



pears.BoilerPlatedivision



Generated schema reference. Regenerate this page from the SQL unload; keep hand-maintained business notes in the narrative namespace.

Description

Links BoilerPlate records with Divisions. Boiler Plates with no records are available to all.

Columns

Column	Type	Null	Default	Comment
BoilerPlateid	char(20)	NOT NULL		
divisionid	char(20)	NOT NULL		

Primary Key

- BoilerPlateid, divisionid

Foreign Keys

Constraint	Columns	References	Delete/update action
BoilerPlate	BoilerPlateid	pears.BoilerPlate (BoilerPlateID)	NOT NULL; ON DELETE CASCADE
division	divisionid	pears.Division (divisionid)	NOT NULL; ON DELETE CASCADE

Referenced By

- No incoming foreign keys found.

Indexes

- No indexes found.



Triggers

Name	Timing	Event
BoilerPlateDivisionDeleteAudit	after	delete order 1
BoilerPlateDivisionInsertAudit	after	insert order 1

Original SQL

```
-- IQX database structure split by table
-- Source: IQXDatabaseStructure - with comments.sql
-- Table: "pears"."BoilerPlatedivision"
-- Table comment: Links BoilerPlate records with Divsions. Boiler Plates
with no records are available to all.
-- Statement count: 8

CREATE TABLE "pears"."BoilerPlatedivision" (
    "BoilerPlateid"          CHAR(20) NOT NULL
    ,"divisionid"            CHAR(20) NOT NULL
    ,PRIMARY KEY ("BoilerPlateid" ASC,"divisionid" ASC)
)
GO

COMMENT ON TABLE "pears"."BoilerPlatedivision" IS
    'Links BoilerPlate records with Divsions. Boiler Plates with no records
are available to all.'
GO

ALTER TABLE "pears"."BoilerPlatedivision"
    ADD NOT NULL FOREIGN KEY "BoilerPlate" ("BoilerPlateid" ASC)
    REFERENCES "pears"."BoilerPlate" ("BoilerPlateID")
    ON DELETE CASCADE
GO

ALTER TABLE "pears"."BoilerPlatedivision"
    ADD NOT NULL FOREIGN KEY "division" ("divisionid" ASC)
    REFERENCES "pears"."Division" ("divisionid")
    ON DELETE CASCADE
GO

CREATE TRIGGER "BoilerPlateDivisionDeleteAudit" after DELETE ORDER 1 ON
"pears"."BoilerPlateDivision"
REFERENCING OLD AS "old_Plate"
```



```
FOR each ROW
BEGIN
    DECLARE "DivName" CHAR(30);
    SELECT "name" INTO "DivName" FROM "Division" WHERE "divisionid" =
"old_Plate"."divisionid";
    CALL
"AuditLog"('BOILERPLATEDIVISION',"old_Plate"."BoilerPlateID",string('Remov
ed Boiler Plate Division - ', "DivName"), "DivName", '')
END
GO
```

```
COMMENT TO PRESERVE FORMAT ON TRIGGER
"pears"."BoilerPlatedivision"."BoilerPlateDivisionDeleteAudit" IS
{CREATE TRIGGER BoilerPlateDivisionDeleteAudit
after DELETE ORDER 1 ON
BoilerPlateDivision
REFERENCING OLD AS old_Plate
FOR each ROW
BEGIN
    DECLARE DivName CHAR(30);
    SELECT name INTO DivName FROM Division WHERE divisionid =
old_Plate.divisionid;
    CALL
AuditLog('BOILERPLATEDIVISION',old_Plate.BoilerPlateID,string('Removed
Boiler Plate Division - ',DivName),DivName,'');
END
}
GO
```

```
CREATE TRIGGER "BoilerPlateDivisionInsertAudit" after INSERT ORDER 1 ON
"pears"."BoilerPlateDivision"
REFERENCING NEW AS "new_Plate"
FOR each ROW
BEGIN
    DECLARE "DivName" CHAR(30);
    SELECT "name" INTO "DivName" FROM "Division" WHERE "divisionid" =
"new_Plate"."divisionID";
    CALL
"AuditLog"('BOILERPLATEDIVISION',"new_Plate"."BoilerPlateID",string('BOILE
RPLATEDIVISION New Division Added : ', "DivName"),'', "DivName")
END
GO
```

```
COMMENT TO PRESERVE FORMAT ON TRIGGER
"pears"."BoilerPlatedivision"."BoilerPlateDivisionInsertAudit" IS
```



```
{CREATE TRIGGER BoilerPlateDivisionInsertAudit
  after INSERT ORDER 1 ON
BoilerPlateDivision
REFERENCING NEW AS new_Plate
FOR each ROW
BEGIN
  DECLARE DivName CHAR(30);
  SELECT name INTO DivName FROM Division WHERE divisionid =
new_Plate.divisionID;
  CALL
AuditLog('BOILERPLATEDIVISION',new_Plate.BoilerPlateID,string('BOILERPLATEDI
VISION New Division Added : ',DivName),'',DivName)
END
}
GO
```

From:

<https://iqxusers.co.uk/iqxhelp/> - iqx

Permanent link:

https://iqxusers.co.uk/iqxhelp/doku.php?id=database:tables:pears_boilerplatedivision

Last update: **2026/08/07 19:24**

