

```

Create table Clientes_Prestamo (
    Numero_prestamo varchar2(10) not null ,
    Fecha_aprob date not null,
    Monto_aprob number,
    Letra_mensual number not null,
    Cod_Prestamo,
    Constraint codigo_prestamo foreign key (Cod_Prestamo) references
    Prestamo_Tipo(Cod_Prestamo),
    ID_cliente,
    Constraint identif_cliente_prestamo foreign key (ID_Cliente) references
    Banco_Clientes (ID_Cliente),
    Constraint Presta_Cliente primary key (Cod_Prestamo , ID_Cliente)
);

create table SUCURSAL_PRESTAMO(
    Cod_sucursal number not null,
    ID_tipo number not null,
    Constraint ID_tipo_suc_fk foreign key (ID_tipo) references TIPO_PRESTAMO
    (ID_tipo),
    Constraint Cod_sucursal_p_fk foreign key (Cod_sucursal) references SUCURSAL
    (Cod_sucursal),
    Constraint Cod_id_suc_pk primary key (Cod_sucursal, ID_tipo),
    Monto_prestamo number not null
);

create table SUCURSAL (
    Cod_sucursal number not null,
    Nombre_sucursal varchar2 (30) not null,
    Constraint Cod_suc_pk primary key (Cod_sucursal)
);

```

Creacion de la tabla Prestamo\_Tipo

```
Create table Prestamo_Tipo (  
Cod_Prestamo varchar2(15) not null primary key,  
Nombre_prestamo varchar2(50) not null,  
Tasas_Interes number not null  
);
```

Create table Suc\_Tipo\_Prestamo (

```
Cod_Sucursal varchar2(5),  
Constraint Fk_cod_suc foreign key (Cod_Sucursal) references Sucursal  
(Cod_Sucursal),  
Cod_Prestamo varchar2(15),  
Constraint Fk_cod_prestamo foreign key (Cod_Prestamo) references  
Prestamo_Tipo (Cod_Prestamo),  
Monto_Prestamos number(15,2),  
Constraint pk_tipo_suc_prest primary key (Cod_Sucursal, Cod_Prestamo)  
);
```

```
Create table Transac_pagos (  
ID_Transaccion number primary key not null,  
Cod_Sucursal varchar2(5) not null,  
Monto_pago number not null,  
Fecha_transac date not null,  
Fecha_inserc date not null,  
Usuario number not null,  
Constraint FK_CodSuc_TransP foreign key (Cod_Sucursal) references Sucursal  
(Cod_Sucursal),  
ID_Cliente number,  
Cod_Prestamo varchar2(15),  
Constraint FK_IDClient_TPrest_TransP foreign key (ID_Cliente, Cod_Prestamo)  
references Clientes_Prestamo (ID_Cliente, Cod_Prestamo)  
);
```

```
Create table TEMP_Auditoria (  
Numero_prestamo varchar2(10) not null ,  
Fecha_aprob date not null,  
Monto_aprob number,  
Letra_mensual number not null,  
Fecha_audi date,  
Usuario varchar2(20),  
Cod_Prestamo,  
Constraint codigo_prestamo foreign key (Cod_Prestamo) references  
Prestamo_Tipo(Cod_Prestamo),  
ID_cliente,  
Constraint identif_cliente_prestamo foreign key (ID_Cliente) references  
Banco_Clientes (ID_Cliente),  
Constraint Presta_Cliente primary key (Cod_Prestamo , ID_Cliente)  
);
```

**\*\*trigger que guarda la transacción de créditos o prestamos aprobados-----**

CREATE OR REPLACE TRIGGER PRES\_APROB  
BEFORE OF INSERT ON Clientes\_PRESTRAMO FOR EACH ROW

BEGIN

INSERT INTO temp\_auditoria (REG,AUDITORIA, Numero\_prestamo,  
Fecha\_aprob, Monto\_aprob,

Letra\_mensual ,Cod\_Pestamo,Id\_clintes)

VALUES ( seq\_reg\_auditoria.NEXTVAL,:new. Numero\_prestamo,:new.  
Fecha\_aprob, :new. Monto\_aprob, Letra\_mensual, new.  
Cod\_Pestamo,new.ID\_cliente);

--END IF;

END PRES\_APROB;

/

**\*\*-----**

CREATE OR REPLACE TRIGGER SUC\_TIP\_PRES  
BEFORE OF INSERT ON CLIENTES\_PRESTAMOS FOR EACH ROW

BEGIN

INSERT INTO temp\_auditoria (cod\_sucursal,cod\_prestamo,cod  
Monto\_aprob)

VALUES (new. cod\_sucursal,:new. cod\_prestamo, :new. Monto\_aprob,  
Letra\_mensual, new. Cod\_Pestamo);

--END IF;

END SUC\_TIP\_PRES;

/

```

CREATE OR REPLACE TRIGGER TRANSAC_PREST
BEFORE UPDATE OR DELETE OR INSERT ON transac_pagos
FOR EACH ROW
DECLARE
    v_TipoCambio CHAR(1);
BEGIN
    IF UPDATING THEN
        v_TipoCambio := 'A';
        INSERT INTO TEMP_AUDITORIA
        VALUES (seq_reg_auditoria.NEXTVAL,:new. Numero_prestamo,:new.
        Fecha_aprob, :new. Monto_aprob, Letra_mensual, new.
        Cod_Pestamo,new.ID_cliente);
    ELSIF INSERTING THEN
        v_TipoCambio := 'B';
        INSERT INTO TEMP_AUDITORIA
        VALUES seq_reg_auditoria.NEXTVAL,:new. Numero_prestamo,:new.
        Fecha_aprob, :new. Monto_aprob, Letra_mensual, new.
        Cod_Pestamo,new.ID_cliente);
    ELSIF DELETING THEN
        v_TipoCambio := 'C';
        INSERT INTO TEMP_AUDITORIA
        VALUES (seq_reg_auditoria.NEXTVAL,:new. Numero_prestamo,:new.
        Fecha_aprob, :new. Monto_aprob, Letra_mensual, new.
        Cod_Pestamo,new.ID_cliente);
    END IF;
END TRANSAC_PREST;
/

```

```

CREATE OR REPLACE TRIGGER PAGOS_SUCUR
BEFORE UPDATE OR DELETE OR INSERT ON transac_pagos
FOR EACH ROW
DECLARE
    v_TipoCambio CHAR(1);
BEGIN
    IF UPDATING THEN
        v_TipoCambio := 'A';
        INSERT INTO SUC_TIPO_PRESTAMOS
        VALUES (new. cod_sucursal,:new. cod_prestamo, :new. Monto_aprob);
    ELSIF INSERTING THEN
        v_TipoCambio := 'B';
        INSERT INTO SUC_TIPO_PRESTAMOS
        VALUES new.( cod_sucursal,:new. cod_prestamo, :new. Monto_aprob,
        ELSIF DELETING THEN
        v_TipoCambio := 'C';
        INSERT INTO SUC_TIPO_PRESTAMOS
        VALUES (new. cod_sucursal,:new. cod_prestamo, :new. Monto_aprob,);
    END IF;
END TRANSAC_PREST;
/
-----

CREATE VIEW SUC_TIPO As
Select NOM_SUCURSAL,MONTO From SUCURSAL S JOIN

```