BALANCE SHEET IMPLEMENTATION

Date	Profit($)	Loss($)	Earnings($)
2018-01-28	576	543	33
2018-02-28	675	521	154
2018-03-28	432	653	-221
2018-04-28	765	321	444

Description:

Creating A Table On A SQL Server Can Help You**. How You Ask?** Well….
A Microsoft Software Like **MS Excel Cannot Do Things Postgresql Can**. Let's Say You Wanna Create And Share Your Financial Conditions Depicted Through A Table To Someone Far Away From You, But You Don't Want To Fuss Over The Work Of Making A Copy Of The File And Sending It Through An E-Mail. You Can Just Create The Same Table In Postgresql Which Will **Automatically Update The Data On It To A Public Server**, Which Can Be Accessed Easily By Anyone With The **Server Name And Password**.
It Also Makes It Easier For People To Add The Values And Then Have Them Automatically Calculated Through A Press Of A Button.
I Bet MS Excel Doesn't Do That!
But Wait **There Is More** Did I Tell You That You Can Even Share Your Server With A Large Group Of People. This Makes Your Work Even **More Quick And Easy.**

Code:

```
CREATE TABLE table1 ("Date" date, "Profit ($)" integer, "Loss ($)" integer, "Earnings ($)" integer);

INSERT INTO public.table1 ("Date", "Profit ($)", "Loss ($)", "Earnings ($)")
        VALUES('2018-01-28', 576, 543, 0),
                ('2018-02-28', 675, 521, 0),
                ('2018-03-28', 432, 653, 0),
                ('2018-04-28', 765, 321, 0);

UPDATE public.table1
        SET "Earnings ($)"="Profit ($)"-"Loss ($)";
```

Code Explanation:

The **CREATE TABLE** statement creates a new table with the name **"table1"** and adds the columns **"Date"**, **"Profit ($)"**, **"Loss ($)"** and **"Earnings ($)"** of the defined data types.

In the newly formed **"table1"**, using the statement **INSERT INTO** new **VALUES** are added.

Finally, the **UPDATE** statement updates the values of the **"Earnings ($)"** column by subtracting the values of the **"Loss ($)"** column from the **"Profit ($)"** one.