



pears.NotificationTemplate



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

Description

SMS/Email templates for shift notification etc

Columns

Column	Type	Null	Default	Comment
NotificationTemplateID	char(50)	NOT NULL		
Description	char(100)	NOT NULL		
TransportType	char(20)	NOT NULL		EMAIL, SMS
NotificationType	char(20)	NOT NULL		QueryConfirm, QueryMatch, NotifyConfirm etc.
RecipientType	char(20)	NOT NULL		CLIENT, CANDIDATE, AGENCY
TemplateHTML	long varchar	NULL		
SortOrder	smallint	NULL	0	
GroupCode	char(20)	NULL		
Defunct	smallint	NULL	0	
DivisionID	char(20)	NULL		
EmailSubject	char(100)	NULL		
TemplateSortOrder	smallint	NULL	0	
SaveAsDocument	smallint	NULL	0	
ClassCode	char(2)	NULL		
PushNotificationTemplateID	char(20)	NULL		
BoilerPlateID	char(20)	NULL		
bcRecipient	long varchar	NULL		
CheckListDocumentTemplate	smallint	NULL	0	
progressstatus	char(4)	NULL		
DeeplinkIDCheck	long varchar	NULL		
DeeplinkBody	long varchar	NULL		
DeeplinkCompletion	long varchar	NULL		
DeeplinkFailure	long varchar	NULL		



Column	Type	Null	Default	Comment
DeepLinkExpiryDays	tinyint	NULL	30	
ID	bigint	NULL	autoincrement	
ccRecipient	long varchar	NULL		
IQXNetTagGroupID	char(20)	NULL		
EmailToSend	smallint	NULL	0	
ESign	smallint	NULL	0	
OwnerID	char(20)	NULL		
AllowEditing	tinyint	NULL		
IQXNetEmailDetailsID	char(20)	NULL		
FromEmailAddress	char(250)	NULL		
DeeplinkConfirmation	long varchar	NULL		

Primary Key

- NotificationTemplateID

Foreign Keys

Constraint	Columns	References	Delete/update action
division	DivisionID	pears.Division (divisionid)	ON DELETE SET NULL
contactclass	ClassCode	pears.contactclass (classcode)	ON DELETE SET NULL
PushNotificationTemplate	PushNotificationTemplateID	pears.PushNotificationTemplate (PushNotificationTemplateID)	ON DELETE SET NULL
BoilerPlate	BoilerPlateID	pears.BoilerPlate (BoilerPlateID)	ON DELETE SET NULL
IQXNetTagGroup	IQXNetTagGroupID	pears.IQXNetTagGroup (IQXNetTagGroupID)	ON DELETE SET NULL
IQXNetEmailDetails	IQXNetEmailDetailsID	pears.IQXNetEmailDetails (IQXNetEmailDetailsID)	ON DELETE SET NULL

Referenced By

Table	Constraint	Columns	Referenced columns
pears.NotificationDepartment	NotificationTemplate	NotificationTemplateID	NotificationTemplateID
pears.NotificationDocuments	NotificationTemplate	NotificationTemplateID	NotificationTemplateID
pears.NotificationReports	NotificationTemplate	NotificationTemplateID	NotificationTemplateID
pears.ReferenceRequest	NotificationTemplate	NotificationTemplateID	NotificationTemplateID



Indexes

Name	Type	Columns	Detail
NotificationTemplate_ID	Unique index	ID	
NotificationTemplateNotificationType	Index	NotificationType	

Triggers

Name	Timing	Event
NotificationAudit	before	update of "TemplateHTML", "DeeplinkIDCheck", "DeeplinkBody", "DeeplinkCompletion", "DeeplinkFailure" order 2

Original SQL

```
-- IQX database structure split by table
-- Source: IQXDatabaseStructure - with comments.sql
-- Table: "pears"."NotificationTemplate"
-- Table comment: SMS/Email templates for shift notification etc
-- Statement count: 15
```

```
CREATE TABLE "pears"."NotificationTemplate" (
    "NotificationTemplateID"          CHAR(50) NOT NULL
    , "Description"                    CHAR(100) NOT NULL
    , "TransportType"                  CHAR(20) NOT NULL
    , "NotificationType"               CHAR(20) NOT NULL
    , "RecipientType"                  CHAR(20) NOT NULL
    , "TemplateHTML"                   long VARCHAR NULL
    , "SortOrder"                      SMALLINT NULL DEFAULT 0
    , "GroupCode"                      CHAR(20) NULL
    , "Defunct"                        SMALLINT NULL DEFAULT 0
    , "DivisionID"                     CHAR(20) NULL
    , "EmailSubject"                   CHAR(100) NULL
    , "TemplateSortOrder"              SMALLINT NULL DEFAULT 0
    , "SaveAsDocument"                SMALLINT NULL DEFAULT 0
    , "ClassCode"                      CHAR(2) NULL
    , "PushNotificationTemplateID"      CHAR(20) NULL
    , "BoilerPlateID"                  CHAR(20) NULL
    , "bcRecipient"                    long VARCHAR NULL
    , "CheckListDocumentTemplate"      SMALLINT NULL DEFAULT 0
    , "progressstatus"                 CHAR(4) NULL
    , "DeeplinkIDCheck"                long VARCHAR NULL
    , "DeeplinkBody"                   long VARCHAR NULL
    , "DeeplinkCompletion"             long VARCHAR NULL
    , "DeeplinkFailure"                long VARCHAR NULL
    , "DeepLinkExpiryDays"             tinyint NULL DEFAULT 30
```



```
, "ID"                BIGINT NULL DEFAULT autoincrement
, "ccRecipient"       long VARCHAR NULL
, "IQXNetTagGroupID"   CHAR(20) NULL
, "EmailToSend"        SMALLINT NULL DEFAULT 0
, "ESign"              SMALLINT NULL DEFAULT 0
, "OwnerID"            CHAR(20) NULL
, "AllowEditing"       tinyint NULL
, "IQXNetEmailDetailsID" CHAR(20) NULL
, "FromEmailAddress"   CHAR(250) NULL
, "DeeplinkConfirmation" long VARCHAR NULL
, PRIMARY KEY ("NotificationTemplateID" ASC)
)
GO

COMMENT ON COLUMN "pears"."NotificationTemplate"."TransportType" IS
    'EMAIL, SMS'
GO

COMMENT ON COLUMN "pears"."NotificationTemplate"."NotificationType" IS
    'QueryConfirm, QueryMatch, NotifyConfirm etc.'
GO

COMMENT ON COLUMN "pears"."NotificationTemplate"."RecipientType" IS
    'CLIENT, CANDIDATE, AGENCY'
GO

COMMENT ON TABLE "pears"."NotificationTemplate" IS
    'SMS/Email templates for shift notification etc'
GO

ALTER TABLE "pears"."NotificationTemplate"
    ADD FOREIGN KEY "division" ("DivisionID" ASC)
    REFERENCES "pears"."Division" ("divisionid")
    ON DELETE SET NULL
GO

ALTER TABLE "pears"."NotificationTemplate"
    ADD FOREIGN KEY "contactclass" ("ClassCode" ASC)
    REFERENCES "pears"."contactclass" ("classcode")
    ON DELETE SET NULL
GO
```



```
ALTER TABLE "pears"."NotificationTemplate"  
    ADD FOREIGN KEY "PushNotificationTemplate" ("PushNotificationTemplateID"  
ASC)  
    REFERENCES "pears"."PushNotificationTemplate"  
    ("PushNotificationTemplateID")  
    ON DELETE SET NULL  
GO
```

```
ALTER TABLE "pears"."NotificationTemplate"  
    ADD FOREIGN KEY "BoilerPlate" ("BoilerPlateID" ASC)  
    REFERENCES "pears"."BoilerPlate" ("BoilerPlateID")  
    ON DELETE SET NULL  
GO
```

```
ALTER TABLE "pears"."NotificationTemplate"  
    ADD FOREIGN KEY "IQXNetTagGroup" ("IQXNetTagGroupID" ASC)  
    REFERENCES "pears"."IQXNetTagGroup" ("IQXNetTagGroupID")  
    ON DELETE SET NULL  
GO
```

```
ALTER TABLE "pears"."NotificationTemplate"  
    ADD FOREIGN KEY "IQXNetEmailDetails" ("IQXNetEmailDetailsID" ASC)  
    REFERENCES "pears"."IQXNetEmailDetails" ("IQXNetEmailDetailsID")  
    ON DELETE SET NULL  
GO
```

```
CREATE UNIQUE INDEX "NotificationTemplate_ID" ON  
"pears"."NotificationTemplate"  
    ( "ID" )  
GO
```

```
CREATE INDEX "NotificationTemplateNotificationType" ON  
"pears"."NotificationTemplate"  
    ( "NotificationType" )  
GO
```

```
CREATE TRIGGER "NotificationAudit" BEFORE UPDATE OF "TemplateHTML",  
"DeeplinkIDCheck","DeeplinkBody","DeeplinkCompletion","DeeplinkFailure"  
ORDER 2 ON "pears"."NotificationTemplate"  
REFERENCING OLD AS "old_not" NEW AS "new_not"  
FOR each ROW
```



```
BEGIN
  DECLARE @OldDescrip CHAR(250);
  DECLARE @NewDescrip CHAR(250);
  IF UPDATE("TemplateHTML") THEN
    CALL "AuditLog"('NOTIFY',"old_not"."id",'Template Updated
',"old_not"."TemplateHTML","new_not"."TemplateHTML")
  END IF;
  IF UPDATE("DeeplinkIDCheck") THEN
    CALL "AuditLog"('NOTIFY',"old_not"."id",'Deeplink ID Check Updated
',"old_not"."DeeplinkIDCheck","new_not"."DeeplinkIDCheck")
  END IF;
  IF UPDATE("DeeplinkBody") THEN
    CALL "AuditLog"('NOTIFY',"old_not"."id",'Deeplink Body Updated
',"old_not"."DeeplinkBody","new_not"."DeeplinkBody")
  END IF;
  IF UPDATE("DeeplinkCompletion") THEN
    CALL "AuditLog"('NOTIFY',"old_not"."id",'Deeplink Completion
Updated',"old_not"."DeeplinkCompletion","new_not"."DeeplinkCompletion")
  END IF;
  IF UPDATE("DeeplinkFailure") THEN
    CALL "AuditLog"('NOTIFY',"old_not"."id",'Deeplink Failure Updated
',"old_not"."DeeplinkFailure","new_not"."DeeplinkFailure")
  END IF;
  IF UPDATE("DeeplinkConfirmation") THEN
    CALL "AuditLog"('NOTIFY',"old_not"."id",'DeeplinkConfirmation Updated
',"old_not"."DeeplinkConfirmation","new_not"."DeeplinkConfirmation")
  END IF
END
GO

COMMENT TO PRESERVE FORMAT ON TRIGGER
"pears"."NotificationTemplate"."NotificationAudit" IS
{CREATE TRIGGER NotificationAudit
  BEFORE UPDATE OF
TemplateHTML,DeeplinkIDCheck,DeeplinkBody,DeeplinkCompletion,DeeplinkFailure
ORDER 2 ON pears.NotificationTemplate
REFERENCING OLD AS old_not NEW AS new_not
FOR each ROW
BEGIN
  DECLARE @OldDescrip CHAR(250);
  DECLARE @NewDescrip CHAR(250);
  IF UPDATE(TemplateHTML) THEN
    CALL AuditLog('NOTIFY',old_not.id,'Template Updated
',old_not.TemplateHTML,new_not.TemplateHTML)
  END IF;
  IF UPDATE(DeeplinkIDCheck) THEN
    CALL AuditLog('NOTIFY',old_not.id,'Deeplink ID Check Updated
```



```
' ,old_not.DeeplinkIDCheck,new_not.DeeplinkIDCheck)
END IF;
IF UPDATE(DeeplinkBody) THEN
    CALL AuditLog('NOTIFY',old_not.id,'Deeplink Body Updated
' ,old_not.DeeplinkBody,new_not.DeeplinkBody)
END IF;
IF UPDATE(DeeplinkCompletion) THEN
    CALL AuditLog('NOTIFY',old_not.id,'Deeplink Completion
Updated',old_not.DeeplinkCompletion,new_not.DeeplinkCompletion)
END IF;
IF UPDATE(DeeplinkFailure) THEN
    CALL AuditLog('NOTIFY',old_not.id,'Deeplink Failure Updated
' ,old_not.DeeplinkFailure,new_not.DeeplinkFailure)
END IF;
IF UPDATE(DeeplinkConfirmation) THEN
    CALL AuditLog('NOTIFY',old_not.id,'DeeplinkConfirmation Updated
' ,old_not.DeeplinkConfirmation,new_not.DeeplinkConfirmation)
END IF;
END
}
GO
```

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

Permanent link:
https://iqxusers.co.uk/iqxhelp/doku.php?id=database:tables:pears_notificationtemplate

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

