

The background of the slide is a light gray gradient with several realistic water droplets of various sizes scattered across it. The droplets have highlights and shadows, giving them a three-dimensional appearance.

DEVELOPER STORED PROCEDURES

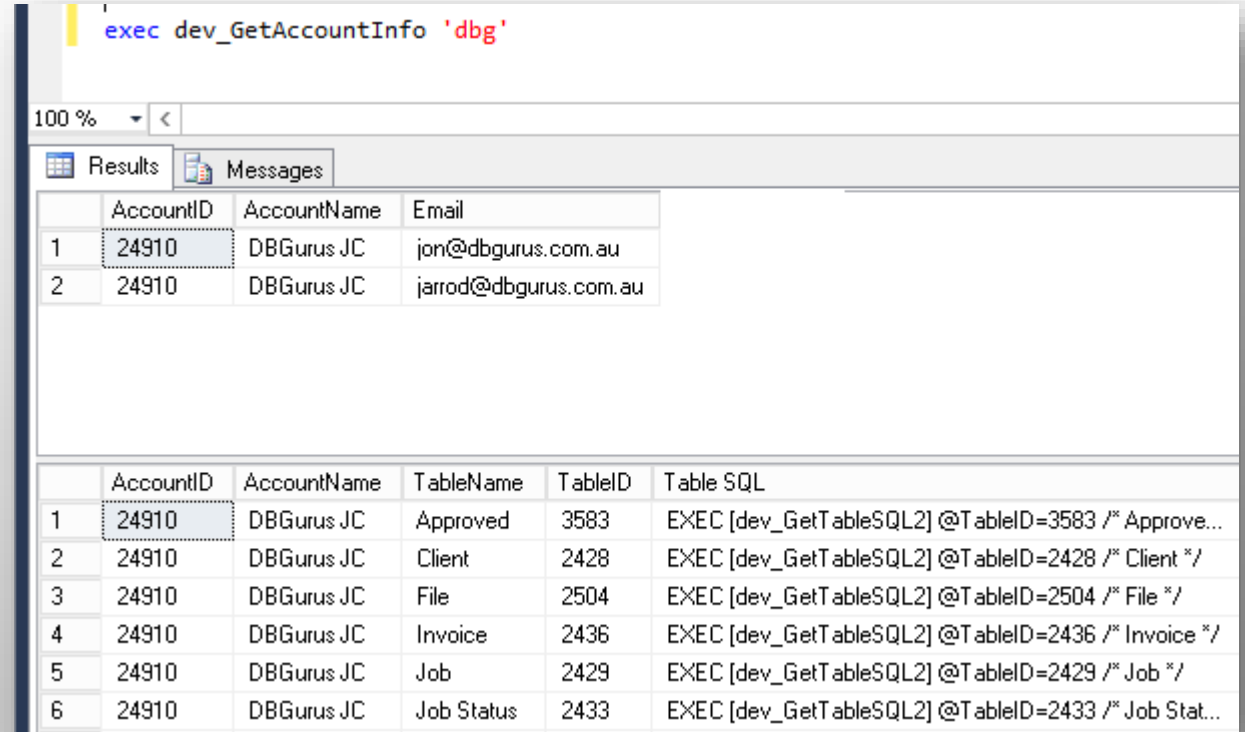
DB GURUS INTERNAL TRAINING

dev_GetAccountInfo

This SP is the starting point,

The 1st result set shows the Account ID, Account Name and Account Holders

The 2nd result set shows the tables and the calls to the other SPs explained here



```
exec dev_GetAccountInfo 'dbg'
```

	AccountID	AccountName	Email
1	24910	DBGurus JC	jon@dbgurus.com.au
2	24910	DBGurus JC	jarrod@dbgurus.com.au

	AccountID	AccountName	TableName	TableID	Table SQL
1	24910	DBGurus JC	Approved	3583	EXEC [dev_GetTableSQL2] @TableID=3583 /* Approve...
2	24910	DBGurus JC	Client	2428	EXEC [dev_GetTableSQL2] @TableID=2428 /* Client */
3	24910	DBGurus JC	File	2504	EXEC [dev_GetTableSQL2] @TableID=2504 /* File */
4	24910	DBGurus JC	Invoice	2436	EXEC [dev_GetTableSQL2] @TableID=2436 /* Invoice */
5	24910	DBGurus JC	Job	2429	EXEC [dev_GetTableSQL2] @TableID=2429 /* Job */
6	24910	DBGurus JC	Job Status	2433	EXEC [dev_GetTableSQL2] @TableID=2433 /* Job Stat...

dev_GetTableSQL2

This SP gives you example SQL statements laid out in the way we like to see them.

It also puts in labels to show you what each field means which saves a lot of time looking things up.

```
EXEC [dev_GetTableSQL2] @TableID=3583 /* Approved */

100 % <
Messages
SELECT      [Approved].[TableID]
            , [Approved].V001 AS [Approved By]
            , [Approved].DateAdded
FROM        [Record] [Approved]
WHERE       [Approved].TableID = 3583
AND         [Approved].IsActive = 1

INSERT INTO [Record] /*Approved*/
(
    [TableID]
    ,EnteredBy
    ,IsActive
    ,V001      /* [Approved By] */
)
VALUES
(
    @TableID
    ,@EnteredBy
    ,@IsActive
    ,@ApprovedBy
)

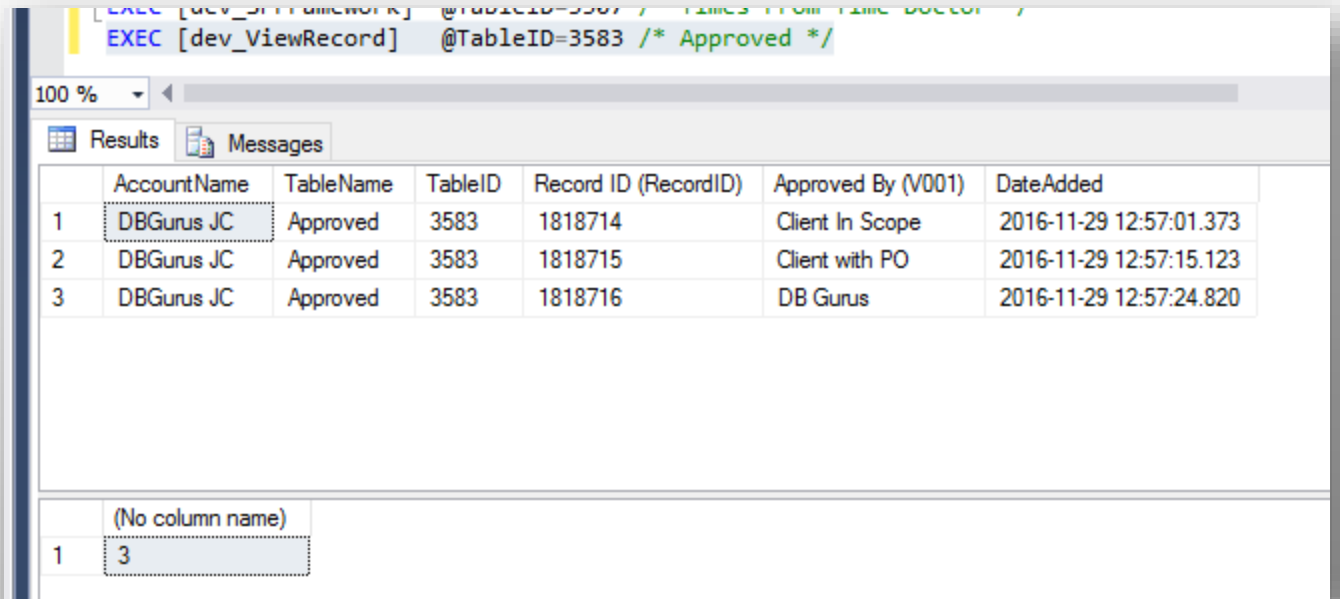
UPDATE [Record] /* Approved */
SET
    ,V001 = @ApprovedBy
WHERE [RecordID] = @RecordID

-- Use with caution
-- DELETE
-- FROM [Record] [Approved]
-- WHERE [TableID] = 3583
```

dev_ViewRecord

Handy little SP that allows you to see the data in a table in its raw form whilst also showing you the DisplayName and SystemName in the column headers.

2nd result set tells you how many rows of data there are.



EXEC [dev_ViewRecord] @TableID=3583 /* Approved */

100 %

Results Messages

	AccountName	TableName	TableID	Record ID (RecordID)	Approved By (V001)	DateAdded
1	DBGurus JC	Approved	3583	1818714	Client In Scope	2016-11-29 12:57:01.373
2	DBGurus JC	Approved	3583	1818715	Client with PO	2016-11-29 12:57:15.123
3	DBGurus JC	Approved	3583	1818716	DB Gurus	2016-11-29 12:57:24.820

	(No column name)
1	3

dev_SPFramework

Essential help for creating SPs that are triggered by events in TheDatabase

- input and output parameters
- UPDATE statement to hook it up

```
EXEC [dev_SPFramework] @TableID=3307 /* Times From Time Doctor */
```

100 %

Messages

```
-----  
CREATE PROCEDURE [DBGurusJC_TimesFromTimeDoctor_Add]  
(  
    @RecordID int,  
    @FormSetID int = null /* Optional */  
)  
AS  
BEGIN  
    -- Add your code here  
    -- WHERE TableID = 3307  
  
    RETURN @@ERROR  
END  
GO  
  
/* Run this to activate it: */  
UPDATE [Table] SET [AddRecordSP] = 'DBGurusJC_TimesFromTimeDoctor_Add' WHERE TableID = 3307  
GO  
  
-----  
/* Template to create a Save Stored Procedure */  
-----  
CREATE PROCEDURE [DBGurusJC_TimesFromTimeDoctor_Save]  
(  
    @RecordID int,  
    @UserID int,  
    @Return varchar(max) output  
)
```