERSON.Id\_Position = deleted.Id\_Position

)

BEGIN

SELECT @errno = 30001,

@errmsg = 'Cannot delete EMPL\_POSITION because PERSON exists.'

GOTO ERROR

END

/\* ERwin Builtin Trigger \*/

/\* EMPL\_POSITION NORM on parent delete restrict \*/

/\* ERWIN\_RELATION:CHECKSUM="00000000", PARENT\_OWNER="", PARENT\_TABLE="EMPL\_POSITION"

CHILD\_OWNER="", CHILD\_TABLE="NORM"

P2C\_VERB\_PHRASE="", C2P\_VERB\_PHRASE="",

FK\_CONSTRAINT="R\_13", FK\_COLUMNS="Id\_Position" \*/

IF EXISTS (

SELECT \* FROM deleted,NORM

WHERE

/\* %JoinFKPK(NORM,deleted," = "," AND") \*/

NORM.Id\_Position = deleted.Id\_Position

)

BEGIN

SELECT @errno = 30001,

@errmsg = 'Cannot delete EMPL\_POSITION because NORM exists.'

GOTO ERROR

END

/\* ERwin Builtin Trigger \*/

RETURN

ERROR:

raiserror @errno @errmsg

rollback transaction

END

go

CREATE TRIGGER tU\_EMPL\_POSITION ON EMPL\_POSITION FOR UPDATE AS

/\* ERwin Builtin Trigger \*/

/\* UPDATE trigger on EMPL\_POSITION \*/

BEGIN

DECLARE @NUMROWS int,

@nullcnt int,

@validcnt int,

@insId\_Position int,

@errno int,

@errmsg varchar(255)

SELECT @NUMROWS = @@rowcount

/\* ERwin Builtin Trigger \*/

/\* EMPL\_POSITION PERSON on parent update restrict \*/

/\* ERWIN\_RELATION:CHECKSUM="000370b2", PARENT\_OWNER="", PARENT\_TABLE="EMPL\_POSITION"

CHILD\_OWNER="", CHILD\_TABLE="PERSON"

P2C\_VERB\_PHRASE="", C2P\_VERB\_PHRASE="",

FK\_CONSTRAINT="R\_10", FK\_COLUMNS="Id\_Position" \*/

IF

/\* %ParentPK(" OR",UPDATE) \*/

UPDATE(Id\_Position)

BEGIN

IF EXISTS (

SELECT \* FROM deleted,PERSON

WHERE

/\* %JoinFKPK(PERSON,deleted," = "," AND") \*/

PERSON.Id\_Position = deleted.Id\_Position

)

BEGIN

SELECT @errno = 30005,

@errmsg = 'Cannot update EMPL\_POSITION because PERSON exists.'

GOTO ERROR

END

END

/\* ERwin Builtin Trigger \*/

/\* EMPL\_POSITION NORM on parent update restrict \*/

/\* ERWIN\_RELATION:CHECKSUM="00000000", PARENT\_OWNER="", PARENT\_TABLE="EMPL\_POSITION"

CHILD\_OWNER="", CHILD\_TABLE="NORM"

P2C\_VERB\_PHRASE="", C2P\_VERB\_PHRASE="",

FK\_CONSTRAINT="R\_13", FK\_COLUMNS="Id\_Position" \*/

IF

/\* %ParentPK(" OR",UPDATE) \*/

UPDATE(Id\_Position)

BEGIN

IF EXISTS (

SELECT \* FROM deleted,NORM

WHERE

/\* %JoinFKPK(NORM,deleted," = "," AND") \*/

NORM.Id\_Position = deleted.Id\_Position

)

BEGIN

SELECT @errno = 30005,

@errmsg = 'Cannot update EMPL\_POSITION because NORM exists.'

GOTO ERROR

END

END

/\* ERwin Builtin Trigger \*/

/\* MANUFACTORY EMPL\_POSITION on child update restrict \*/

/\* ERWIN\_RELATION:CHECKSUM="00000000", PARENT\_OWNER="", PARENT\_TABLE="MANUFACTORY"

CHILD\_OWNER="", CHILD\_TABLE="EMPL\_POSITION"

P2C\_VERB\_PHRASE="", C2P\_VERB\_PHRASE="",

FK\_CONSTRAINT="R\_15", FK\_COLUMNS="ID\_Manufactory" \*/

IF

/\* %ChildFK(" OR",UPDATE) \*/

UPDATE(ID\_Manufactory)

BEGIN

SELECT @nullcnt = 0

SELECT @validcnt = count(\*)

FROM inserted,MANUFACTORY

WHERE

/\* %JoinFKPK(inserted,MANUFACTORY) \*/

inserted.ID\_Manufactory = MANUFACTORY.ID\_Manufactory

/\* %NotnullFK(inserted," IS NULL","select @nullcnt = count(\*) from inserted where"," AND") \*/

IF @validcnt + @nullcnt != @NUMROWS

BEGIN

SELECT @errno = 30007,

@errmsg = 'Cannot update EMPL\_POSITION because MANUFACTORY does not exist.'

GOTO ERROR

END

END

/\* ERwin Builtin Trigger \*/

RETURN

ERROR:

raiserror @errno @errmsg

rollback transaction

END

go

CREATE TRIGGER tD\_ISSUED ON ISSUED FOR DELETE AS

/\* ERwin Builtin Trigger \*/

/\* DELETE trigger on ISSUED \*/

BEGIN

DECLARE @errno int,

@errmsg varchar(255)

/\* ERwin Builtin Trigger \*/

/\* ISSUED CANCELLATION on parent delete restrict \*/

/\* ERWIN\_RELATION:CHECKSUM="0000ef9e", PARENT\_OWNER="", PARENT\_TABLE="ISSUED"

CHILD\_OWNER="", CHILD\_TABLE="CANCELLATION"

P2C\_VERB\_PHRASE="", C2P\_VERB\_PHRASE="",

FK\_CONSTRAINT="R\_12", FK\_COLUMNS="ID\_Issued" \*/

IF EXISTS (

SELECT \* FROM deleted,CANCELLATION

WHERE

/\* %JoinFKPK(CANCELLATION,deleted," = "," AND") \*/

CANCELLATION.ID\_Issued = deleted.ID\_Issued

)

BEGIN

SELECT @errno = 30001,

@errmsg = 'Cannot delete ISSUED because CANCELLATION exists.'

GOTO ERROR

END

/\* ERwin Builtin Trigger \*/

RETURN

ERROR:

raiserror @errno @errmsg

rollback transaction

END

go

CREATE TRIGGER tU\_ISSUED ON ISSUED FOR UPDATE AS

/\* ERwin Builtin Trigger \*/

/\* UPDATE trigger on ISSUED \*/

BEGIN

DECLARE @NUMROWS int,

@nullcnt int,

@validcnt int,

@insID\_Issued int,

@errno int,

@errmsg varchar(255)

SELECT @NUMROWS = @@rowcount

/\* ERwin Builtin Trigger \*/

/\* ISSUED CANCELLATION on parent update restrict \*/

/\* ERWIN\_RELATION:CHECKSUM="000374ad", PARENT\_OWNER="", PARENT\_TABLE="ISSUED"

CHILD\_OWNER="", CHILD\_TABLE="CANCELLATION"

P2C\_VERB\_PHRASE="", C2P\_VERB\_PHRASE="",

FK\_CONSTRAINT="R\_12", FK\_COLUMNS="ID\_Issued" \*/

IF

/\* %ParentPK(" OR",UPDATE) \*/

UPDATE(ID\_Issued)

BEGIN

IF EXISTS (

SELECT \* FROM deleted,CANCELLATION

WHERE

/\* %JoinFKPK(CANCELLATION,deleted," = "," AND") \*/

CANCELLATION.ID\_Issued = deleted.ID\_Issued

)

BEGIN

SELECT @errno = 30005,

@errmsg = 'Cannot update ISSUED because CANCELLATION exists.'

GOTO ERROR

END

END

/\* ERwin Builtin Trigger \*/

/\* REVENUE ISSUED on child update restrict \*/

/\* ERWIN\_RELATION:CHECKSUM="00000000", PARENT\_OWNER="", PARENT\_TABLE="REVENUE"

CHILD\_OWNER="", CHILD\_TABLE="ISSUED"

P2C\_VERB\_PHRASE="", C2P\_VERB\_PHRASE="",

FK\_CONSTRAINT="R\_14", FK\_COLUMNS="ID\_Revenue" \*/

IF

/\* %ChildFK(" OR",UPDATE) \*/

UPDATE(ID\_Revenue)

BEGIN

SELECT @nullcnt = 0

SELECT @validcnt = count(\*)

FROM inserted,REVENUE

WHERE

/\* %JoinFKPK(inserted,REVENUE) \*/

inserted.ID\_Revenue = REVENUE.ID\_Revenue

/\* %NotnullFK(inserted," IS NULL","select @nullcnt = count(\*) from inserted where"," AND") \*/

IF @validcnt + @nullcnt != @NUMROWS

BEGIN

SELECT @errno = 30007,

@errmsg = 'Cannot update ISSUED because REVENUE does not exist.'

GOTO ERROR

END

END

/\* ERwin Builtin Trigger \*/

/\* PERSON ISSUED on child update restrict \*/

/\* ERWIN\_RELATION:CHECKSUM="00000000", PARENT\_OWNER="", PARENT\_TABLE="PERSON"

CHILD\_OWNER="", CHILD\_TABLE="ISSUED"

P2C\_VERB\_PHRASE="", C2P\_VERB\_PHRASE="",

FK\_CONSTRAINT="R\_11", FK\_COLUMNS="ID\_person" \*/

IF

/\* %ChildFK(" OR",UPDATE) \*/

UPDATE(ID\_person)

BEGIN

SELECT @nullcnt = 0

SELECT @validcnt = count(\*)

FROM inserted,PERSON

WHERE

/\* %JoinFKPK(inserted,PERSON) \*/

inserted.ID\_person = PERSON.ID\_person

/\* %NotnullFK(inserted," IS NULL","select @nullcnt = count(\*) from inserted where"," AND") \*/

IF @validcnt + @nullcnt != @NUMROWS

BEGIN

SELECT @errno = 30007,

@errmsg = 'Cannot update ISSUED because PERSON does not exist.'

GOTO ERROR

END

END

/\* ERwin Builtin Trigger \*/

RETURN

ERROR:

raiserror @errno @errmsg

rollback transaction

END

go

CREATE TRIGGER tD\_MANUFACTORY ON MANUFACTORY FOR DELETE AS

/\* ERwin Builtin Trigger \*/

/\* DELETE trigger on MANUFACTORY \*/

BEGIN

DECLARE @errno int,

@errmsg varchar(255)

/\* ERwin Builtin Trigger \*/

/\* MANUFACTORY EMPL\_POSITION on parent delete restrict \*/

/\* ERWIN\_RELATION:CHECKSUM="0001073b", PARENT\_OWNER="", PARENT\_TABLE="MANUFACTORY"

CHILD\_OWNER="", CHILD\_TABLE="EMPL\_POSITION"

P2C\_VERB\_PHRASE="", C2P\_VERB\_PHRASE="",

FK\_CONSTRAINT="R\_15", FK\_COLUMNS="ID\_Manufactory" \*/

IF EXISTS (

SELECT \* FROM deleted,EMPL\_POSITION

WHERE

/\* %JoinFKPK(EMPL\_POSITION,deleted," = "," AND") \*/

EMPL\_POSITION.ID\_Manufactory = deleted.ID\_Manufactory

)

BEGIN

SELECT @errno = 30001,

@errmsg = 'Cannot delete MANUFACTORY because EMPL\_POSITION exists.'

GOTO ERROR

END

/\* ERwin Builtin Trigger \*/

RETURN

ERROR:

raiserror @errno @errmsg

rollback transaction

END

go

CREATE TRIGGER tU\_MANUFACTORY ON MANUFACTORY FOR UPDATE AS

/\* ERwin Builtin Trigger \*/

/\* UPDATE trigger on MANUFACTORY \*/

BEGIN

DECLARE @NUMROWS int,

@nullcnt int,

@validcnt int,

@insID\_Manufactory int,

@errno int,

@errmsg varchar(255)

SELECT @NUMROWS = @@rowcount

/\* ERwin Builtin Trigger \*/

/\* MANUFACTORY EMPL\_POSITION on parent update restrict \*/

/\* ERWIN\_RELATION:CHECKSUM="00011fce", PARENT\_OWNER="", PARENT\_TABLE="MANUFACTORY"

CHILD\_OWNER="", CHILD\_TABLE="EMPL\_POSITION"

P2C\_VERB\_PHRASE="", C2P\_VERB\_PHRASE="",

FK\_CONSTRAINT="R\_15", FK\_COLUMNS="ID\_Manufactory" \*/

IF

/\* %ParentPK(" OR",UPDATE) \*/

UPDATE(ID\_Manufactory)

BEGIN

IF EXISTS (

SELECT \* FROM deleted,EMPL\_POSITION

WHERE

/\* %JoinFKPK(EMPL\_POSITION,deleted," = "," AND") \*/

EMPL\_POSITION.ID\_Manufactory = deleted.ID\_Manufactory

)

BEGIN

SELECT @errno = 30005,

@errmsg = 'Cannot update MANUFACTORY because EMPL\_POSITION exists.'

GOTO ERROR

END

END

/\* ERwin Builtin Trigger \*/

RETURN

ERROR:

raiserror @errno @errmsg

rollback transaction

END

go

CREATE TRIGGER tI\_NORM ON NORM FOR INSERT AS

/\* ERwin Builtin Trigger \*/

/\* INSERT trigger on NORM \*/

BEGIN

DECLARE @NUMROWS int,

@nullcnt int,

@validcnt int,

@errno int,

@errmsg varchar(255)

SELECT @NUMROWS = @@rowcount

/\* ERwin Builtin Trigger \*/

/\* WORKWEAR\_DIRECTORY NORM on child insert restrict \*/

/\* ERWIN\_RELATION:CHECKSUM="0002b6e2", PARENT\_OWNER="", PARENT\_TABLE="WORKWEAR\_DIRECTORY"

CHILD\_OWNER="", CHILD\_TABLE="NORM"

P2C\_VERB\_PHRASE="", C2P\_VERB\_PHRASE="",

FK\_CONSTRAINT="R\_8", FK\_COLUMNS="ID\_Workwear" \*/

IF

/\* %ChildFK(" OR",UPDATE) \*/

UPDATE(ID\_Workwear)

BEGIN

SELECT @nullcnt = 0

SELECT @validcnt = count(\*)

FROM inserted,WORKWEAR\_DIRECTORY

WHERE

/\* %JoinFKPK(inserted,WORKWEAR\_DIRECTORY) \*/

inserted.ID\_Workwear = WORKWEAR\_DIRECTORY.ID\_Workwear

/\* %NotnullFK(inserted," IS NULL","select @nullcnt = count(\*) from inserted where"," and") \*/

IF @validcnt + @nullcnt != @NUMROWS

BEGIN

SELECT @errno = 30002,

@errmsg = 'Cannot insert NORM because WORKWEAR\_DIRECTORY does not exist.'

GOTO ERROR

END

END

/\* ERwin Builtin Trigger \*/

/\* EMPL\_POSITION NORM on child insert restrict \*/

/\* ERWIN\_RELATION:CHECKSUM="00000000", PARENT\_OWNER="", PARENT\_TABLE="EMPL\_POSITION"

CHILD\_OWNER="", CHILD\_TABLE="NORM"

P2C\_VERB\_PHRASE="", C2P\_VERB\_PHRASE="",

FK\_CONSTRAINT="R\_13", FK\_COLUMNS="Id\_Position" \*/

IF

/\* %ChildFK(" OR",UPDATE) \*/

UPDATE(Id\_Position)

BEGIN

SELECT @nullcnt = 0

SELECT @validcnt = count(\*)

FROM inserted,EMPL\_POSITION

WHERE

/\* %JoinFKPK(inserted,EMPL\_POSITION) \*/

inserted.Id\_Position = EMPL\_POSITION.Id\_Position

/\* %NotnullFK(inserted," IS NULL","select @nullcnt = count(\*) from inserted where"," and") \*/

IF @validcnt + @nullcnt != @NUMROWS

BEGIN

SELECT @errno = 30002,

@errmsg = 'Cannot insert NORM because EMPL\_POSITION does not exist.'

GOTO ERROR

END

END

/\* ERwin Builtin Trigger \*/

RETURN

ERROR:

raiserror @errno @errmsg

rollback transaction

END

go

CREATE TRIGGER tU\_NORM ON NORM FOR UPDATE AS

/\* ERwin Builtin Trigger \*/

/\* UPDATE trigger on NORM \*/

BEGIN

DECLARE @NUMROWS int,

@nullcnt int,

@validcnt int,

@insID\_Norm int,

@errno int,

@errmsg varchar(255)

SELECT @NUMROWS = @@rowcount

/\* ERwin Builtin Trigger \*/

/\* WORKWEAR\_DIRECTORY NORM on child update restrict \*/

/\* ERWIN\_RELATION:CHECKSUM="0002ba31", PARENT\_OWNER="", PARENT\_TABLE="WORKWEAR\_DIRECTORY"

CHILD\_OWNER="", CHILD\_TABLE="NORM"

P2C\_VERB\_PHRASE="", C2P\_VERB\_PHRASE="",

FK\_CONSTRAINT="R\_8", FK\_COLUMNS="ID\_Workwear" \*/

IF

/\* %ChildFK(" OR",UPDATE) \*/

UPDATE(ID\_Workwear)

BEGIN

SELECT @nullcnt = 0

SELECT @validcnt = count(\*)

FROM inserted,WORKWEAR\_DIRECTORY

WHERE

/\* %JoinFKPK(inserted,WORKWEAR\_DIRECTORY) \*/

inserted.ID\_Workwear = WORKWEAR\_DIRECTORY.ID\_Workwear

/\* %NotnullFK(inserted," IS NULL","select @nullcnt = count(\*) from inserted where"," AND") \*/

IF @validcnt + @nullcnt != @NUMROWS

BEGIN

SELECT @errno = 30007,

@errmsg = 'Cannot update NORM because WORKWEAR\_DIRECTORY does not exist.'

GOTO ERROR

END

END

/\* ERwin Builtin Trigger \*/

/\* EMPL\_POSITION NORM on child update restrict \*/

/\* ERWIN\_RELATION:CHECKSUM="00000000", PARENT\_OWNER="", PARENT\_TABLE="EMPL\_POSITION"

CHILD\_OWNER="", CHILD\_TABLE="NORM"

P2C\_VERB\_PHRASE="", C2P\_VERB\_PHRASE="",

FK\_CONSTRAINT="R\_13", FK\_COLUMNS="Id\_Position" \*/

IF

/\* %ChildFK(" OR",UPDATE) \*/

UPDATE(Id\_Position)

BEGIN

SELECT @nullcnt = 0

SELECT @validcnt = count(\*)

FROM inserted,EMPL\_POSITION

WHERE

/\* %JoinFKPK(inserted,EMPL\_POSITION) \*/

inserted.Id\_Position = EMPL\_POSITION.Id\_Position

/\* %NotnullFK(inserted," IS NULL","select @nullcnt = count(\*) from inserted where"," AND") \*/

IF @validcnt + @nullcnt != @NUMROWS

BEGIN

SELECT @errno = 30007,

@errmsg = 'Cannot update NORM because EMPL\_POSITION does not exist.'

GOTO ERROR

END

END

/\* ERwin Builtin Trigger \*/

RETURN

ERROR:

raiserror @errno @errmsg

rollback transaction

END

go

CREATE TRIGGER tI\_PERSON ON PERSON FOR INSERT AS

/\* ERwin Builtin Trigger \*/

/\* INSERT trigger on PERSON \*/

BEGIN

DECLARE @NUMROWS int,

@nullcnt int,

@validcnt int,

@errno int,

@errmsg varchar(255)

SELECT @NUMROWS = @@rowcount

/\* ERwin Builtin Trigger \*/

/\* EMPL\_POSITION PERSON on child insert restrict \*/

/\* ERWIN\_RELATION:CHECKSUM="00015ad9", PARENT\_OWNER="", PARENT\_TABLE="EMPL\_POSITION"

CHILD\_OWNER="", CHILD\_TABLE="PERSON"

P2C\_VERB\_PHRASE="", C2P\_VERB\_PHRASE="",

FK\_CONSTRAINT="R\_10", FK\_COLUMNS="Id\_Position" \*/

IF

/\* %ChildFK(" OR",UPDATE) \*/

UPDATE(Id\_Position)

BEGIN

SELECT @nullcnt = 0

SELECT @validcnt = count(\*)

FROM inserted,EMPL\_POSITION

WHERE

/\* %JoinFKPK(inserted,EMPL\_POSITION) \*/

inserted.Id\_Position = EMPL\_POSITION.Id\_Position

/\* %NotnullFK(inserted," IS NULL","select @nullcnt = count(\*) from inserted where"," and") \*/

IF @validcnt + @nullcnt != @NUMROWS

BEGIN

SELECT @errno = 30002,

@errmsg = 'Cannot insert PERSON because EMPL\_POSITION does not exist.'

GOTO ERROR

END

END

/\* ERwin Builtin Trigger \*/

RETURN

ERROR:

raiserror @errno @errmsg

rollback transaction

END

go

CREATE TRIGGER tD\_PERSON ON PERSON FOR DELETE AS

/\* ERwin Builtin Trigger \*/

/\* DELETE trigger on PERSON \*/

BEGIN

DECLARE @errno int,

@errmsg varchar(255)

/\* ERwin Builtin Trigger \*/

/\* PERSON ISSUED on parent delete restrict \*/

/\* ERWIN\_RELATION:CHECKSUM="0000ec73", PARENT\_OWNER="", PARENT\_TABLE="PERSON"

CHILD\_OWNER="", CHILD\_TABLE="ISSUED"

P2C\_VERB\_PHRASE="", C2P\_VERB\_PHRASE="",

FK\_CONSTRAINT="R\_11", FK\_COLUMNS="ID\_person" \*/

IF EXISTS (

SELECT \* FROM deleted,ISSUED

WHERE

/\* %JoinFKPK(ISSUED,deleted," = "," AND") \*/

ISSUED.ID\_person = deleted.ID\_person

)

BEGIN

SELECT @errno = 30001,

@errmsg = 'Cannot delete PERSON because ISSUED exists.'

GOTO ERROR

END

/\* ERwin Builtin Trigger \*/

RETURN

ERROR:

raiserror @errno @errmsg

rollback transaction

END

go

CREATE TRIGGER tU\_PERSON ON PERSON FOR UPDATE AS

/\* ERwin Builtin Trigger \*/

/\* UPDATE trigger on PERSON \*/

BEGIN

DECLARE @NUMROWS int,

@nullcnt int,

@validcnt int,

@insID\_person int,

@errno int,

@errmsg varchar(255)

SELECT @NUMROWS = @@rowcount

/\* ERwin Builtin Trigger \*/

/\* PERSON ISSUED on parent update restrict \*/

/\* ERWIN\_RELATION:CHECKSUM="00026a4f", PARENT\_OWNER="", PARENT\_TABLE="PERSON"

CHILD\_OWNER="", CHILD\_TABLE="ISSUED"

P2C\_VERB\_PHRASE="", C2P\_VERB\_PHRASE="",

FK\_CONSTRAINT="R\_11", FK\_COLUMNS="ID\_person" \*/

IF

/\* %ParentPK(" OR",UPDATE) \*/

UPDATE(ID\_person)

BEGIN

IF EXISTS (

SELECT \* FROM deleted,ISSUED

WHERE

/\* %JoinFKPK(ISSUED,deleted," = "," AND") \*/

ISSUED.ID\_person = deleted.ID\_person

)

BEGIN

SELECT @errno = 30005,

@errmsg = 'Cannot update PERSON because ISSUED exists.'

GOTO ERROR

END

END

/\* ERwin Builtin Trigger \*/

/\* EMPL\_POSITION PERSON on child update restrict \*/

/\* ERWIN\_RELATION:CHECKSUM="00000000", PARENT\_OWNER="", PARENT\_TABLE="EMPL\_POSITION"

CHILD\_OWNER="", CHILD\_TABLE="PERSON"

P2C\_VERB\_PHRASE="", C2P\_VERB\_PHRASE="",

FK\_CONSTRAINT="R\_10", FK\_COLUMNS="Id\_Position" \*/

IF

/\* %ChildFK(" OR",UPDATE) \*/

UPDATE(Id\_Position)

BEGIN

SELECT @nullcnt = 0

SELECT @validcnt = count(\*)

FROM inserted,EMPL\_POSITION

WHERE

/\* %JoinFKPK(inserted,EMPL\_POSITION) \*/

inserted.Id\_Position = EMPL\_POSITION.Id\_Position

/\* %NotnullFK(inserted," IS NULL","select @nullcnt = count(\*) from inserted where"," AND") \*/

IF @validcnt + @nullcnt != @NUMROWS

BEGIN

SELECT @errno = 30007,

@errmsg = 'Cannot update PERSON because EMPL\_POSITION does not exist.'

GOTO ERROR

END

END

/\* ERwin Builtin Trigger \*/

RETURN

ERROR:

raiserror @errno @errmsg

rollback transaction

END

go

CREATE TRIGGER tI\_REVENUE ON REVENUE FOR INSERT AS

/\* ERwin Builtin Trigger \*/

/\* INSERT trigger on REVENUE \*/

BEGIN

DECLARE @NUMROWS int,

@nullcnt int,

@validcnt int,

@errno int,

@errmsg varchar(255)

SELECT @NUMROWS = @@rowcount

/\* ERwin Builtin Trigger \*/

/\* WORKWEAR\_DIRECTORY REVENUE on child insert restrict \*/

/\* ERWIN\_RELATION:CHECKSUM="00016cf0", PARENT\_OWNER="", PARENT\_TABLE="WORKWEAR\_DIRECTORY"

CHILD\_OWNER="", CHILD\_TABLE="REVENUE"

P2C\_VERB\_PHRASE="", C2P\_VERB\_PHRASE="",

FK\_CONSTRAINT="R\_9", FK\_COLUMNS="ID\_Workwear" \*/

IF

/\* %ChildFK(" OR",UPDATE) \*/

UPDATE(ID\_Workwear)

BEGIN

SELECT @nullcnt = 0

SELECT @validcnt = count(\*)

FROM inserted,WORKWEAR\_DIRECTORY

WHERE

/\* %JoinFKPK(inserted,WORKWEAR\_DIRECTORY) \*/

inserted.ID\_Workwear = WORKWEAR\_DIRECTORY.ID\_Workwear

/\* %NotnullFK(inserted," IS NULL","select @nullcnt = count(\*) from inserted where"," and") \*/

IF @validcnt + @nullcnt != @NUMROWS

BEGIN

SELECT @errno = 30002,

@errmsg = 'Cannot insert REVENUE because WORKWEAR\_DIRECTORY does not exist.'

GOTO ERROR

END

END

/\* ERwin Builtin Trigger \*/

RETURN

ERROR:

raiserror @errno @errmsg

rollback transaction

END

go

CREATE TRIGGER tD\_REVENUE ON REVENUE FOR DELETE AS

/\* ERwin Builtin Trigger \*/

/\* DELETE trigger on REVENUE \*/

BEGIN

DECLARE @errno int,

@errmsg varchar(255)

/\* ERwin Builtin Trigger \*/

/\* REVENUE ISSUED on parent delete restrict \*/

/\* ERWIN\_RELATION:CHECKSUM="0000e3d5", PARENT\_OWNER="", PARENT\_TABLE="REVENUE"

CHILD\_OWNER="", CHILD\_TABLE="ISSUED"

P2C\_VERB\_PHRASE="", C2P\_VERB\_PHRASE="",

FK\_CONSTRAINT="R\_14", FK\_COLUMNS="ID\_Revenue" \*/

IF EXISTS (

SELECT \* FROM deleted,ISSUED

WHERE

/\* %JoinFKPK(ISSUED,deleted," = "," AND") \*/

ISSUED.ID\_Revenue = deleted.ID\_Revenue

)

BEGIN

SELECT @errno = 30001,

@errmsg = 'Cannot delete REVENUE because ISSUED exists.'

GOTO ERROR

END

/\* ERwin Builtin Trigger \*/

RETURN

ERROR:

raiserror @errno @errmsg

rollback transaction

END

go

CREATE TRIGGER tU\_REVENUE ON REVENUE FOR UPDATE AS

/\* ERwin Builtin Trigger \*/

/\* UPDATE trigger on REVENUE \*/

BEGIN

DECLARE @NUMROWS int,

@nullcnt int,

@validcnt int,

@insID\_Revenue int,

@errno int,

@errmsg varchar(255)

SELECT @NUMROWS = @@rowcount

/\* ERwin Builtin Trigger \*/

/\* REVENUE ISSUED on parent update restrict \*/

/\* ERWIN\_RELATION:CHECKSUM="000259d3", PARENT\_OWNER="", PARENT\_TABLE="REVENUE"

CHILD\_OWNER="", CHILD\_TABLE="ISSUED"

P2C\_VERB\_PHRASE="", C2P\_VERB\_PHRASE="",

FK\_CONSTRAINT="R\_14", FK\_COLUMNS="ID\_Revenue" \*/

IF

/\* %ParentPK(" OR",UPDATE) \*/

UPDATE(ID\_Revenue)

BEGIN

IF EXISTS (

SELECT \* FROM deleted,ISSUED

WHERE

/\* %JoinFKPK(ISSUED,deleted," = "," AND") \*/

ISSUED.ID\_Revenue = deleted.ID\_Revenue

)

BEGIN

SELECT @errno = 30005,

@errmsg = 'Cannot update REVENUE because ISSUED exists.'

GOTO ERROR

END

END

/\* ERwin Builtin Trigger \*/

/\* WORKWEAR\_DIRECTORY REVENUE on child update restrict \*/

/\* ERWIN\_RELATION:CHECKSUM="00000000", PARENT\_OWNER="", PARENT\_TABLE="WORKWEAR\_DIRECTORY"

CHILD\_OWNER="", CHILD\_TABLE="REVENUE"

P2C\_VERB\_PHRASE="", C2P\_VERB\_PHRASE="",

FK\_CONSTRAINT="R\_9", FK\_COLUMNS="ID\_Workwear" \*/

IF

/\* %ChildFK(" OR",UPDATE) \*/

UPDATE(ID\_Workwear)

BEGIN

SELECT @nullcnt = 0

SELECT @validcnt = count(\*)

FROM inserted,WORKWEAR\_DIRECTORY

WHERE

/\* %JoinFKPK(inserted,WORKWEAR\_DIRECTORY) \*/

inserted.ID\_Workwear = WORKWEAR\_DIRECTORY.ID\_Workwear

/\* %NotnullFK(inserted," IS NULL","select @nullcnt = count(\*) from inserted where"," AND") \*/

IF @validcnt + @nullcnt != @NUMROWS

BEGIN

SELECT @errno = 30007,

@errmsg = 'Cannot update REVENUE because WORKWEAR\_DIRECTORY does not exist.'

GOTO ERROR

END

END

/\* ERwin Builtin Trigger \*/

RETURN

ERROR:

raiserror @errno @errmsg

rollback transaction

END

go

CREATE TRIGGER tD\_WORKWEAR\_DIRECTORY ON WORKWEAR\_DIRECTORY FOR DELETE AS

/\* ERwin Builtin Trigger \*/

/\* DELETE trigger on WORKWEAR\_DIRECTORY \*/

BEGIN

DECLARE @errno int,

@errmsg varchar(255)

/\* ERwin Builtin Trigger \*/

/\* WORKWEAR\_DIRECTORY NORM on parent delete restrict \*/

/\* ERWIN\_RELATION:CHECKSUM="0001e55b", PARENT\_OWNER="", PARENT\_TABLE="WORKWEAR\_DIRECTORY"

CHILD\_OWNER="", CHILD\_TABLE="NORM"

P2C\_VERB\_PHRASE="", C2P\_VERB\_PHRASE="",

FK\_CONSTRAINT="R\_8", FK\_COLUMNS="ID\_Workwear" \*/

IF EXISTS (

SELECT \* FROM deleted,NORM

WHERE

/\* %JoinFKPK(NORM,deleted," = "," AND") \*/

NORM.ID\_Workwear = deleted.ID\_Workwear

)

BEGIN

SELECT @errno = 30001,

@errmsg = 'Cannot delete WORKWEAR\_DIRECTORY because NORM exists.'

GOTO ERROR

END

/\* ERwin Builtin Trigger \*/

/\* WORKWEAR\_DIRECTORY REVENUE on parent delete restrict \*/

/\* ERWIN\_RELATION:CHECKSUM="00000000", PARENT\_OWNER="", PARENT\_TABLE="WORKWEAR\_DIRECTORY"

CHILD\_OWNER="", CHILD\_TABLE="REVENUE"

P2C\_VERB\_PHRASE="", C2P\_VERB\_PHRASE="",

FK\_CONSTRAINT="R\_9", FK\_COLUMNS="ID\_Workwear" \*/

IF EXISTS (

SELECT \* FROM deleted,REVENUE

WHERE

/\* %JoinFKPK(REVENUE,deleted," = "," AND") \*/

REVENUE.ID\_Workwear = deleted.ID\_Workwear

)

BEGIN

SELECT @errno = 30001,

@errmsg = 'Cannot delete WORKWEAR\_DIRECTORY because REVENUE exists.'

GOTO ERROR

END

/\* ERwin Builtin Trigger \*/

RETURN

ERROR:

raiserror @errno @errmsg

rollback transaction

END

go

CREATE TRIGGER tU\_WORKWEAR\_DIRECTORY ON WORKWEAR\_DIRECTORY FOR UPDATE AS

/\* ERwin Builtin Trigger \*/

/\* UPDATE trigger on WORKWEAR\_DIRECTORY \*/

BEGIN

DECLARE @NUMROWS int,

@nullcnt int,

@validcnt int,

@insID\_Workwear int,

@errno int,

@errmsg varchar(255)

SELECT @NUMROWS = @@rowcount

/\* ERwin Builtin Trigger \*/

/\* WORKWEAR\_DIRECTORY NORM on parent update restrict \*/

/\* ERWIN\_RELATION:CHECKSUM="00022100", PARENT\_OWNER="", PARENT\_TABLE="WORKWEAR\_DIRECTORY"

CHILD\_OWNER="", CHILD\_TABLE="NORM"

P2C\_VERB\_PHRASE="", C2P\_VERB\_PHRASE="",

FK\_CONSTRAINT="R\_8", FK\_COLUMNS="ID\_Workwear" \*/

IF

/\* %ParentPK(" OR",UPDATE) \*/

UPDATE(ID\_Workwear)

BEGIN

IF EXISTS (

SELECT \* FROM deleted,NORM

WHERE

/\* %JoinFKPK(NORM,deleted," = "," AND") \*/

NORM.ID\_Workwear = deleted.ID\_Workwear

)

BEGIN

SELECT @errno = 30005,

@errmsg = 'Cannot update WORKWEAR\_DIRECTORY because NORM exists.'

GOTO ERROR

END

END

/\* ERwin Builtin Trigger \*/

/\* WORKWEAR\_DIRECTORY REVENUE on parent update restrict \*/

/\* ERWIN\_RELATION:CHECKSUM="00000000", PARENT\_OWNER="", PARENT\_TABLE="WORKWEAR\_DIRECTORY"

CHILD\_OWNER="", CHILD\_TABLE="REVENUE"

P2C\_VERB\_PHRASE="", C2P\_VERB\_PHRASE="",

FK\_CONSTRAINT="R\_9", FK\_COLUMNS="ID\_Workwear" \*/

IF

/\* %ParentPK(" OR",UPDATE) \*/

UPDATE(ID\_Workwear)

BEGIN

IF EXISTS (

SELECT \* FROM deleted,REVENUE

WHERE

/\* %JoinFKPK(REVENUE,deleted," = "," AND") \*/

REVENUE.ID\_Workwear = deleted.ID\_Workwear

)

BEGIN

SELECT @errno = 30005,

@errmsg = 'Cannot update WORKWEAR\_DIRECTORY because REVENUE exists.'

GOTO ERROR

END

END

/\* ERwin Builtin Trigger \*/

RETURN

ERROR:

raiserror @errno @errmsg

rollback transaction

END

go