```
CREATE TABLE [Account]

(

        [AccountID]            integer  NOT NULL ,

        [AccName]              varchar(150)  NULL ,

        [StartYear]            datetime  NULL ,

        [EndYear]              datetime  NULL

)

go



ALTER TABLE [Account]

        ADD CONSTRAINT [XPKAccount] PRIMARY KEY  CLUSTERED ([AccountID] ASC)

go



CREATE TABLE [Brokrage]

(

        [BrokerID]            integer  NOT NULL ,

        [BrokrageRate]        char(18)  NULL ,

        [BrokrageForItem]     char(18)  NULL

)

go



ALTER TABLE [Brokrage]

        ADD CONSTRAINT [XPKBrokrage] PRIMARY KEY  CLUSTERED ([BrokerID] ASC)
```

```
go


CREATE TABLE [Buy]

(

        [AccountID]         integer  NOT NULL ,

        [ContractID]        integer  NOT NULL ,

        [BuyID]             char(18)  NOT NULL ,

        [Rate]              char(18)  NULL ,

        [Quantity]          char(18)  NULL ,

        [BuyDate]           char(18)  NULL ,

        [BrokerID]          integer  NULL ,

        [TaxID]             integer  NULL ,

        [Brokrage]          char(18)  NULL ,

        [Tax]               char(18)  NULL ,

        [Total]             char(18)  NULL

)

go


ALTER TABLE [Buy]

        ADD CONSTRAINT [XPKAccount_Contract] PRIMARY KEY  CLUSTERED ([BuyID]
ASC,[AccountID] ASC,[ContractID] ASC)

go


CREATE TABLE [Contract]

(
```

```
        [ContractID]        integer  NOT NULL ,

        [ContractName]      char(18)  NULL ,

        [OpenPrice]         char(18)  NULL ,

        [ClosePrice]        char(18)  NULL ,

        [HighPrice]         char(18)  NULL ,

        [LowPrice]          char(18)  NULL ,

        [Type]              char(18)  NULL ,

        [ExpiryDate]        char(18)  NULL
)
go


ALTER TABLE [Contract]

        ADD CONSTRAINT [XPKContract] PRIMARY KEY  CLUSTERED ([ContractID] ASC)
go


CREATE TABLE [IntraDay]
(
        [TransactionID]     integer  NOT NULL ,

        [BRate]             char(18)  NULL ,

        [SRate]             char(18)  NULL ,

        [Quantity]          char(18)  NULL ,

        [TransactionDate]   char(18)  NULL ,

        [AccountID]         integer  NOT NULL ,

        [ContractID]        integer  NOT NULL ,
```

```
        [BrokerID]          integer  NULL ,

        [TaxID]             integer  NULL ,

        [Brokrage]          char(18)  NULL ,

        [Tax]               char(18)  NULL ,

        [Total]             char(18)  NULL

)

go



ALTER TABLE [IntraDay]

        ADD CONSTRAINT [XPKIntraDay] PRIMARY KEY  CLUSTERED ([TransactionID]
ASC,[AccountID] ASC,[ContractID] ASC)

go



CREATE TABLE [Inventory]

(

        [AverageCost]       char(18)  NULL ,

        [AccountID]         integer  NOT NULL ,

        [ContractID]        integer  NOT NULL ,

        [InventoryItemID]   integer  NOT NULL ,

        [BuyID]             char(18)  NULL ,

        [SellID]            integer  NULL

)

go
```

```
ALTER TABLE [Inventory]

        ADD CONSTRAINT [XPKInventory] PRIMARY KEY  CLUSTERED ([InventoryItemID]
ASC,[AccountID] ASC,[ContractID] ASC)

go



CREATE TABLE [Sell]

(

        [AccountID]          integer  NOT NULL ,

        [ContractID]         integer  NOT NULL ,

        [SellID]             integer  NOT NULL ,

        [Rate]               char(18)  NULL ,

        [Quantity]           char(18)  NULL ,

        [SellDate]           char(18)  NULL ,

        [BrokerID]           integer  NULL ,

        [TaxID]              integer  NULL ,

        [Brokrage]           char(18)  NULL ,

        [Tax]                char(18)  NULL ,

        [Total]              char(18)  NULL

)

go



ALTER TABLE [Sell]

        ADD CONSTRAINT [XPKAccount_Contract] PRIMARY KEY  CLUSTERED ([AccountID]
ASC,[ContractID] ASC,[SellID] ASC)

go
```

```
CREATE TABLE [Taxes]

(

        [TaxID]              integer  NOT NULL ,

        [TaxRate]            char(18)  NULL ,

        [TaxName]            char(18)  NULL

)

go



ALTER TABLE [Taxes]

        ADD CONSTRAINT [XPKTax] PRIMARY KEY  CLUSTERED ([TaxID] ASC)

go




ALTER TABLE [Buy]

        ADD CONSTRAINT [R_6] FOREIGN KEY ([AccountID]) REFERENCES
[Account]([AccountID])

                ON DELETE NO ACTION

                ON UPDATE NO ACTION

go



ALTER TABLE [Buy]

        ADD CONSTRAINT [R_7] FOREIGN KEY ([ContractID]) REFERENCES
[Contract]([ContractID])

                ON DELETE NO ACTION
```

ON UPDATE NO ACTION

go

ALTER TABLE [Buy]

        ADD CONSTRAINT [R_13] FOREIGN KEY ([BrokerID]) REFERENCES
[Brokrage]([BrokerID])

                ON DELETE NO ACTION

                ON UPDATE NO ACTION

go

ALTER TABLE [Buy]

        ADD CONSTRAINT [R_16] FOREIGN KEY ([TaxID]) REFERENCES [Taxes]([TaxID])

                ON DELETE NO ACTION

                ON UPDATE NO ACTION

go

ALTER TABLE [IntraDay]

        ADD CONSTRAINT [R_11] FOREIGN KEY ([AccountID]) REFERENCES
[Account]([AccountID])

                ON DELETE NO ACTION

                ON UPDATE NO ACTION

go

ALTER TABLE [IntraDay]

```
        ADD CONSTRAINT [R_12] FOREIGN KEY ([ContractID]) REFERENCES
[Contract]([ContractID])

                ON DELETE NO ACTION

                ON UPDATE NO ACTION

go



ALTER TABLE [IntraDay]

        ADD CONSTRAINT [R_15] FOREIGN KEY ([BrokerID]) REFERENCES
[Brokrage]([BrokerID])

                ON DELETE NO ACTION

                ON UPDATE NO ACTION

go



ALTER TABLE [IntraDay]

        ADD CONSTRAINT [R_18] FOREIGN KEY ([TaxID]) REFERENCES [Taxes]([TaxID])

                ON DELETE NO ACTION

                ON UPDATE NO ACTION

go



ALTER TABLE [Inventory]

        ADD CONSTRAINT [R_19] FOREIGN KEY ([AccountID]) REFERENCES
[Account]([AccountID])

                ON DELETE NO ACTION

                ON UPDATE NO ACTION

go
```

ALTER TABLE [Inventory]

      ADD CONSTRAINT [R_20] FOREIGN KEY ([ContractID]) REFERENCES [Contract]([ContractID])

           ON DELETE NO ACTION

           ON UPDATE NO ACTION

go

ALTER TABLE [Inventory]

      ADD CONSTRAINT [R_21] FOREIGN KEY ([BuyID],[AccountID],[ContractID]) REFERENCES [Buy]([BuyID],[AccountID],[ContractID])

           ON DELETE NO ACTION

           ON UPDATE NO ACTION

go

ALTER TABLE [Inventory]

      ADD CONSTRAINT [R_22] FOREIGN KEY ([AccountID],[ContractID],[SellID]) REFERENCES [Sell]([AccountID],[ContractID],[SellID])

           ON DELETE NO ACTION

           ON UPDATE NO ACTION

go

ALTER TABLE [Sell]

      ADD CONSTRAINT [R_3] FOREIGN KEY ([AccountID]) REFERENCES [Account]([AccountID])

```
        ON DELETE NO ACTION

        ON UPDATE NO ACTION

go



ALTER TABLE [Sell]

    ADD CONSTRAINT [R_4] FOREIGN KEY ([ContractID]) REFERENCES
[Contract]([ContractID])

        ON DELETE NO ACTION

        ON UPDATE NO ACTION

go



ALTER TABLE [Sell]

    ADD CONSTRAINT [R_14] FOREIGN KEY ([BrokerID]) REFERENCES
[Brokrage]([BrokerID])

        ON DELETE NO ACTION

        ON UPDATE NO ACTION

go



ALTER TABLE [Sell]

    ADD CONSTRAINT [R_17] FOREIGN KEY ([TaxID]) REFERENCES [Taxes]([TaxID])

        ON DELETE NO ACTION

        ON UPDATE NO ACTION

go
```

```
CREATE TRIGGER tD_Account ON Account FOR DELETE AS

/* erwin Builtin Trigger */

/* DELETE trigger on Account */

BEGIN

  DECLARE  @errno   int,

           @severity int,

           @state    int,

           @errmsg  varchar(255)

    /* erwin Builtin Trigger */

    /* Account   Inventory on parent delete no action */

    /* ERWIN_RELATION:CHECKSUM="0003a30d", PARENT_OWNER="", PARENT_TABLE="Account"

    CHILD_OWNER="", CHILD_TABLE="Inventory"

    P2C_VERB_PHRASE="", C2P_VERB_PHRASE="",

    FK_CONSTRAINT="R_19", FK_COLUMNS="AccountID" */

    IF EXISTS (

      SELECT * FROM deleted,Inventory

      WHERE

        /*  %JoinFKPK(Inventory,deleted," = "," AND") */

        Inventory.AccountID = deleted.AccountID

    )

    BEGIN

      SELECT @errno  = 30001,

             @errmsg = 'Cannot delete Account because Inventory exists.'

      GOTO error
```

END


```
/* erwin Builtin Trigger */

/* Account  IntraDay on parent delete no action */

/* ERWIN_RELATION:CHECKSUM="00000000", PARENT_OWNER="", PARENT_TABLE="Account"

CHILD_OWNER="", CHILD_TABLE="IntraDay"

P2C_VERB_PHRASE="", C2P_VERB_PHRASE="",

FK_CONSTRAINT="R_11", FK_COLUMNS="AccountID" */

IF EXISTS (

  SELECT * FROM deleted,IntraDay

  WHERE

    /*  %JoinFKPK(IntraDay,deleted," = "," AND") */

    IntraDay.AccountID = deleted.AccountID

)

BEGIN

  SELECT @errno  = 30001,

         @errmsg = 'Cannot delete Account because IntraDay exists.'

  GOTO error

END


/* erwin Builtin Trigger */

/* Account  Buy on parent delete no action */

/* ERWIN_RELATION:CHECKSUM="00000000", PARENT_OWNER="", PARENT_TABLE="Account"

CHILD_OWNER="", CHILD_TABLE="Buy"
```

```
P2C_VERB_PHRASE="", C2P_VERB_PHRASE="",

FK_CONSTRAINT="R_6", FK_COLUMNS="AccountID" */

IF EXISTS (

  SELECT * FROM deleted,Buy

  WHERE

    /*  %JoinFKPK(Buy,deleted," = "," AND") */

    Buy.AccountID = deleted.AccountID

)

BEGIN

  SELECT @errno  = 30001,

         @errmsg = 'Cannot delete Account because Buy exists.'

  GOTO error

END


/* erwin Builtin Trigger */

/* Account  Sell on parent delete no action */

/* ERWIN_RELATION:CHECKSUM="00000000", PARENT_OWNER="", PARENT_TABLE="Account"

CHILD_OWNER="", CHILD_TABLE="Sell"

P2C_VERB_PHRASE="", C2P_VERB_PHRASE="",

FK_CONSTRAINT="R_3", FK_COLUMNS="AccountID" */

IF EXISTS (

  SELECT * FROM deleted,Sell

  WHERE

    /*  %JoinFKPK(Sell,deleted," = "," AND") */
```

```
        Sell.AccountID = deleted.AccountID

    )

    BEGIN

      SELECT @errno  = 30001,

             @errmsg = 'Cannot delete Account because Sell exists.'

      GOTO error

    END



    /* erwin Builtin Trigger */

    RETURN

error:

    RAISERROR (@errmsg, -- Message text.

               @severity, -- Severity (0~25).

               @state) -- State (0~255).

    rollback transaction

END



go



CREATE TRIGGER tU_Account ON Account FOR UPDATE AS

/* erwin Builtin Trigger */

/* UPDATE trigger on Account */
```

```
BEGIN

  DECLARE  @numrows int,

           @nullcnt int,

           @validcnt int,

           @insAccountID integer,

           @errno   int,

           @severity int,

           @state    int,

           @errmsg  varchar(255)



  SELECT @numrows = @@rowcount

  /* erwin Builtin Trigger */

  /* Account  Inventory on parent update no action */

  /* ERWIN_RELATION:CHECKSUM="000427fa", PARENT_OWNER="", PARENT_TABLE="Account"

    CHILD_OWNER="", CHILD_TABLE="Inventory"

    P2C_VERB_PHRASE="", C2P_VERB_PHRASE="",

    FK_CONSTRAINT="R_19", FK_COLUMNS="AccountID" */

  IF

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

    UPDATE(AccountID)

  BEGIN

    IF EXISTS (

      SELECT * FROM deleted,Inventory

      WHERE
```

```
    /*  %JoinFKPK(Inventory,deleted," = "," AND") */

    Inventory.AccountID = deleted.AccountID

 )

 BEGIN

   SELECT @errno  = 30005,

          @errmsg = 'Cannot update Account because Inventory exists.'

   GOTO error

 END

END


/* erwin Builtin Trigger */

/* Account  IntraDay on parent update no action */

/* ERWIN_RELATION:CHECKSUM="00000000", PARENT_OWNER="", PARENT_TABLE="Account"

 CHILD_OWNER="", CHILD_TABLE="IntraDay"

 P2C_VERB_PHRASE="", C2P_VERB_PHRASE="",

 FK_CONSTRAINT="R_11", FK_COLUMNS="AccountID" */

IF

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

 UPDATE(AccountID)

BEGIN

 IF EXISTS (

   SELECT * FROM deleted,IntraDay

   WHERE

     /*  %JoinFKPK(IntraDay,deleted," = "," AND") */
```

```
        IntraDay.AccountID = deleted.AccountID

    )

  BEGIN

    SELECT @errno  = 30005,

           @errmsg = 'Cannot update Account because IntraDay exists.'

    GOTO error

  END

END


/* erwin Builtin Trigger */

/* Account  Buy on parent update no action */

/* ERWIN_RELATION:CHECKSUM="00000000", PARENT_OWNER="", PARENT_TABLE="Account"

  CHILD_OWNER="", CHILD_TABLE="Buy"

  P2C_VERB_PHRASE="", C2P_VERB_PHRASE="",

  FK_CONSTRAINT="R_6", FK_COLUMNS="AccountID" */

IF

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

  UPDATE(AccountID)

BEGIN

  IF EXISTS (

    SELECT * FROM deleted,Buy

    WHERE

      /*  %JoinFKPK(Buy,deleted," = "," AND") */

      Buy.AccountID = deleted.AccountID
```

```
  )

  BEGIN

    SELECT @errno  = 30005,

           @errmsg = 'Cannot update Account because Buy exists.'

    GOTO error

  END

END


/* erwin Builtin Trigger */

/* Account  Sell on parent update no action */

/* ERWIN_RELATION:CHECKSUM="00000000", PARENT_OWNER="", PARENT_TABLE="Account"

  CHILD_OWNER="", CHILD_TABLE="Sell"

  P2C_VERB_PHRASE="", C2P_VERB_PHRASE="",

  FK_CONSTRAINT="R_3", FK_COLUMNS="AccountID" */

IF

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

  UPDATE(AccountID)

BEGIN

  IF EXISTS (

    SELECT * FROM deleted,Sell

    WHERE

      /*  %JoinFKPK(Sell,deleted," = "," AND") */

      Sell.AccountID = deleted.AccountID

  )
```

```
    BEGIN

      SELECT @errno  = 30005,

             @errmsg = 'Cannot update Account because Sell exists.'

        GOTO error

    END

  END



  /* erwin Builtin Trigger */

  RETURN

error:

   RAISERROR (@errmsg, -- Message text.

              @severity, -- Severity (0~25).

              @state) -- State (0~255).

    rollback transaction

END



go




CREATE TRIGGER tD_Brokrage ON Brokrage FOR DELETE AS

/* erwin Builtin Trigger */
```

```
/* DELETE trigger on Brokrage */

BEGIN

  DECLARE  @errno   int,

           @severity int,

           @state    int,

           @errmsg  varchar(255)

    /* erwin Builtin Trigger */

    /* Brokrage  IntraDay on parent delete no action */

    /* ERWIN_RELATION:CHECKSUM="0002caa7", PARENT_OWNER="", PARENT_TABLE="Brokrage"

    CHILD_OWNER="", CHILD_TABLE="IntraDay"

    P2C_VERB_PHRASE="", C2P_VERB_PHRASE="",

    FK_CONSTRAINT="R_15", FK_COLUMNS="BrokerID" */

    IF EXISTS (

      SELECT * FROM deleted,IntraDay

      WHERE

        /*  %JoinFKPK(IntraDay,deleted," = "," AND") */

        IntraDay.BrokerID = deleted.BrokerID

    )

    BEGIN

      SELECT @errno  = 30001,

             @errmsg = 'Cannot delete Brokrage because IntraDay exists.'

      GOTO error

    END
```

```
/* erwin Builtin Trigger */

/* Brokrage   Sell on parent delete no action */

/* ERWIN_RELATION:CHECKSUM="00000000", PARENT_OWNER="", PARENT_TABLE="Brokrage"

CHILD_OWNER="", CHILD_TABLE="Sell"

P2C_VERB_PHRASE="", C2P_VERB_PHRASE="",

FK_CONSTRAINT="R_14", FK_COLUMNS="BrokerID" */

IF EXISTS (

  SELECT * FROM deleted,Sell

  WHERE

    /*  %JoinFKPK(Sell,deleted," = "," AND") */

    Sell.BrokerID = deleted.BrokerID

)

BEGIN

  SELECT @errno  = 30001,

        @errmsg = 'Cannot delete Brokrage because Sell exists.'

  GOTO error

END


/* erwin Builtin Trigger */

/* Brokrage   Buy on parent delete no action */

/* ERWIN_RELATION:CHECKSUM="00000000", PARENT_OWNER="", PARENT_TABLE="Brokrage"

CHILD_OWNER="", CHILD_TABLE="Buy"

P2C_VERB_PHRASE="", C2P_VERB_PHRASE="",

FK_CONSTRAINT="R_13", FK_COLUMNS="BrokerID" */
```

```
  IF EXISTS (

    SELECT * FROM deleted,Buy

    WHERE

      /*  %JoinFKPK(Buy,deleted," = "," AND") */

      Buy.BrokerID = deleted.BrokerID

  )

  BEGIN

    SELECT @errno  = 30001,

           @errmsg = 'Cannot delete Brokrage because Buy exists.'

    GOTO error

  END



  /* erwin Builtin Trigger */

  RETURN
error:

  RAISERROR (@errmsg, -- Message text.

             @severity, -- Severity (0~25).

             @state) -- State (0~255).

  rollback transaction

END


go
```

```
CREATE TRIGGER tU_Brokrage ON Brokrage FOR UPDATE AS

/* erwin Builtin Trigger */

/* UPDATE trigger on Brokrage */

BEGIN

  DECLARE  @numrows int,

           @nullcnt int,

           @validcnt int,

           @insBrokerID integer,

           @errno   int,

           @severity int,

           @state    int,

           @errmsg  varchar(255)


  SELECT @numrows = @@rowcount

  /* erwin Builtin Trigger */

  /* Brokrage  IntraDay on parent update no action */

  /* ERWIN_RELATION:CHECKSUM="00031443", PARENT_OWNER="", PARENT_TABLE="Brokrage"

    CHILD_OWNER="", CHILD_TABLE="IntraDay"

    P2C_VERB_PHRASE="", C2P_VERB_PHRASE="",

    FK_CONSTRAINT="R_15", FK_COLUMNS="BrokerID" */

  IF

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

    UPDATE(BrokerID)
```

```
BEGIN

  IF EXISTS (

    SELECT * FROM deleted,IntraDay

    WHERE

      /*  %JoinFKPK(IntraDay,deleted," = "," AND") */

      IntraDay.BrokerID = deleted.BrokerID

  )

  BEGIN

    SELECT @errno  = 30005,

           @errmsg = 'Cannot update Brokrage because IntraDay exists.'

    GOTO error

  END

END


/* erwin Builtin Trigger */

/* Brokrage  Sell on parent update no action */

/* ERWIN_RELATION:CHECKSUM="00000000", PARENT_OWNER="", PARENT_TABLE="Brokrage"

  CHILD_OWNER="", CHILD_TABLE="Sell"

  P2C_VERB_PHRASE="", C2P_VERB_PHRASE="",

  FK_CONSTRAINT="R_14", FK_COLUMNS="BrokerID" */

IF

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

  UPDATE(BrokerID)

BEGIN
```

```
  IF EXISTS (

    SELECT * FROM deleted,Sell

    WHERE

      /*  %JoinFKPK(Sell,deleted," = "," AND") */

      Sell.BrokerID = deleted.BrokerID

  )

  BEGIN

    SELECT @errno  = 30005,

           @errmsg = 'Cannot update Brokrage because Sell exists.'

    GOTO error

  END

END


/* erwin Builtin Trigger */

/* Brokrage  Buy on parent update no action */

/* ERWIN_RELATION:CHECKSUM="00000000", PARENT_OWNER="", PARENT_TABLE="Brokrage"

  CHILD_OWNER="", CHILD_TABLE="Buy"

  P2C_VERB_PHRASE="", C2P_VERB_PHRASE="",

  FK_CONSTRAINT="R_13", FK_COLUMNS="BrokerID" */

IF

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

  UPDATE(BrokerID)

BEGIN

  IF EXISTS (
```

```
    SELECT * FROM deleted,Buy

    WHERE

      /*  %JoinFKPK(Buy,deleted," = "," AND") */

      Buy.BrokerID = deleted.BrokerID

    )

  BEGIN

    SELECT @errno  = 30005,

            @errmsg = 'Cannot update Brokrage because Buy exists.'

    GOTO error

  END

END


  /* erwin Builtin Trigger */

  RETURN

error:

   RAISERROR (@errmsg, -- Message text.

              @severity, -- Severity (0~25).

              @state) -- State (0~255).

    rollback transaction

END


go
```

```
CREATE TRIGGER tD_Buy ON Buy FOR DELETE AS

/* erwin Builtin Trigger */

/* DELETE trigger on Buy */

BEGIN

  DECLARE  @errno    int,

           @severity int,

           @state    int,

           @errmsg  varchar(255)

    /* erwin Builtin Trigger */

    /* Buy   Inventory on parent delete no action */

    /* ERWIN_RELATION:CHECKSUM="00056c45", PARENT_OWNER="", PARENT_TABLE="Buy"

    CHILD_OWNER="", CHILD_TABLE="Inventory"

    P2C_VERB_PHRASE="", C2P_VERB_PHRASE="",

    FK_CONSTRAINT="R_21", FK_COLUMNS="AccountID""ContractID""BuyID" */

    IF EXISTS (

      SELECT * FROM deleted,Inventory

      WHERE

        /*  %JoinFKPK(Inventory,deleted," = "," AND") */

        Inventory.AccountID = deleted.AccountID AND

        Inventory.ContractID = deleted.ContractID AND

        Inventory.BuyID = deleted.BuyID
```

```
)

BEGIN

  SELECT @errno  = 30001,

         @errmsg = 'Cannot delete Buy because Inventory exists.'

  GOTO error

END


/* erwin Builtin Trigger */

/* Taxes  Buy on child delete no action */

/* ERWIN_RELATION:CHECKSUM="00000000", PARENT_OWNER="", PARENT_TABLE="Taxes"

CHILD_OWNER="", CHILD_TABLE="Buy"

P2C_VERB_PHRASE="", C2P_VERB_PHRASE="",

FK_CONSTRAINT="R_16", FK_COLUMNS="TaxID" */

IF EXISTS (SELECT * FROM deleted,Taxes

  WHERE

    /* %JoinFKPK(deleted,Taxes," = "," AND") */

    deleted.TaxID = Taxes.TaxID AND

    NOT EXISTS (

      SELECT * FROM Buy

      WHERE

        /* %JoinFKPK(Buy,Taxes," = "," AND") */

        Buy.TaxID = Taxes.TaxID

    )

)
```

```
BEGIN

  SELECT @errno  = 30010,

        @errmsg = 'Cannot delete last Buy because Taxes exists.'

  GOTO error

END


/* erwin Builtin Trigger */

/* Brokrage  Buy on child delete no action */

/* ERWIN_RELATION:CHECKSUM="00000000", PARENT_OWNER="", PARENT_TABLE="Brokrage"

CHILD_OWNER="", CHILD_TABLE="Buy"

P2C_VERB_PHRASE="", C2P_VERB_PHRASE="",

FK_CONSTRAINT="R_13", FK_COLUMNS="BrokerID" */

IF EXISTS (SELECT * FROM deleted,Brokrage

  WHERE

    /* %JoinFKPK(deleted,Brokrage," = "," AND") */

    deleted.BrokerID = Brokrage.BrokerID AND

    NOT EXISTS (

      SELECT * FROM Buy

      WHERE

        /* %JoinFKPK(Buy,Brokrage," = "," AND") */

        Buy.BrokerID = Brokrage.BrokerID

    )

)

BEGIN
```

```
   SELECT @errno  = 30010,

          @errmsg = 'Cannot delete last Buy because Brokrage exists.'

   GOTO error

END


/* erwin Builtin Trigger */

/* Contract  Buy on child delete no action */

/* ERWIN_RELATION:CHECKSUM="00000000", PARENT_OWNER="", PARENT_TABLE="Contract"

CHILD_OWNER="", CHILD_TABLE="Buy"

P2C_VERB_PHRASE="", C2P_VERB_PHRASE="",

FK_CONSTRAINT="R_7", FK_COLUMNS="ContractID" */

IF EXISTS (SELECT * FROM deleted,Contract

   WHERE

     /* %JoinFKPK(deleted,Contract," = "," AND") */

     deleted.ContractID = Contract.ContractID AND

     NOT EXISTS (

       SELECT * FROM Buy

       WHERE

         /* %JoinFKPK(Buy,Contract," = "," AND") */

         Buy.ContractID = Contract.ContractID

     )

)

BEGIN

   SELECT @errno  = 30010,
```

```
        @errmsg = 'Cannot delete last Buy because Contract exists.'

  GOTO error

END


/* erwin Builtin Trigger */

/* Account  Buy on child delete no action */

/* ERWIN_RELATION:CHECKSUM="00000000", PARENT_OWNER="", PARENT_TABLE="Account"

CHILD_OWNER="", CHILD_TABLE="Buy"

P2C_VERB_PHRASE="", C2P_VERB_PHRASE="",

FK_CONSTRAINT="R_6", FK_COLUMNS="AccountID" */

IF EXISTS (SELECT * FROM deleted,Account

  WHERE

    /* %JoinFKPK(deleted,Account," = "," AND") */

    deleted.AccountID = Account.AccountID AND

    NOT EXISTS (

      SELECT * FROM Buy

      WHERE

        /* %JoinFKPK(Buy,Account," = "," AND") */

        Buy.AccountID = Account.AccountID

    )

)

BEGIN

  SELECT @errno  = 30010,

        @errmsg = 'Cannot delete last Buy because Account exists.'
```

```
    GOTO error

  END



    /* erwin Builtin Trigger */

    RETURN

error:

   RAISERROR (@errmsg, -- Message text.

              @severity, -- Severity (0~25).

              @state) -- State (0~255).

    rollback transaction

END



go



CREATE TRIGGER tU_Buy ON Buy FOR UPDATE AS

/* erwin Builtin Trigger */

/* UPDATE trigger on Buy */

BEGIN

  DECLARE  @numrows int,

           @nullcnt int,

           @validcnt int,

           @insAccountID integer,
```

```
        @insContractID integer,

        @insBuyID char(18),

        @errno    int,

        @severity int,

        @state    int,

        @errmsg  varchar(255)


SELECT @numrows = @@rowcount

/* erwin Builtin Trigger */

/* Buy   Inventory on parent update no action */

/* ERWIN_RELATION:CHECKSUM="00069888", PARENT_OWNER="", PARENT_TABLE="Buy"

  CHILD_OWNER="", CHILD_TABLE="Inventory"

  P2C_VERB_PHRASE="", C2P_VERB_PHRASE="",

  FK_CONSTRAINT="R_21", FK_COLUMNS="AccountID""ContractID""BuyID" */

IF

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

  UPDATE(AccountID) OR

  UPDATE(ContractID) OR

  UPDATE(BuyID)

BEGIN

  IF EXISTS (

    SELECT * FROM deleted,Inventory

    WHERE

      /*  %JoinFKPK(Inventory,deleted," = "," AND") */
```

```
      Inventory.AccountID = deleted.AccountID AND

      Inventory.ContractID = deleted.ContractID AND

      Inventory.BuyID = deleted.BuyID

  )

  BEGIN

    SELECT @errno  = 30005,

           @errmsg = 'Cannot update Buy because Inventory exists.'

    GOTO error

  END

END


/* erwin Builtin Trigger */

/* Taxes  Buy on child update no action */

/* ERWIN_RELATION:CHECKSUM="00000000", PARENT_OWNER="", PARENT_TABLE="Taxes"

  CHILD_OWNER="", CHILD_TABLE="Buy"

  P2C_VERB_PHRASE="", C2P_VERB_PHRASE="",

  FK_CONSTRAINT="R_16", FK_COLUMNS="TaxID" */

IF

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

  UPDATE(TaxID)

BEGIN

  SELECT @nullcnt = 0

  SELECT @validcnt = count(*)

    FROM inserted,Taxes
```

```
    WHERE

        /* %JoinFKPK(inserted,Taxes) */

        inserted.TaxID = Taxes.TaxID

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

  select @nullcnt = count(*) from inserted where

    inserted.TaxID IS NULL

  IF @validcnt + @nullcnt != @numrows

  BEGIN

    SELECT @errno  = 30007,

          @errmsg = 'Cannot update Buy because Taxes does not exist.'

    GOTO error

  END

END


/* erwin Builtin Trigger */

/* Brokrage  Buy on child update no action */

/* ERWIN_RELATION:CHECKSUM="00000000", PARENT_OWNER="", PARENT_TABLE="Brokrage"

  CHILD_OWNER="", CHILD_TABLE="Buy"

  P2C_VERB_PHRASE="", C2P_VERB_PHRASE="",

  FK_CONSTRAINT="R_13", FK_COLUMNS="BrokerID" */

IF

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

  UPDATE(BrokerID)
```

```
BEGIN

  SELECT @nullcnt = 0

  SELECT @validcnt = count(*)

    FROM inserted,Brokrage

      WHERE

        /* %JoinFKPK(inserted,Brokrage) */

        inserted.BrokerID = Brokrage.BrokerID

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

  select @nullcnt = count(*) from inserted where

    inserted.BrokerID IS NULL

  IF @validcnt + @nullcnt != @numrows

  BEGIN

    SELECT @errno  = 30007,

          @errmsg = 'Cannot update Buy because Brokrage does not exist.'

    GOTO error

  END

END


/* erwin Builtin Trigger */

/* Contract  Buy on child update no action */

/* ERWIN_RELATION:CHECKSUM="00000000", PARENT_OWNER="", PARENT_TABLE="Contract"

  CHILD_OWNER="", CHILD_TABLE="Buy"

  P2C_VERB_PHRASE="", C2P_VERB_PHRASE="",

  FK_CONSTRAINT="R_7", FK_COLUMNS="ContractID" */
```

```
IF

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

  UPDATE(ContractID)

BEGIN

  SELECT @nullcnt = 0

  SELECT @validcnt = count(*)

    FROM inserted,Contract

      WHERE

        /* %JoinFKPK(inserted,Contract) */

        inserted.ContractID = Contract.ContractID

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


  IF @validcnt + @nullcnt != @numrows

  BEGIN

    SELECT @errno  = 30007,

          @errmsg = 'Cannot update Buy because Contract does not exist.'

    GOTO error

  END

END


/* erwin Builtin Trigger */

/* Account  Buy on child update no action */

/* ERWIN_RELATION:CHECKSUM="00000000", PARENT_OWNER="", PARENT_TABLE="Account"
```

```
  CHILD_OWNER="", CHILD_TABLE="Buy"

  P2C_VERB_PHRASE="", C2P_VERB_PHRASE="",

  FK_CONSTRAINT="R_6", FK_COLUMNS="AccountID" */

IF

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

  UPDATE(AccountID)

BEGIN

  SELECT @nullcnt = 0

  SELECT @validcnt = count(*)

    FROM inserted,Account

      WHERE

        /* %JoinFKPK(inserted,Account) */

        inserted.AccountID = Account.AccountID

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


  IF @validcnt + @nullcnt != @numrows

  BEGIN

    SELECT @errno  = 30007,

          @errmsg = 'Cannot update Buy because Account does not exist.'

    GOTO error

  END

END
```

```
  /* erwin Builtin Trigger */

  RETURN

error:

  RAISERROR (@errmsg, -- Message text.

             @severity, -- Severity (0~25).

             @state) -- State (0~255).

    rollback transaction

END



go




CREATE TRIGGER tD_Contract ON Contract FOR DELETE AS

/* erwin Builtin Trigger */

/* DELETE trigger on Contract */

BEGIN

  DECLARE  @errno   int,

           @severity int,

           @state    int,

           @errmsg  varchar(255)

    /* erwin Builtin Trigger */

    /* Contract  Inventory on parent delete no action */
```

```
/* ERWIN_RELATION:CHECKSUM="0003b585", PARENT_OWNER="", PARENT_TABLE="Contract"

CHILD_OWNER="", CHILD_TABLE="Inventory"

P2C_VERB_PHRASE="", C2P_VERB_PHRASE="",

FK_CONSTRAINT="R_20", FK_COLUMNS="ContractID" */

IF EXISTS (

  SELECT * FROM deleted,Inventory

  WHERE

    /*  %JoinFKPK(Inventory,deleted," = "," AND") */

    Inventory.ContractID = deleted.ContractID

)

BEGIN

  SELECT @errno  = 30001,

         @errmsg = 'Cannot delete Contract because Inventory exists.'

  GOTO error

END


/* erwin Builtin Trigger */

/* Contract  IntraDay on parent delete no action */

/* ERWIN_RELATION:CHECKSUM="00000000", PARENT_OWNER="", PARENT_TABLE="Contract"

CHILD_OWNER="", CHILD_TABLE="IntraDay"

P2C_VERB_PHRASE="", C2P_VERB_PHRASE="",

FK_CONSTRAINT="R_12", FK_COLUMNS="ContractID" */

IF EXISTS (

  SELECT * FROM deleted,IntraDay
```

```
   WHERE

     /*  %JoinFKPK(IntraDay,deleted," = "," AND") */

     IntraDay.ContractID = deleted.ContractID

)

BEGIN

   SELECT @errno  = 30001,

          @errmsg = 'Cannot delete Contract because IntraDay exists.'

   GOTO error

END


/* erwin Builtin Trigger */

/* Contract  Buy on parent delete no action */

/* ERWIN_RELATION:CHECKSUM="00000000", PARENT_OWNER="", PARENT_TABLE="Contract"

CHILD_OWNER="", CHILD_TABLE="Buy"

P2C_VERB_PHRASE="", C2P_VERB_PHRASE="",

FK_CONSTRAINT="R_7", FK_COLUMNS="ContractID" */

IF EXISTS (

   SELECT * FROM deleted,Buy

   WHERE

     /*  %JoinFKPK(Buy,deleted," = "," AND") */

     Buy.ContractID = deleted.ContractID

)

BEGIN

   SELECT @errno  = 30001,
```

```
        @errmsg = 'Cannot delete Contract because Buy exists.'

  GOTO error

END



/* erwin Builtin Trigger */

/* Contract  Sell on parent delete no action */

/* ERWIN_RELATION:CHECKSUM="00000000", PARENT_OWNER="", PARENT_TABLE="Contract"

CHILD_OWNER="", CHILD_TABLE="Sell"

P2C_VERB_PHRASE="", C2P_VERB_PHRASE="",

FK_CONSTRAINT="R_4", FK_COLUMNS="ContractID" */

IF EXISTS (

  SELECT * FROM deleted,Sell

  WHERE

    /*  %JoinFKPK(Sell,deleted," = "," AND") */

    Sell.ContractID = deleted.ContractID

)

BEGIN

  SELECT @errno  = 30001,

        @errmsg = 'Cannot delete Contract because Sell exists.'

  GOTO error

END



/* erwin Builtin Trigger */
```

```
    RETURN

error:

    RAISERROR (@errmsg, -- Message text.

               @severity, -- Severity (0~25).

               @state) -- State (0~255).

    rollback transaction

END



go



CREATE TRIGGER tU_Contract ON Contract FOR UPDATE AS

/* erwin Builtin Trigger */

/* UPDATE trigger on Contract */

BEGIN

  DECLARE  @numrows int,

           @nullcnt int,

           @validcnt int,

           @insContractID integer,

           @errno    int,

           @severity int,

           @state    int,

           @errmsg  varchar(255)
```

```
SELECT @numrows = @@rowcount

/* erwin Builtin Trigger */

/* Contract  Inventory on parent update no action */

/* ERWIN_RELATION:CHECKSUM="00040461", PARENT_OWNER="", PARENT_TABLE="Contract"

  CHILD_OWNER="", CHILD_TABLE="Inventory"

  P2C_VERB_PHRASE="", C2P_VERB_PHRASE="",

  FK_CONSTRAINT="R_20", FK_COLUMNS="ContractID" */

IF

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

  UPDATE(ContractID)

BEGIN

  IF EXISTS (

    SELECT * FROM deleted,Inventory

    WHERE

      /*  %JoinFKPK(Inventory,deleted," = "," AND") */

      Inventory.ContractID = deleted.ContractID

  )

  BEGIN

    SELECT @errno  = 30005,

          @errmsg = 'Cannot update Contract because Inventory exists.'

    GOTO error

  END

END
```

```
/* erwin Builtin Trigger */

/* Contract  IntraDay on parent update no action */

/* ERWIN_RELATION:CHECKSUM="00000000", PARENT_OWNER="", PARENT_TABLE="Contract"

  CHILD_OWNER="", CHILD_TABLE="IntraDay"

  P2C_VERB_PHRASE="", C2P_VERB_PHRASE="",

  FK_CONSTRAINT="R_12", FK_COLUMNS="ContractID" */

IF

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

  UPDATE(ContractID)

BEGIN

  IF EXISTS (

    SELECT * FROM deleted,IntraDay

    WHERE

      /*  %JoinFKPK(IntraDay,deleted," = "," AND") */

      IntraDay.ContractID = deleted.ContractID

  )

  BEGIN

    SELECT @errno  = 30005,

          @errmsg = 'Cannot update Contract because IntraDay exists.'

    GOTO error

  END

END


/* erwin Builtin Trigger */
```

```
/* Contract  Buy on parent update no action */

/* ERWIN_RELATION:CHECKSUM="00000000", PARENT_OWNER="", PARENT_TABLE="Contract"

  CHILD_OWNER="", CHILD_TABLE="Buy"

  P2C_VERB_PHRASE="", C2P_VERB_PHRASE="",

  FK_CONSTRAINT="R_7", FK_COLUMNS="ContractID" */

IF

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

  UPDATE(ContractID)

BEGIN

  IF EXISTS (

    SELECT * FROM deleted,Buy

    WHERE

      /*  %JoinFKPK(Buy,deleted," = "," AND") */

      Buy.ContractID = deleted.ContractID

  )

  BEGIN

    SELECT @errno  = 30005,

           @errmsg = 'Cannot update Contract because Buy exists.'

    GOTO error

  END

END


/* erwin Builtin Trigger */

/* Contract  Sell on parent update no action */
```

```
/* ERWIN_RELATION:CHECKSUM="00000000", PARENT_OWNER="", PARENT_TABLE="Contract"

  CHILD_OWNER="", CHILD_TABLE="Sell"

  P2C_VERB_PHRASE="", C2P_VERB_PHRASE="",

  FK_CONSTRAINT="R_4", FK_COLUMNS="ContractID" */

IF

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

  UPDATE(ContractID)

BEGIN

  IF EXISTS (

    SELECT * FROM deleted,Sell

    WHERE

      /*  %JoinFKPK(Sell,deleted," = "," AND") */

      Sell.ContractID = deleted.ContractID

  )

  BEGIN

    SELECT @errno  = 30005,

          @errmsg = 'Cannot update Contract because Sell exists.'

    GOTO error

  END

END



/* erwin Builtin Trigger */

RETURN
```

error:

    RAISERROR (@errmsg, -- Message text.

                @severity, -- Severity (0~25).

                @state) -- State (0~255).

      rollback transaction

END

go

CREATE TRIGGER tD_IntraDay ON IntraDay FOR DELETE AS

/* erwin Builtin Trigger */

/* DELETE trigger on IntraDay */

BEGIN

  DECLARE @errno   int,

           @severity int,

           @state   int,

           @errmsg varchar(255)

    /* erwin Builtin Trigger */

    /* Taxes   IntraDay on child delete no action */

    /* ERWIN_RELATION:CHECKSUM="0004a2b0", PARENT_OWNER="", PARENT_TABLE="Taxes"

    CHILD_OWNER="", CHILD_TABLE="IntraDay"

```
P2C_VERB_PHRASE="", C2P_VERB_PHRASE="",

FK_CONSTRAINT="R_18", FK_COLUMNS="TaxID" */

IF EXISTS (SELECT * FROM deleted,Taxes

  WHERE

    /* %JoinFKPK(deleted,Taxes," = "," AND") */

    deleted.TaxID = Taxes.TaxID AND

    NOT EXISTS (

      SELECT * FROM IntraDay

      WHERE

        /* %JoinFKPK(IntraDay,Taxes," = "," AND") */

        IntraDay.TaxID = Taxes.TaxID

    )

)

BEGIN

  SELECT @errno  = 30010,

         @errmsg = 'Cannot delete last IntraDay because Taxes exists.'

  GOTO error

END


/* erwin Builtin Trigger */

/* Brokrage  IntraDay on child delete no action */

/* ERWIN_RELATION:CHECKSUM="00000000", PARENT_OWNER="", PARENT_TABLE="Brokrage"

CHILD_OWNER="", CHILD_TABLE="IntraDay"

P2C_VERB_PHRASE="", C2P_VERB_PHRASE="",
```

```
FK_CONSTRAINT="R_15", FK_COLUMNS="BrokerID" */

IF EXISTS (SELECT * FROM deleted,Brokrage

   WHERE

      /* %JoinFKPK(deleted,Brokrage," = "," AND") */

      deleted.BrokerID = Brokrage.BrokerID AND

      NOT EXISTS (

        SELECT * FROM IntraDay

        WHERE

          /* %JoinFKPK(IntraDay,Brokrage," = "," AND") */

          IntraDay.BrokerID = Brokrage.BrokerID

      )

)

BEGIN

   SELECT @errno  = 30010,

          @errmsg = 'Cannot delete last IntraDay because Brokrage exists.'

   GOTO error

END


/* erwin Builtin Trigger */

/* Contract  IntraDay on child delete no action */

/* ERWIN_RELATION:CHECKSUM="00000000", PARENT_OWNER="", PARENT_TABLE="Contract"

CHILD_OWNER="", CHILD_TABLE="IntraDay"

P2C_VERB_PHRASE="", C2P_VERB_PHRASE="",

FK_CONSTRAINT="R_12", FK_COLUMNS="ContractID" */
```

```
IF EXISTS (SELECT * FROM deleted,Contract

  WHERE

    /* %JoinFKPK(deleted,Contract," = "," AND") */

    deleted.ContractID = Contract.ContractID AND

    NOT EXISTS (

      SELECT * FROM IntraDay

      WHERE

        /* %JoinFKPK(IntraDay,Contract," = "," AND") */

        IntraDay.ContractID = Contract.ContractID

    )

)

BEGIN

  SELECT @errno  = 30010,

        @errmsg = 'Cannot delete last IntraDay because Contract exists.'

  GOTO error

END


/* erwin Builtin Trigger */

/* Account  IntraDay on child delete no action */

/* ERWIN_RELATION:CHECKSUM="00000000", PARENT_OWNER="", PARENT_TABLE="Account"

CHILD_OWNER="", CHILD_TABLE="IntraDay"

P2C_VERB_PHRASE="", C2P_VERB_PHRASE="",

FK_CONSTRAINT="R_11", FK_COLUMNS="AccountID" */

IF EXISTS (SELECT * FROM deleted,Account
```

```
    WHERE

      /* %JoinFKPK(deleted,Account," = "," AND") */

      deleted.AccountID = Account.AccountID AND

      NOT EXISTS (

        SELECT * FROM IntraDay

        WHERE

          /* %JoinFKPK(IntraDay,Account," = "," AND") */

          IntraDay.AccountID = Account.AccountID

      )

   )

   BEGIN

     SELECT @errno  = 30010,

            @errmsg = 'Cannot delete last IntraDay because Account exists.'

     GOTO error

   END




   /* erwin Builtin Trigger */

   RETURN

error:

   RAISERROR (@errmsg, -- Message text.

            @severity, -- Severity (0~25).

            @state) -- State (0~255).

   rollback transaction
```

END

go

CREATE TRIGGER tU_IntraDay ON IntraDay FOR UPDATE AS

/* erwin Builtin Trigger */

/* UPDATE trigger on IntraDay */

BEGIN

```
  DECLARE  @numrows int,

           @nullcnt int,

           @validcnt int,

           @insTransactionID integer,

           @insAccountID integer,

           @insContractID integer,

           @errno    int,

           @severity int,

           @state    int,

           @errmsg  varchar(255)


  SELECT @numrows = @@rowcount
```

  /* erwin Builtin Trigger */

  /* Taxes IntraDay on child update no action */

  /* ERWIN_RELATION:CHECKSUM="00055fe6", PARENT_OWNER="", PARENT_TABLE="Taxes"

```
  CHILD_OWNER="", CHILD_TABLE="IntraDay"

  P2C_VERB_PHRASE="", C2P_VERB_PHRASE="",

  FK_CONSTRAINT="R_18", FK_COLUMNS="TaxID" */

IF

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

  UPDATE(TaxID)

BEGIN

  SELECT @nullcnt = 0

  SELECT @validcnt = count(*)

    FROM inserted,Taxes

      WHERE

        /* %JoinFKPK(inserted,Taxes) */

        inserted.TaxID = Taxes.TaxID

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

  select @nullcnt = count(*) from inserted where

    inserted.TaxID IS NULL

  IF @validcnt + @nullcnt != @numrows

  BEGIN

    SELECT @errno  = 30007,

          @errmsg = 'Cannot update IntraDay because Taxes does not exist.'

    GOTO error

  END

END
```

```
/* erwin Builtin Trigger */

/* Brokrage  IntraDay on child update no action */

/* ERWIN_RELATION:CHECKSUM="00000000", PARENT_OWNER="", PARENT_TABLE="Brokrage"

  CHILD_OWNER="", CHILD_TABLE="IntraDay"

  P2C_VERB_PHRASE="", C2P_VERB_PHRASE="",

  FK_CONSTRAINT="R_15", FK_COLUMNS="BrokerID" */

IF

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

  UPDATE(BrokerID)

BEGIN

  SELECT @nullcnt = 0

  SELECT @validcnt = count(*)

    FROM inserted,Brokrage

      WHERE

        /* %JoinFKPK(inserted,Brokrage) */

        inserted.BrokerID = Brokrage.BrokerID

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

  select @nullcnt = count(*) from inserted where

    inserted.BrokerID IS NULL

  IF @validcnt + @nullcnt != @numrows

  BEGIN

    SELECT @errno  = 30007,

          @errmsg = 'Cannot update IntraDay because Brokrage does not exist.'
```

```
    GOTO error

  END

END


/* erwin Builtin Trigger */

/* Contract  IntraDay on child update no action */

/* ERWIN_RELATION:CHECKSUM="00000000", PARENT_OWNER="", PARENT_TABLE="Contract"

  CHILD_OWNER="", CHILD_TABLE="IntraDay"

  P2C_VERB_PHRASE="", C2P_VERB_PHRASE="",

  FK_CONSTRAINT="R_12", FK_COLUMNS="ContractID" */

IF

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

  UPDATE(ContractID)

BEGIN

  SELECT @nullcnt = 0

  SELECT @validcnt = count(*)

    FROM inserted,Contract

      WHERE

        /* %JoinFKPK(inserted,Contract) */

        inserted.ContractID = Contract.ContractID

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


  IF @validcnt + @nullcnt != @numrows
```

```
  BEGIN

    SELECT @errno  = 30007,

          @errmsg = 'Cannot update IntraDay because Contract does not exist.'

      GOTO error

    END

  END


  /* erwin Builtin Trigger */

  /* Account  IntraDay on child update no action */

  /* ERWIN_RELATION:CHECKSUM="00000000", PARENT_OWNER="", PARENT_TABLE="Account"

    CHILD_OWNER="", CHILD_TABLE="IntraDay"

    P2C_VERB_PHRASE="", C2P_VERB_PHRASE="",

    FK_CONSTRAINT="R_11", FK_COLUMNS="AccountID" */

  IF

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

    UPDATE(AccountID)

  BEGIN

    SELECT @nullcnt = 0

    SELECT @validcnt = count(*)

      FROM inserted,Account

        WHERE

          /* %JoinFKPK(inserted,Account) */

          inserted.AccountID = Account.AccountID

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

```
   IF @validcnt + @nullcnt != @numrows

   BEGIN

     SELECT @errno  = 30007,

            @errmsg = 'Cannot update IntraDay because Account does not exist.'

       GOTO error

   END

 END



 /* erwin Builtin Trigger */

 RETURN

error:

  RAISERROR (@errmsg, -- Message text.

             @severity, -- Severity (0~25).

             @state) -- State (0~255).

    rollback transaction

END


go
```

```
CREATE TRIGGER tD_Inventory ON Inventory FOR DELETE AS

/* erwin Builtin Trigger */

/* DELETE trigger on Inventory */

BEGIN

  DECLARE  @errno    int,

           @severity int,

           @state    int,

           @errmsg  varchar(255)

    /* erwin Builtin Trigger */

    /* Sell   Inventory on child delete no action */

    /* ERWIN_RELATION:CHECKSUM="00053cdf", PARENT_OWNER="", PARENT_TABLE="Sell"

    CHILD_OWNER="", CHILD_TABLE="Inventory"

    P2C_VERB_PHRASE="", C2P_VERB_PHRASE="",

    FK_CONSTRAINT="R_22", FK_COLUMNS="AccountID""ContractID""SellID" */

    IF EXISTS (SELECT * FROM deleted,Sell

      WHERE

        /* %JoinFKPK(deleted,Sell," = "," AND") */

        deleted.AccountID = Sell.AccountID AND

        deleted.ContractID = Sell.ContractID AND

        deleted.SellID = Sell.SellID AND

        NOT EXISTS (

          SELECT * FROM Inventory

          WHERE

            /* %JoinFKPK(Inventory,Sell," = "," AND") */
```

Page 59

```
        Inventory.AccountID = Sell.AccountID AND

        Inventory.ContractID = Sell.ContractID AND

        Inventory.SellID = Sell.SellID

    )

)

BEGIN

  SELECT @errno  = 30010,

         @errmsg = 'Cannot delete last Inventory because Sell exists.'

  GOTO error

END


/* erwin Builtin Trigger */

/* Buy  Inventory on child delete no action */

/* ERWIN_RELATION:CHECKSUM="00000000", PARENT_OWNER="", PARENT_TABLE="Buy"

CHILD_OWNER="", CHILD_TABLE="Inventory"

P2C_VERB_PHRASE="", C2P_VERB_PHRASE="",

FK_CONSTRAINT="R_21", FK_COLUMNS="AccountID""ContractID""BuyID" */

IF EXISTS (SELECT * FROM deleted,Buy

  WHERE

    /* %JoinFKPK(deleted,Buy," = "," AND") */

    deleted.AccountID = Buy.AccountID AND

    deleted.ContractID = Buy.ContractID AND

    deleted.BuyID = Buy.BuyID AND

    NOT EXISTS (
```

```
      SELECT * FROM Inventory

      WHERE

        /* %JoinFKPK(Inventory,Buy," = "," AND") */

        Inventory.AccountID = Buy.AccountID AND

        Inventory.ContractID = Buy.ContractID AND

        Inventory.BuyID = Buy.BuyID

    )

)

BEGIN

  SELECT @errno  = 30010,

         @errmsg = 'Cannot delete last Inventory because Buy exists.'

  GOTO error

END


/* erwin Builtin Trigger */

/* Contract  Inventory on child delete no action */

/* ERWIN_RELATION:CHECKSUM="00000000", PARENT_OWNER="", PARENT_TABLE="Contract"

CHILD_OWNER="", CHILD_TABLE="Inventory"

P2C_VERB_PHRASE="", C2P_VERB_PHRASE="",

FK_CONSTRAINT="R_20", FK_COLUMNS="ContractID" */

IF EXISTS (SELECT * FROM deleted,Contract

  WHERE

    /* %JoinFKPK(deleted,Contract," = "," AND") */

    deleted.ContractID = Contract.ContractID AND
```

```
    NOT EXISTS (

      SELECT * FROM Inventory

      WHERE

        /* %JoinFKPK(Inventory,Contract," = "," AND") */

        Inventory.ContractID = Contract.ContractID

    )

)

BEGIN

  SELECT @errno  = 30010,

         @errmsg = 'Cannot delete last Inventory because Contract exists.'

  GOTO error

END


/* erwin Builtin Trigger */

/* Account  Inventory on child delete no action */

/* ERWIN_RELATION:CHECKSUM="00000000", PARENT_OWNER="", PARENT_TABLE="Account"

CHILD_OWNER="", CHILD_TABLE="Inventory"

P2C_VERB_PHRASE="", C2P_VERB_PHRASE="",

FK_CONSTRAINT="R_19", FK_COLUMNS="AccountID" */

IF EXISTS (SELECT * FROM deleted,Account

  WHERE

    /* %JoinFKPK(deleted,Account," = "," AND") */

    deleted.AccountID = Account.AccountID AND

    NOT EXISTS (
```

```
       SELECT * FROM Inventory

       WHERE

         /* %JoinFKPK(Inventory,Account," = "," AND") */

         Inventory.AccountID = Account.AccountID

     )

   )

   BEGIN

     SELECT @errno  = 30010,

            @errmsg = 'Cannot delete last Inventory because Account exists.'

     GOTO error

   END




   /* erwin Builtin Trigger */

   RETURN

error:

   RAISERROR (@errmsg, -- Message text.

             @severity, -- Severity (0~25).

             @state) -- State (0~255).

   rollback transaction

END


go
```

```
CREATE TRIGGER tU_Inventory ON Inventory FOR UPDATE AS

/* erwin Builtin Trigger */

/* UPDATE trigger on Inventory */

BEGIN

  DECLARE  @numrows int,

           @nullcnt int,

           @validcnt int,

           @insAccountID integer,

           @insContractID integer,

           @insInventoryItemID integer,

           @errno   int,

           @severity int,

           @state    int,

           @errmsg  varchar(255)


  SELECT @numrows = @@rowcount

  /* erwin Builtin Trigger */

  /* Sell  Inventory on child update no action */

  /* ERWIN_RELATION:CHECKSUM="0005f276", PARENT_OWNER="", PARENT_TABLE="Sell"

    CHILD_OWNER="", CHILD_TABLE="Inventory"

    P2C_VERB_PHRASE="", C2P_VERB_PHRASE="",

    FK_CONSTRAINT="R_22", FK_COLUMNS="AccountID""ContractID""SellID" */

  IF
```

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

   UPDATE(AccountID) OR

   UPDATE(ContractID) OR

   UPDATE(SellID)

 BEGIN

   SELECT @nullcnt = 0

   SELECT @validcnt = count(*)

     FROM inserted,Sell

       WHERE

         /* %JoinFKPK(inserted,Sell) */

         inserted.AccountID = Sell.AccountID and

         inserted.ContractID = Sell.ContractID and

         inserted.SellID = Sell.SellID

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

   select @nullcnt = count(*) from inserted where

     inserted.AccountID IS NULL AND

     inserted.ContractID IS NULL AND

     inserted.SellID IS NULL

   IF @validcnt + @nullcnt != @numrows

   BEGIN

     SELECT @errno  = 30007,

           @errmsg = 'Cannot update Inventory because Sell does not exist.'

     GOTO error
```

```
   END

 END


 /* erwin Builtin Trigger */

 /* Buy   Inventory on child update no action */

 /* ERWIN_RELATION:CHECKSUM="00000000", PARENT_OWNER="", PARENT_TABLE="Buy"

   CHILD_OWNER="", CHILD_TABLE="Inventory"

   P2C_VERB_PHRASE="", C2P_VERB_PHRASE="",

   FK_CONSTRAINT="R_21", FK_COLUMNS="AccountID""ContractID""BuyID" */

 IF

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

   UPDATE(AccountID) OR

   UPDATE(ContractID) OR

   UPDATE(BuyID)

 BEGIN

   SELECT @nullcnt = 0

   SELECT @validcnt = count(*)

     FROM inserted,Buy

       WHERE

         /* %JoinFKPK(inserted,Buy) */

         inserted.AccountID = Buy.AccountID and

         inserted.ContractID = Buy.ContractID and

         inserted.BuyID = Buy.BuyID

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

```
select @nullcnt = count(*) from inserted where

  inserted.AccountID IS NULL AND

  inserted.ContractID IS NULL AND

  inserted.BuyID IS NULL

IF @validcnt + @nullcnt != @numrows

BEGIN

  SELECT @errno  = 30007,

         @errmsg = 'Cannot update Inventory because Buy does not exist.'

  GOTO error

END

END


/* erwin Builtin Trigger */

/* Contract  Inventory on child update no action */

/* ERWIN_RELATION:CHECKSUM="00000000", PARENT_OWNER="", PARENT_TABLE="Contract"

  CHILD_OWNER="", CHILD_TABLE="Inventory"

  P2C_VERB_PHRASE="", C2P_VERB_PHRASE="",

  FK_CONSTRAINT="R_20", FK_COLUMNS="ContractID" */

IF

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

  UPDATE(ContractID)

BEGIN

  SELECT @nullcnt = 0

  SELECT @validcnt = count(*)
```

```
    FROM inserted,Contract

      WHERE

        /* %JoinFKPK(inserted,Contract) */

        inserted.ContractID = Contract.ContractID

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



   IF @validcnt + @nullcnt != @numrows

   BEGIN

     SELECT @errno  = 30007,

            @errmsg = 'Cannot update Inventory because Contract does not exist.'

     GOTO error

   END

 END


 /* erwin Builtin Trigger */

 /* Account  Inventory on child update no action */

 /* ERWIN_RELATION:CHECKSUM="00000000", PARENT_OWNER="", PARENT_TABLE="Account"

   CHILD_OWNER="", CHILD_TABLE="Inventory"

   P2C_VERB_PHRASE="", C2P_VERB_PHRASE="",

   FK_CONSTRAINT="R_19", FK_COLUMNS="AccountID" */

 IF

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

   UPDATE(AccountID)
```

```
  BEGIN

    SELECT @nullcnt = 0

    SELECT @validcnt = count(*)

      FROM inserted,Account

        WHERE

          /* %JoinFKPK(inserted,Account) */

          inserted.AccountID = Account.AccountID

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



    IF @validcnt + @nullcnt != @numrows

    BEGIN

      SELECT @errno  = 30007,

            @errmsg = 'Cannot update Inventory because Account does not exist.'

      GOTO error

    END

  END



  /* erwin Builtin Trigger */

  RETURN

error:

   RAISERROR (@errmsg, -- Message text.

            @severity, -- Severity (0~25).

            @state) -- State (0~255).
```

```
     rollback transaction

END


go




CREATE TRIGGER tD_Sell ON Sell FOR DELETE AS

/* erwin Builtin Trigger */

/* DELETE trigger on Sell */

BEGIN

  DECLARE  @errno   int,

           @severity int,

           @state    int,

           @errmsg  varchar(255)

    /* erwin Builtin Trigger */

    /* Sell   Inventory on parent delete no action */

    /* ERWIN_RELATION:CHECKSUM="000583b6", PARENT_OWNER="", PARENT_TABLE="Sell"

    CHILD_OWNER="", CHILD_TABLE="Inventory"

    P2C_VERB_PHRASE="", C2P_VERB_PHRASE="",

    FK_CONSTRAINT="R_22", FK_COLUMNS="AccountID""ContractID""SellID" */

    IF EXISTS (

      SELECT * FROM deleted,Inventory
```

```
   WHERE

     /*  %JoinFKPK(Inventory,deleted," = "," AND") */

     Inventory.AccountID = deleted.AccountID AND

     Inventory.ContractID = deleted.ContractID AND

     Inventory.SellID = deleted.SellID

)

BEGIN

  SELECT @errno  = 30001,

         @errmsg = 'Cannot delete Sell because Inventory exists.'

   GOTO error

END


/* erwin Builtin Trigger */

/* Taxes  Sell on child delete no action */

/* ERWIN_RELATION:CHECKSUM="00000000", PARENT_OWNER="", PARENT_TABLE="Taxes"

CHILD_OWNER="", CHILD_TABLE="Sell"

P2C_VERB_PHRASE="", C2P_VERB_PHRASE="",

FK_CONSTRAINT="R_17", FK_COLUMNS="TaxID" */

IF EXISTS (SELECT * FROM deleted,Taxes

   WHERE

     /* %JoinFKPK(deleted,Taxes," = "," AND") */

     deleted.TaxID = Taxes.TaxID AND

     NOT EXISTS (

       SELECT * FROM Sell
```

```
      WHERE

        /* %JoinFKPK(Sell,Taxes," = "," AND") */

        Sell.TaxID = Taxes.TaxID

    )

)

BEGIN

   SELECT @errno  = 30010,

          @errmsg = 'Cannot delete last Sell because Taxes exists.'

   GOTO error

END


/* erwin Builtin Trigger */

/* Brokrage  Sell on child delete no action */

/* ERWIN_RELATION:CHECKSUM="00000000", PARENT_OWNER="", PARENT_TABLE="Brokrage"

CHILD_OWNER="", CHILD_TABLE="Sell"

P2C_VERB_PHRASE="", C2P_VERB_PHRASE="",

FK_CONSTRAINT="R_14", FK_COLUMNS="BrokerID" */

IF EXISTS (SELECT * FROM deleted,Brokrage

   WHERE

     /* %JoinFKPK(deleted,Brokrage," = "," AND") */

     deleted.BrokerID = Brokrage.BrokerID AND

     NOT EXISTS (

       SELECT * FROM Sell

       WHERE
```

```
        /* %JoinFKPK(Sell,Brokrage," = "," AND") */

        Sell.BrokerID = Brokrage.BrokerID

    )

)

BEGIN

  SELECT @errno  = 30010,

         @errmsg = 'Cannot delete last Sell because Brokrage exists.'

  GOTO error

END


/* erwin Builtin Trigger */

/* Contract  Sell on child delete no action */

/* ERWIN_RELATION:CHECKSUM="00000000", PARENT_OWNER="", PARENT_TABLE="Contract"

CHILD_OWNER="", CHILD_TABLE="Sell"

P2C_VERB_PHRASE="", C2P_VERB_PHRASE="",

FK_CONSTRAINT="R_4", FK_COLUMNS="ContractID" */

IF EXISTS (SELECT * FROM deleted,Contract

  WHERE

    /* %JoinFKPK(deleted,Contract," = "," AND") */

    deleted.ContractID = Contract.ContractID AND

    NOT EXISTS (

      SELECT * FROM Sell

      WHERE

        /* %JoinFKPK(Sell,Contract," = "," AND") */
```

```
        Sell.ContractID = Contract.ContractID

    )

)

BEGIN

  SELECT @errno  = 30010,

         @errmsg = 'Cannot delete last Sell because Contract exists.'

  GOTO error

END



/* erwin Builtin Trigger */

/* Account  Sell on child delete no action */

/* ERWIN_RELATION:CHECKSUM="00000000", PARENT_OWNER="", PARENT_TABLE="Account"

CHILD_OWNER="", CHILD_TABLE="Sell"

P2C_VERB_PHRASE="", C2P_VERB_PHRASE="",

FK_CONSTRAINT="R_3", FK_COLUMNS="AccountID" */

IF EXISTS (SELECT * FROM deleted,Account

  WHERE

    /* %JoinFKPK(deleted,Account," = "," AND") */

    deleted.AccountID = Account.AccountID AND

    NOT EXISTS (

      SELECT * FROM Sell

      WHERE

        /* %JoinFKPK(Sell,Account," = "," AND") */

        Sell.AccountID = Account.AccountID
```

```
        )

    )

    BEGIN

      SELECT @errno  = 30010,

             @errmsg = 'Cannot delete last Sell because Account exists.'

        GOTO error

    END




    /* erwin Builtin Trigger */

    RETURN

error:

   RAISERROR (@errmsg, -- Message text.

              @severity, -- Severity (0~25).

              @state) -- State (0~255).

    rollback transaction

END



go



CREATE TRIGGER tU_Sell ON Sell FOR UPDATE AS

/* erwin Builtin Trigger */

/* UPDATE trigger on Sell */
```

```
BEGIN

  DECLARE  @numrows int,

           @nullcnt int,

           @validcnt int,

           @insAccountID integer,

           @insContractID integer,

           @insSellID integer,

           @errno   int,

           @severity int,

           @state    int,

           @errmsg  varchar(255)


  SELECT @numrows = @@rowcount

  /* erwin Builtin Trigger */

  /* Sell   Inventory on parent update no action */

  /* ERWIN_RELATION:CHECKSUM="00067634", PARENT_OWNER="", PARENT_TABLE="Sell"

    CHILD_OWNER="", CHILD_TABLE="Inventory"

    P2C_VERB_PHRASE="", C2P_VERB_PHRASE="",

    FK_CONSTRAINT="R_22", FK_COLUMNS="AccountID""ContractID""SellID" */

  IF

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

    UPDATE(AccountID) OR

    UPDATE(ContractID) OR

    UPDATE(SellID)
```

```
BEGIN

  IF EXISTS (

    SELECT * FROM deleted,Inventory

    WHERE

      /*  %JoinFKPK(Inventory,deleted," = "," AND") */

      Inventory.AccountID = deleted.AccountID AND

      Inventory.ContractID = deleted.ContractID AND

      Inventory.SellID = deleted.SellID

  )

  BEGIN

    SELECT @errno  = 30005,

           @errmsg = 'Cannot update Sell because Inventory exists.'

    GOTO error

  END

END


/* erwin Builtin Trigger */

/* Taxes  Sell on child update no action */

/* ERWIN_RELATION:CHECKSUM="00000000", PARENT_OWNER="", PARENT_TABLE="Taxes"

  CHILD_OWNER="", CHILD_TABLE="Sell"

  P2C_VERB_PHRASE="", C2P_VERB_PHRASE="",

  FK_CONSTRAINT="R_17", FK_COLUMNS="TaxID" */

IF

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

```
   UPDATE(TaxID)

 BEGIN

   SELECT @nullcnt = 0

   SELECT @validcnt = count(*)

     FROM inserted,Taxes

       WHERE

         /* %JoinFKPK(inserted,Taxes) */

         inserted.TaxID = Taxes.TaxID

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

   select @nullcnt = count(*) from inserted where

     inserted.TaxID IS NULL

   IF @validcnt + @nullcnt != @numrows

   BEGIN

     SELECT @errno  = 30007,

           @errmsg = 'Cannot update Sell because Taxes does not exist.'

     GOTO error

   END

 END


 /* erwin Builtin Trigger */

 /* Brokrage  Sell on child update no action */

 /* ERWIN_RELATION:CHECKSUM="00000000", PARENT_OWNER="", PARENT_TABLE="Brokrage"

   CHILD_OWNER="", CHILD_TABLE="Sell"
```

```
   P2C_VERB_PHRASE="", C2P_VERB_PHRASE="",

   FK_CONSTRAINT="R_14", FK_COLUMNS="BrokerID" */

 IF

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

   UPDATE(BrokerID)

 BEGIN

   SELECT @nullcnt = 0

   SELECT @validcnt = count(*)

     FROM inserted,Brokrage

       WHERE

         /* %JoinFKPK(inserted,Brokrage) */

         inserted.BrokerID = Brokrage.BrokerID

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

   select @nullcnt = count(*) from inserted where

     inserted.BrokerID IS NULL

   IF @validcnt + @nullcnt != @numrows

   BEGIN

     SELECT @errno  = 30007,

           @errmsg = 'Cannot update Sell because Brokrage does not exist.'

     GOTO error

   END

 END


 /* erwin Builtin Trigger */
```

```
/* Contract  Sell on child update no action */

/* ERWIN_RELATION:CHECKSUM="00000000", PARENT_OWNER="", PARENT_TABLE="Contract"

  CHILD_OWNER="", CHILD_TABLE="Sell"

  P2C_VERB_PHRASE="", C2P_VERB_PHRASE="",

  FK_CONSTRAINT="R_4", FK_COLUMNS="ContractID" */

IF

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

  UPDATE(ContractID)

BEGIN

  SELECT @nullcnt = 0

  SELECT @validcnt = count(*)

    FROM inserted,Contract

      WHERE

        /* %JoinFKPK(inserted,Contract) */

        inserted.ContractID = Contract.ContractID

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


    IF @validcnt + @nullcnt != @numrows

    BEGIN

      SELECT @errno  = 30007,

             @errmsg = 'Cannot update Sell because Contract does not exist.'

      GOTO error

    END
```

END


```
/* erwin Builtin Trigger */

/* Account  Sell on child update no action */

/* ERWIN_RELATION:CHECKSUM="00000000", PARENT_OWNER="", PARENT_TABLE="Account"

  CHILD_OWNER="", CHILD_TABLE="Sell"

  P2C_VERB_PHRASE="", C2P_VERB_PHRASE="",

  FK_CONSTRAINT="R_3", FK_COLUMNS="AccountID" */

IF

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

  UPDATE(AccountID)

BEGIN

  SELECT @nullcnt = 0

  SELECT @validcnt = count(*)

    FROM inserted,Account

      WHERE

        /* %JoinFKPK(inserted,Account) */

        inserted.AccountID = Account.AccountID

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


  IF @validcnt + @nullcnt != @numrows

  BEGIN

    SELECT @errno  = 30007,

           @errmsg = 'Cannot update Sell because Account does not exist.'
```

```
      GOTO error

    END

  END



  /* erwin Builtin Trigger */

  RETURN

error:

    RAISERROR (@errmsg, -- Message text.

              @severity, -- Severity (0~25).

              @state) -- State (0~255).

    rollback transaction

END



go




CREATE TRIGGER tD_Taxes ON Taxes FOR DELETE AS

/* erwin Builtin Trigger */

/* DELETE trigger on Taxes */

BEGIN

  DECLARE  @errno   int,
```

```
        @severity int,

        @state    int,

        @errmsg   varchar(255)

/* erwin Builtin Trigger */

/* Taxes   IntraDay on parent delete no action */

/* ERWIN_RELATION:CHECKSUM="0002add2", PARENT_OWNER="", PARENT_TABLE="Taxes"

CHILD_OWNER="", CHILD_TABLE="IntraDay"

P2C_VERB_PHRASE="", C2P_VERB_PHRASE="",

FK_CONSTRAINT="R_18", FK_COLUMNS="TaxID" */

IF EXISTS (

  SELECT * FROM deleted,IntraDay

  WHERE

    /*  %JoinFKPK(IntraDay,deleted," = "," AND") */

    IntraDay.TaxID = deleted.TaxID

)

BEGIN

  SELECT @errno  = 30001,

         @errmsg = 'Cannot delete Taxes because IntraDay exists.'

  GOTO error

END


/* erwin Builtin Trigger */

/* Taxes   Sell on parent delete no action */

/* ERWIN_RELATION:CHECKSUM="00000000", PARENT_OWNER="", PARENT_TABLE="Taxes"
```

```
CHILD_OWNER="", CHILD_TABLE="Sell"

P2C_VERB_PHRASE="", C2P_VERB_PHRASE="",

FK_CONSTRAINT="R_17", FK_COLUMNS="TaxID" */

IF EXISTS (

  SELECT * FROM deleted,Sell

  WHERE

    /*  %JoinFKPK(Sell,deleted," = "," AND") */

    Sell.TaxID = deleted.TaxID

)

BEGIN

  SELECT @errno  = 30001,

         @errmsg = 'Cannot delete Taxes because Sell exists.'

  GOTO error

END


/* erwin Builtin Trigger */

/* Taxes  Buy on parent delete no action */

/* ERWIN_RELATION:CHECKSUM="00000000", PARENT_OWNER="", PARENT_TABLE="Taxes"

CHILD_OWNER="", CHILD_TABLE="Buy"

P2C_VERB_PHRASE="", C2P_VERB_PHRASE="",

FK_CONSTRAINT="R_16", FK_COLUMNS="TaxID" */

IF EXISTS (

  SELECT * FROM deleted,Buy

  WHERE
```

```
        /*  %JoinFKPK(Buy,deleted," = "," AND") */

        Buy.TaxID = deleted.TaxID

    )

    BEGIN

      SELECT @errno  = 30001,

             @errmsg = 'Cannot delete Taxes because Buy exists.'

      GOTO error

    END




    /* erwin Builtin Trigger */

    RETURN

error:

    RAISERROR (@errmsg, -- Message text.

               @severity, -- Severity (0~25).

               @state) -- State (0~255).

    rollback transaction

END



go



CREATE TRIGGER tU_Taxes ON Taxes FOR UPDATE AS

/* erwin Builtin Trigger */
```

```
/* UPDATE trigger on Taxes */

BEGIN

  DECLARE  @numrows int,

           @nullcnt int,

           @validcnt int,

           @insTaxID integer,

           @errno    int,

           @severity int,

           @state    int,

           @errmsg   varchar(255)


  SELECT @numrows = @@rowcount

  /* erwin Builtin Trigger */

  /* Taxes   IntraDay on parent update no action */

  /* ERWIN_RELATION:CHECKSUM="0002e7fa", PARENT_OWNER="", PARENT_TABLE="Taxes"

    CHILD_OWNER="", CHILD_TABLE="IntraDay"

    P2C_VERB_PHRASE="", C2P_VERB_PHRASE="",

    FK_CONSTRAINT="R_18", FK_COLUMNS="TaxID" */

  IF

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

    UPDATE(TaxID)

  BEGIN

    IF EXISTS (

      SELECT * FROM deleted,IntraDay
```

```
    WHERE

      /*  %JoinFKPK(IntraDay,deleted," = "," AND") */

      IntraDay.TaxID = deleted.TaxID

  )

  BEGIN

    SELECT @errno  = 30005,

           @errmsg = 'Cannot update Taxes because IntraDay exists.'

    GOTO error

  END

END


/* erwin Builtin Trigger */

/* Taxes  Sell on parent update no action */

/* ERWIN_RELATION:CHECKSUM="00000000", PARENT_OWNER="", PARENT_TABLE="Taxes"

  CHILD_OWNER="", CHILD_TABLE="Sell"

  P2C_VERB_PHRASE="", C2P_VERB_PHRASE="",

  FK_CONSTRAINT="R_17", FK_COLUMNS="TaxID" */

IF

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

  UPDATE(TaxID)

BEGIN

  IF EXISTS (

    SELECT * FROM deleted,Sell

    WHERE
```

```
    /*  %JoinFKPK(Sell,deleted," = "," AND") */

    Sell.TaxID = deleted.TaxID

)

BEGIN

  SELECT @errno  = 30005,

         @errmsg = 'Cannot update Taxes because Sell exists.'

  GOTO error

END

END


/* erwin Builtin Trigger */

/* Taxes  Buy on parent update no action */

/* ERWIN_RELATION:CHECKSUM="00000000", PARENT_OWNER="", PARENT_TABLE="Taxes"

  CHILD_OWNER="", CHILD_TABLE="Buy"

  P2C_VERB_PHRASE="", C2P_VERB_PHRASE="",

  FK_CONSTRAINT="R_16", FK_COLUMNS="TaxID" */

IF

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

  UPDATE(TaxID)

BEGIN

  IF EXISTS (

    SELECT * FROM deleted,Buy

    WHERE

      /*  %JoinFKPK(Buy,deleted," = "," AND") */
```

```
        Buy.TaxID = deleted.TaxID

    )

    BEGIN

      SELECT @errno  = 30005,

             @errmsg = 'Cannot update Taxes because Buy exists.'

      GOTO error

    END

  END



  /* erwin Builtin Trigger */

  RETURN

error:

    RAISERROR (@errmsg, -- Message text.

             @severity, -- Severity (0~25).

             @state) -- State (0~255).

    rollback transaction

END


go
```