**Stored Procedures**

Stored Procedures can be executed by an application that has access to your database. For example PHP could call for this code to execute.

Stored procedures can also execute transactions, which you cannot do with functions. Procedures however traditionally can’t return values, but there is a work around with INOUT.

Procedures also can’t be called by Select. You can execute them with EXECUTE with parameters, or with CALL. If a SP doesn’t have parameters it is called static and those with parameters are called dynamic.

CREATE OR REPLACE PROCEDURE procedure\_name(parameters)

AS

$body$

DECLARE

BEGIN

END;

$body$

LANGUAGE PLPGSQL;

-- Create a sample table that stores customer ids with balances due

CREATE TABLE past\_due(

id SERIAL PRIMARY KEY,

cust\_id INTEGER NOT NULL,

balance NUMERIC(6,2) NOT NULL);

SELECT \* FROM customer;

INSERT INTO past\_due(cust\_id, balance)

VALUES

(1, 123.45),

(2, 324.50);

SELECT \* FROM past\_due;

CREATE OR REPLACE PROCEDURE pr\_debt\_paid(

past\_due\_id int,

payment numeric

)

AS

$body$

DECLARE

BEGIN

UPDATE past\_due

SET balance = balance - payment

WHERE id = past\_due\_id;

COMMIT;

END;

$body$

LANGUAGE PLPGSQL;

-- Execute procedure

CALL pr\_debt\_paid(1, 10.00);

SELECT \* FROM past\_due;

pr\_debt\_paid(

past\_due\_id int,

payment numeric,

INOUT msg VARCHAR

);