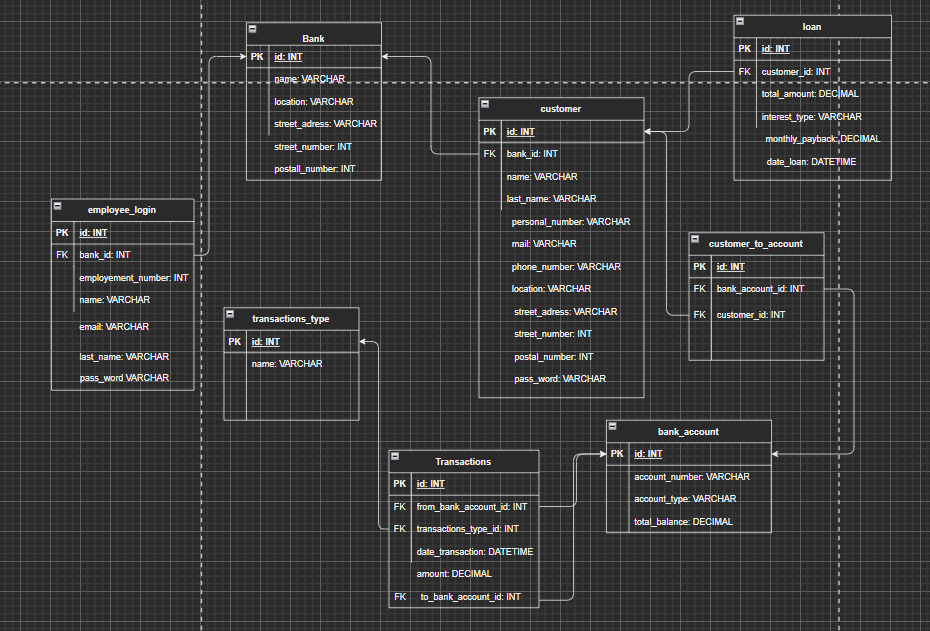
Utvärdering Svans bank

Inlämningsuppgift databas suvnet22

Av: Daniel Svan

Databas design

Kopplingsdiagram:

Databas designförklaring

Banken behöver ha kunder och anställda knutna till sig därav tabellerna employee\_login, customer och bank för att skapa denna koppling.

Bankens största inkomstkälla är lån så därav tabellen (loan). Låntagare är kunderna på banken.

Customer\_to\_account tabellen kopplar samman bank\_account och customer med varandra. Detta möjliggör att fler kunder kan dela på samma konto.

För att pengar ska kunna överföras till och ifrån kunder så behöver kunderna ett bankkonto.

För att hålla kolla på hur, var och när pengarna skickas så kopplas bank\_account till transactions, en transaktion görs från ett bankkonto. Varje transaktion har en typ, därav transactions\_type.

Extra (SQL):

Count funktionen räknar antalet transaktioner som utförts på banken.

Stored procedure operation använder jag mig utav i transaktions syfte, och hur den används förklaras i delen nedan.

Problem

Transaktioner (internal\_transaction):

För att göra transaktioner så vill man att allt sker på en och samma gång. För det krävs att allt ska funka i samflöde. Därav skapar jag en transaction query så att när en kund vill göra en transaktion så utförs den komplett eller inte alls. Inuti denna transaction query behöver olika operationer bli gjorda korrekt så att det kan utföras sömlöst. Alla dessa uppgifter som ska ske är lagrade i databasen, därav skapade jag en stored procedure. Vad som sker där är att vid en ”internal\_transaction” så måsta pengar försvinna först och eftersom det är i emellan egna konton så måste pengar som försvunnit läggas till direkt. När kunden ska överföra pengar så måste de ske sömlöst fungerande eller så sker inget alls. Därav valde jag att göra hela processen i databas i stället för att göra lite i databas och lite som programfunktion.

Registrering av kunder:

Att registrera kunder är en standardprocess, men som bank så behövs det att kunder ska kunna sätta in pengar och börja betala diverse utgifter direkt. För att då slippa processen om att kund ska ansöka och vänta på att ett konto ska skapas efter att kund registrerats, så valde jag att skapa ett konto via en transaction query ihop med registreringen av kund. Detta löser problemet med att slösa resurser på att efterbehandla kunder. Kunder kan nu sömlöst efter registrering sätta in pengar och börja använda sig av bankens mest basiska funktioner.

Utveckling:

För vidare utveckling skulle jag vilja bygga upp programmet för anställda, lån samt för kunder att ansöka om lån. Detta är ej med nu då jag placera det i sin egen lilla bubbla som har ett eget kapitel (lån).

Lån funktion är en operation där kunderna ska fylla i en rad information för att ansöka om lån. Denna ansökan ska skickas till de anställda för granskning. Det är den enklaste utvecklingen som skulle kunna göras.

Ett exempel på hur man kan utveckla det i databasen är att lägga till en bool is\_reviewed samt bool is\_accepted på loan-tabellen. När den är granskad så sätts den till true. På så sätt sparas även de låneansökningar som ej godkänts samt som ej är kollade än.