

IST 659

Course Project- Final Report

MILITARY PENSION PAYROLL DATA BASE MANAGEMENT SYSTEM

Shreyas Raghavan Sadagopan

MS ADS

509366439

TABLE OF CONTENTS

1. INTRODUCTION AND SUMMARY	3
2. TABLES and ENTITIES: Glossary	4
3. ERD	6
I. VISIO	
II. MS ACCESS	
4. BUSINESS RULES AND RESTRICTIONS	7
5. DATA USERS, FUNCTIONS AND QUESTIONS	8
6. SQL QUERIES	8
I. CREATING TABLE	
II. CREATING TRIGGERS	
III. INSERT INTO	
IV. PAYSLIP GENERATOR: MULTI-TABLE TRANSACTION	
7. Queries to Answer Major Data Questions	14
8. INTERFACES	15
I. FORMS	
II. TRIGGERS	
9. TRIGGERS: Testing and Demo	21
I. Trigger 1	
II. Trigger 2	
III. Trigger Demo in Access	

1. SUMMARY

The Military Veterans Pension Payroll Institute (MVPPI) is a government funded organization (hypothetical) and is a subsidiary to the Military Officers Payroll Institute (MOPI), maintains paperwork containing all the payroll information of Ex-Servicemen/Women. Their job is to ensure timely delivery of the pension and other allowances to the veterans or their nominees (For Martyrs).

The MOPI and the MVPPI act as separate entities. For an organization like the MVPPI it is imperative to have a dynamic, well connected and electronic payroll data base. As the MOPI data bases and the MVPPI are not linked, the officers once retiring have to enroll themselves for the pension scheme. The system in place has a sense of ambiguity attached as clerical errors are bound to happen when accounts are handled manually. Handling confidential (SSN and Bank details) data about the nation's veterans should be done securely. The current paper-based system is totally insecure and calls for a digital framework with encryptions. The current process a veteran must undertake to change banking and personal details, would take weeks to be processed. Most veterans and their families would not be able to undergo this ordeal every time they request a change.

The payroll system should be dynamic and be able to account for changes in costs of living and lifestyle. This would not be possible if on a monthly basis, tons of paperwork need to be processed.

Main functions considered in the proposed system will be:

- 1) A digital framework will be created. All the paperwork will be digitalized.
- 2) The details of a serviceman once retired /Dead/Martyred would be updated to the MVPPI data base from the MOPI data base by the HR department. The MOPI contains information about all current servicemen. The MVPPI has veteran/ex-servicemen details.
- 3) The HR department and the Accounts department of the MVPPI will be connected. The Accounts department is responsible for the following:
 - I. Accessing veteran details from the MVPPI data base
 - II. Accessing and calculating the pension and allowance Slabs
 - III. Evaluating Nominee details
 - IV. Evaluating Bank details
 - V. Creating monthly Pay Slips
 - VI. Ensuring timely disbursement of the pension to the rightful veteran or his/her nominee
- 4) Inter organization data base access: The government's finance office updates the pension and the allowance slabs. The MVPPI's accounts dept does not have write access to these two data sets.
- 5) The MOPI acts as the bridge between the veterans and the MVPPI to enable users to request changes. The user has to log in to his account with the MOPI / or reach out to the nearest MOPI office to request changes.
- 6) The MOPI has access to the MVPPI data base and will be able to make any changes in the personal details when requested by a veteran or their nominees.
- 7) The veterans: They can request any changes only to the following tables(online)
 - Nominees
 - Bank details
 - PaySlip (View access only. Limited to the user's account)
- 8) Encrypting confidential information and ensuring data security (Not Included)

2. TABLES AND ENTITIES: Glossary

MOPI : The MOPI's HR and Accts Dept. Read and Write access for both depts.

MVPPI_Veterans : The MVPPI's HR and Acct Depts. Automatically updated for every officer who matches the criteria to be a veteran (Retired or Martyred).

Pension Slab : The Government's Finance Department , MOPI and MVPPI. Only Read access for the MVPPI's HR and Accounts depts and write for the govt.

Allowance Slab : The Government's Finance Department, MOPI and MVPPI . Only Read access for the MVPPI's HR and Accounts depts. Read and write for the govt.

Nominee Details : Contains Nominee details for all the servicemen in MOPI data base. Read and write access to MOPI. Restricted Read and write access to the Servicemen(Only their records).

Bank Details : Contains bank details for all Servicemen and Nominees. Read and write access to MOPI. Restricted Read and write access to the Servicemen(Only their records)

Payslip : Gathers information from the MVPPI, Pension Slab, Allowance Slab , Nominees and bank details to calculate monthly Pension Amount to be disbursed. Read access to MVPPI . Restricted read access to the user (Only their payslip).

PRIMARY KEY **FOREIGN KEY**

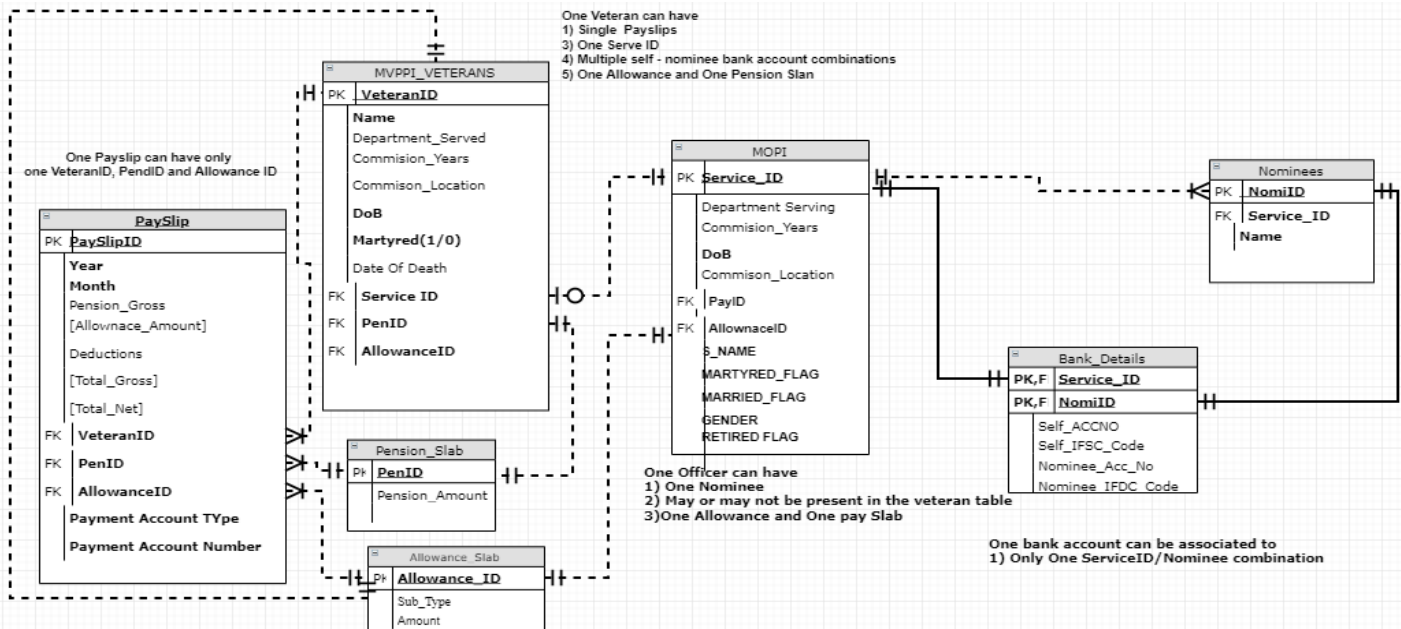
Entity Name	Attribute Name	Description	Domain Constraint
<u>MVPPI VETERANS</u>	Veteran_ID	Unique Identifier	Unique, Not Null
	Name	Full Name of the Veteran	Not null
	Department_Served	Department Served before retiring/martyr	
	Commission_years	Number of years served	
	Commission_location	String (Can Contain Multiple Locations) separted by commas	
	DoB	Date of Birth	Required
	DoD	Date of Death (Can be NULL)	
	Martyred (1 or 0)	Flag to tell whether Martyred or not.) mean Retired	Required
	ServiceID	Ex – Serviceman's ServiceID	Unique
	PenID	Pension Slab identifier	Required
	AllowanceID	Allowance Slab Identifier	Required
<u>MOPI</u>	Service_ID	Unique identifier	Unique Not Null
	Department Serving	Department Serving	
	S-Name	Serviceman_Name	Required

	Commission_years	Number of years currently Serving/Served	
	Commission_location	String (Can Contain Multiple Locations) sepearted by commas	
	DoB	Date of Birth	Required
	PayId	Pay Slab Identifier (Not included in our system)	Required
	AllowanceId	Allowance Slab Identifier	Required
	Retirement/ Martyr Date	Date of retirement (Can be Null)	Required
	Retired Flag	1/0 or zero flag for MVPPI update	Required
	Martyr Flag	1/0 for Martyr status	Required
	Married Flag	1/0 for martial status	Required
	Gender	1- Male 0-Female	Required
<u>Nominees</u>	Service_ID	Veteran's Service ID	Unique Not Null
	NomiID	Veternan's Nominee ID	Unique
	Name	Nominee Name	Required
<u>Pension Slab</u>	PenID	Pension Slab Identifier	Unique Not Null
	Pension_Amount	Pension Amount	Required
<u>Allowance slab</u>	Allowance_ID	Allownace Slab Identifier	Unique Not Null
	Sub_types	Type Of Allowance: Housing, Clothing etc	Required
	Amount	Prescribed Amount	Required
<u>Bank Details</u>	ServiceID	Service ID	Unique Not Null
	NomID	Nominee ID	Unique Not Null
	Self_acc_no	Veteran's Acc Number	Required
	Nominee_acc_no	Nominee's Acc Number	Required
	Self_IFSC_Code	Veteran's routing Number	Required

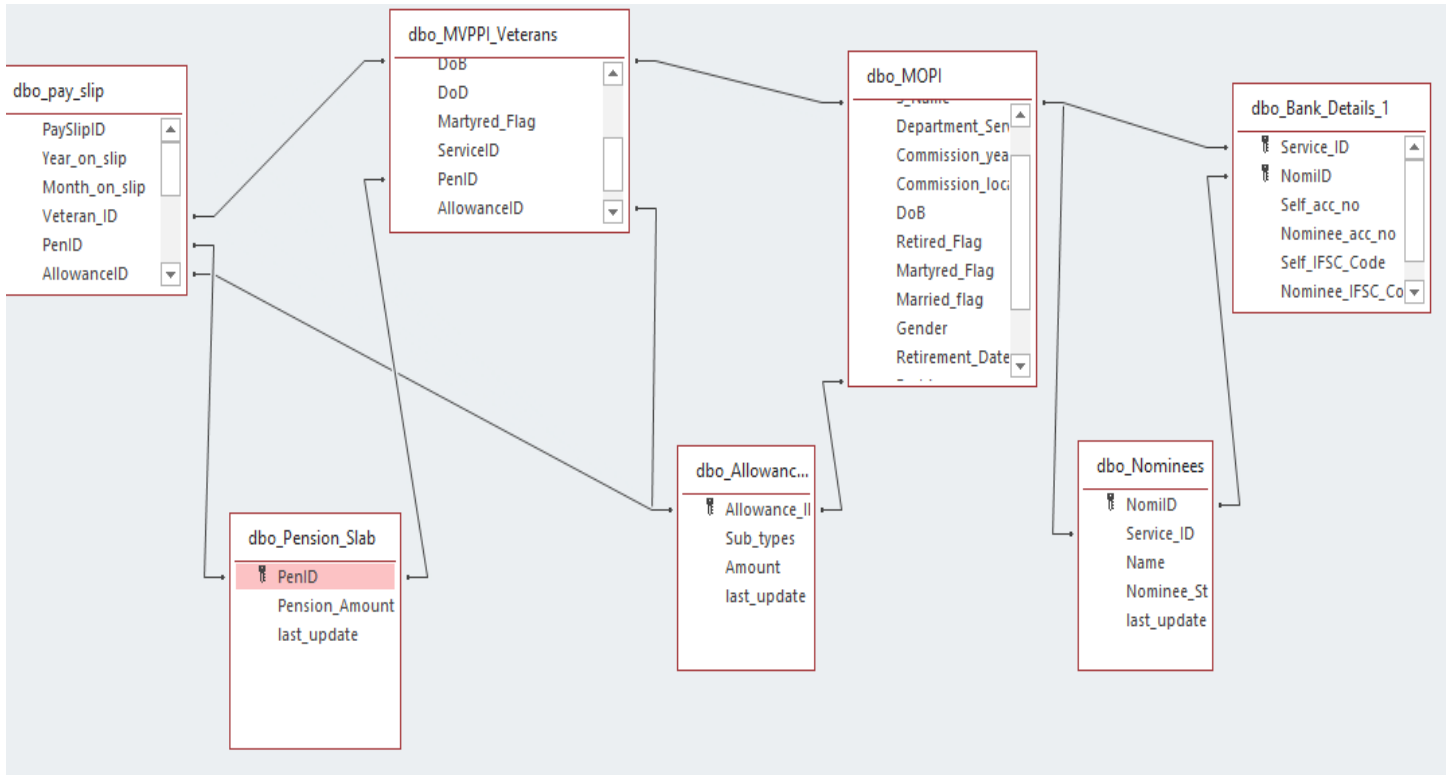
	Nominee_IFSC_Code	Nominee's routing Number	Required
<u>PaySlip</u>	PaySlipID	Unique Identifier	Unique Not Null
	Year		Required
	Month		Required
	VeteranID	Veteran ID	Required
	PenID	Pension ID	Required
	AllownaceID	Allowance ID	Required
	Pension_Gross	Gross Pension Amount	
	Allowance Amount		
	Deductions	Deductions	
	Total_gross		Required
	Total_Net		Required
	Payment Account Type (1/0)	Nominee/ Self	Required
	Payment Account Number	Account Number	Required

3. Entity Relationship Diagrams

I. VISIO :



II. ERD from MS Access:



4. Business Rules and Restrictions

- 1) One Person can be only a nominee to just on veteran/officer
- 2) A veteran can have a maximum of 1 nominee.
- 3) Allowance Subtype can be combinations of the following;
 - Cadre 1
 - Cadre 2
 - Cadre 3
 - Cadre 4
 - Cadre 5
 - Cadre 6
 - Cadre 7
 - Unmarried Martyred + Cadre(1-3)
 - Unmarried Martyred + Cadre(4-7)
 - Widowed_Nominee+Cadre(1-3)
 - Widowed_Nominee+Cadre(4-7)
 - Widowed_Nominee+Cadre(1-3)
 - Widowed_Nominee+Cadre(4-7)
- 4) Nominee table's Activity Status can take only 3 values : 1, 2 or 3
- 5) Payment Account type can take only 2 values 0 or 1
- 6) The Accounts dept does not have write access to the pension slab and allowance slab tables

5. Data Users, Functions and Questions

The major users of this data base are the following:

- 1) MVPPI's HR Department
- 2) MVPPI's Accounts Department
- 3) MOPI's HR Department
- 4) The Government's Finance Department
- 5) The USERS (Veterans/Nominees)

The Major Functions Performed:

- 1) To get the details of and to check for new retirees/Martyrs from the MOPI Data base and update the MVPPI data base.
- 2) Maintain and update Nominee & Banking information
 - I. To maintain and update the nominee & Banking tables for all the Officers in the MOPI Table
 - II. To update the nominee & Banking tables when there is a change requested by the veteran through the MOPI online portal.
- 3) Accessing the Pension Slabs and the Allowance Slabs to calculate the Pension Amount
- 4) Generating Payslips by calculating the Total amount to be disbursed and ensuring disbursement to the rightful recipients(Veterans/Nominees) and the right accounts.

The Top Data Questions:

- 1) To maintain a record of number of personnel serving in each of the departments
- 2) To Calculate the Pension Amount, Allowance amount and the Total net pay after deductions for each of the Veterans for a given month.
- 3) To maintain a record of all the Payslips issued for all veterans.
- 4) To calculate the total pension drawn till date (Sum total of all Payslips) for all the veterans.
- 5) To calculate number of payslips issued to nominees and issued to veterans.
- 6) To keep track of the number of officers Martyred in each of the Post Locations (Countries).

6. SQL SCRIPTS

I. Create Table Queries

```
--drop table pay_slip_all
--drop table Pay_Slip
--drop table Bank_Details
--drop table MVPPI_Veterans
--drop table Nominees
--drop table MOPI
--drop table Pension_Slab
--drop table Allowance_Slab
```

```
CREATE TABLE Pension_Slab (
PenID NUMERIC(5,0) PRIMARY KEY,
Pension_Amount int NOT NULL,
last_update date
```



```

);

CREATE TABLE Allowance_Slab (
Allowance_ID NUMERIC(5,0) PRIMARY KEY,
Sub_types VARCHAR(300) NOT NULL,
Amount int not null,
last_update date
);

CREATE TABLE MOPI(
Service_ID NUMERIC(5,0) PRIMARY KEY,
S_Name VARCHAR(30) NOT NULL,
Department_Serving VARCHAR(30) NOT NULL,
Commission_years NUMERIC(2,0) NOT NULL,
Commission_location VARCHAR(20) NOT NULL,
DoB date not null,
Retired_Flag NUMERIC(1,0) NOT NULL,
Martyred_Flag NUMERIC(1,0) NOT NULL,
Married_flag Numeric(1,0) NOT NULL,
Gender Numeric(1,0) NOT NULL,
Retirement_DateorMartyr_Date Date ,
PayId NUMERIC(5,0) ,
AllowanceID NUMERIC(5,0) FOREIGN KEY REFERENCES Allowance_Slab( Allowance_ID),
last_update date
);

CREATE TABLE MVPPI_Veterans(
Veteran_ID NUMERIC(5,0) PRIMARY KEY,
Name VARCHAR(30) NOT NULL,
Department_Served VARCHAR(30) NOT NULL,
Commission_years NUMERIC(2,0) NOT NULL,
Commission_location VARCHAR(20) NOT NULL,
DoB date not null,
DoD date ,
Martyred_Flag NUMERIC(1,0) not null,
ServiceID NUMERIC(5,0) FOREIGN KEY REFERENCES MOPI(Service_ID),
PenID NUMERIC(5,0) FOREIGN KEY REFERENCES Pension_slab (PenID),
AllowanceID NUMERIC(5,0) FOREIGN KEY REFERENCES Allowance_Slab( Allowance_ID),
last_update date
);

CREATE TABLE Nominees (
NomiID NUMERIC(5,0) PRIMARY KEY,
Service_ID NUMERIC(5,0) FOREIGN KEY REFERENCES MOPI(Service_ID),
Name VARCHAR(30) NOT NULL,
last_update date
);

CREATE TABLE Bank_Details(
Service_ID NUMERIC(5,0) FOREIGN KEY REFERENCES MOPI(Service_ID),
NomiID NUMERIC(5,0) FOREIGN KEY REFERENCES Nominees(NomiId),
Self_acc_no NUMERIC(9,0) NOT NULL,
Nominee_acc_no NUMERIC(9,0) NOT NULL,
Self_IFSC_Code VARCHAR(10) NOT NULL,
Nominee_IFSC_Code VARCHAR(10) NOT NULL,
last_update date
CONSTRAINT Bank_Details_PK PRIMARY KEY (Service_Id, NomiID)
);

CREATE TABLE [dbo].[pay_slip](

```

```

[PaySlipID] [bigint] PRIMARY KEY,
[Year_on_slip] [int] Not Null,
[Month_on_slip] [int] NULL,
[Veteran_ID] [numeric](5, 0) NOT NULL,
[PenID] [numeric](5, 0) NULL,
[AllowanceID] [numeric](5, 0) NULL,
[Pension_Gross] [int] NULL,
[Allowance_Amount] [int] NULL,
[Total_Gross] [int] NULL,
[Deductions] [numeric](12, 1) NULL,
[Total_net] [numeric](12, 1) NULL,
[Payment_account_type] [varchar](15) NOT NULL,
[Payment_Account_Number] [numeric](9, 0) NULL
) ON [PRIMARY]
GO

```

II. CREATE TRIGGERS

In this DBMS, Triggers are used to automate the process of populating tables and data transfer and update from one table to the other. So, the triggers need to be created before the 'Insert Into Commands are executed'

Trigger 1:

This trigger is created to push the Serviceman being inserted into the MOPI (Officers Database) into the ' MVPPI (Veterans's Data Base) if the retired_flag=1 or martyred_flag =1 for the incoming serviceman.

```

drop trigger trigger1
create trigger trigger1
on MOPI
for insert --triggered when a new entry is made in the MOPI data base
as
if @@ROWCOUNT>=1
begin
Insert into MVPPI_Veterans
select i.Service_ID, i.S_name
,i.Department_Serving,i.Commission_years,i.Commission_location,i.DoB,
case when i.Martyred_flag = 1 then i.Retirement_DateorMartyr_Date else NULL end,
i.Martyred_flag, i.Service_ID,
case when i.Commission_years > 30 then 10007
when i.Commission_years <=30 and i.Commission_years > 25 then 10006
when i.Commission_years <=25 and i.Commission_years > 20 then 10005
when i.Commission_years <=20 and i.Commission_years > 15 then 10004
when i.Commission_years <=15 and i.Commission_years > 10 then 10003
when i.Commission_years <=10 and i.Commission_years > 5 then 10002
else 10001 end,
i.AllowanceID,i.last_update
from inserted i
where i.retired_flag=1 or i. martyred_flag =1
end

```

Trigger 2:

This trigger is created to push a particular Serviceman present in the MOPI (Officers Database) into the ' MVPPI (Veterans's Data Base) when he is retired or martyred. The triggered is executed when the retirement date or martyr date of a serviceman is updated.

```

drop trigger trigger2
create trigger trigger2
on MOPI

```

```

for update
as
if update(Retirement_DateorMartyr_Date)
begin
delete MVPPI_Veterans
insert into MVPPI_Veterans
select i.Service_ID, i.S_name
,i.Department_Serving,i.Commission_years,i.Commission_location,i.DoB,
case when i.Martyred_flag = 1 then i.Retirement_DateorMartyr_Date else NULL end,
i.Martyred_flag, i.Service_ID,
case when i.Commission_years > 30 then 10007
when i.Commission_years <=30 and i.Commission_years > 25 then 10006
when i.Commission_years <=25 and i.Commission_years > 20 then 10005
when i.Commission_years <=20 and i.Commission_years > 15 then 10004
when i.Commission_years <=15 and i.Commission_years > 10 then 10003
when i.Commission_years <=10 and i.Commission_years > 5 then 10002
else 10001 end,
i.AllowanceID,i.last_update
from MOPI i
where i.retired_flag=1 or i. martyred_flag =1
end

```

III. 'Insert Into' Queries:

--Pension slab table

```

INSERT INTO Pension_Slab VALUES (10001, 4500, getdate()); --getdate function returns current date
INSERT INTO Pension_Slab VALUES (10002, 2100, getdate() );
INSERT INTO Pension_Slab VALUES (10003,2200, getdate() );
INSERT INTO Pension_Slab VALUES (10004,2500, getdate() );
INSERT INTO Pension_Slab VALUES (10005,2500, getdate() );
INSERT INTO Pension_Slab VALUES (10006,3000, getdate() );
INSERT INTO Pension_Slab VALUES (10007,4500, getdate() );

```

--Allowance slab table

```

INSERT INTO Allowance_Slab VALUES (1, 'Cadre 1', 50, getdate() );
INSERT INTO Allowance_Slab VALUES (2, 'Cadre 2', 60, getdate() );
INSERT INTO Allowance_Slab VALUES (3, 'Cadre 3', 70, getdate() );
INSERT INTO Allowance_Slab VALUES (4, 'Cadre 4', 80, getdate() );
INSERT INTO Allowance_Slab VALUES (5, 'Cadre 5', 90, getdate() );
INSERT INTO Allowance_Slab VALUES (6, 'Cadre 6', 100, getdate() );
INSERT INTO Allowance_Slab VALUES (7, 'Cadre 7', 200, getdate() );
INSERT INTO Allowance_Slab VALUES (8, 'Unmarried Martyred + Cadre(1-3)', 300, getdate() );
INSERT INTO Allowance_Slab VALUES (9, 'Unmarried Martyred + Cadre(4-7)', 350, getdate() );
INSERT INTO Allowance_Slab VALUES (10, 'Widowed Nominee+Cadre(1-3)', 500, getdate() );
INSERT INTO Allowance_Slab VALUES (11, 'Widowed Nominee+Cadre(4-7)', 700, getdate() );
INSERT INTO Allowance_Slab VALUES (12, 'Widowed Nominee+Cadre(1-3)', 450, getdate() );
INSERT INTO Allowance_Slab VALUES (13, 'Widowed Nominee+Cadre(4-7)', 650, getdate() );

```

--Service men table (MOPI)

```

INSERT INTO MOPI VALUES (1101, 'Mark Butler', 'Medical', 23, 'Afghanistan', '06-08-1978', 1, 0, 1, 1, '8-03-2009', 1007, 7, getdate() ); --Retired
INSERT INTO MOPI VALUES (1102, 'Lucas Giligan', 'Medical', 20, 'Afghanistan', '08-09-1980', 1, 0, 1, 1, '09-17-2009', 1007, 6, getdate() ); --Retired
INSERT INTO MOPI VALUES (1103, 'Jose Martinez', 'Medical', 16, 'Afghanistan', '11-12-1988', 0, 0, 1, 1, NULL, 1006, 6, getdate() );
INSERT INTO MOPI VALUES (1104, 'Paul Rutger', 'Medical', 16, 'Afghanistan', '10-14-1985', 0, 0, 0, 1, NULL, 1007, 5, getdate() );

```

```

INSERT INTO MOPI VALUES (1201,'Jade Scorsesse','Stratergy',32,'Afghanistan','03-11-1965',1,0,0,1,'03-11-2015',1009,7, getdate() ); --Reitred
INSERT INTO MOPI VALUES (1202,'Marcus North','Stratergy',20,'Afghanistan','11-07-1980',0,0,0,1,NULL,1006,6, getdate() );
INSERT INTO MOPI VALUES (1203,'Joe Lee','Stratergy',12,'Afghanistan','01-18-1988',0,0,0,0,NULL,1007,3, getdate() );
INSERT INTO MOPI VALUES (1204,'Drake Vincent','Stratergy',2,'Afghanistan','01-05-1995',0,1,0,1,'05-06-2014',1002,8, getdate() ); --Martyred
INSERT INTO MOPI VALUES (1301,'Priyanka Talpade','Armory',35,'Afghanistan','11-13-1962',1,0,1,1,'11-13-2017',1009,7, getdate() ); --Reitred
INSERT INTO MOPI VALUES (1302,'Blake Anderson','Armory',30,'Afghanistan','05-31-1968',0,0,1,1,NULL,1007,7, getdate() );
INSERT INTO MOPI VALUES (1303,'Kiara Affleck','Armory',15,'Afghanistan','10-02-1986',0,1,1,0,'01-20-2015',1007,13, getdate() ); --Martyred
INSERT INTO MOPI VALUES (1304,'Adam Kapur','Armory',10,'Afghanistan','01-05-1995',0,1,1,1,'05-06-2014',1002,11, getdate() ); --Martyred
INSERT INTO MOPI VALUES (2101,'Girisha Matunga','Medical',35,'Iraq','06-08-1967',0,1,1,0,'12-01-2019',1009,13, getdate() ); --Martyred
INSERT INTO MOPI VALUES (2102,'Elliot Greene','Medical',20,'Iraq','05-18-1980',0,0,1,0,NULL,1007,6, getdate() );
INSERT INTO MOPI VALUES (2103,'Narcissa Malfoy','Medical',30,'Iraq','05-31-1968',0,1,1,0,'1-20-2012',1006,13, getdate() ); --Martyred
INSERT INTO MOPI VALUES (2104,'Adam Williams','Medical',30,'Iraq','02-14-1973',0,0,1,1,NULL,1007,7, getdate() );
INSERT INTO MOPI VALUES (2201,'Mahesh Kaushal','Stratergy',32,'Iraq','03-11-1965',1,0,1,1,'03-11-2015',1009,7, getdate() ); --Reitred
INSERT INTO MOPI VALUES (2202,'Eric Makakwa','Stratergy',20,'Iraq','08-09-1981',0,0,0,1,NULL,1006,6, getdate() );
INSERT INTO MOPI VALUES (2203,'Jim Beesley','Stratergy',22,'Iraq','01-18-1988',0,1,1,0,'05-06-2016',1007,11, getdate() ); --Martyred
INSERT INTO MOPI VALUES (2204,'Lucas Mared','Stratergy',5,'Iraq','01-05-1995',0,0,1,1,NULL,1002,2, getdate() );
INSERT INTO MOPI VALUES (2301,'Alfred Ebone','Armory',12,'Iraq','11-14-1992',0,0,0,0,null,1004,5, getdate() );
INSERT INTO MOPI VALUES (2302,'Albert Black Potter','Armory',34,'Iraq','05-31-1968',1,0,1,1,'01-20-2015',1009,7, getdate() ); --Retired
INSERT INTO MOPI VALUES (2303,'Timothy Halpet','Armory',3,'Iraq','11-12-1998',0,0,1,1,NULL,1007,3, getdate() );
INSERT INTO MOPI VALUES (2304,'Paul Murphy','Armory',5,'Iraq','4-13-1997',0,1,0,1,'05-02-2019',1003,9, getdate() ); --Martyred

--Nominee Table
INSERT INTO Nominees VALUES (10011, 1301, 'Shreyas Talpade',getdate() );
INSERT INTO Nominees VALUES (10021, 1201, 'Michael Scorsesse', getdate() );
INSERT INTO Nominees VALUES (10031, 2201, 'Vicky kaushal', getdate() );
INSERT INTO Nominees VALUES (10041, 2101, 'Vignesh Matunga', getdate() );
INSERT INTO Nominees VALUES (10051, 1302, 'John Karsinki', getdate() );
INSERT INTO Nominees VALUES (10061, 2103, 'Draco Malfoy', getdate() );
INSERT INTO Nominees VALUES (10071, 2302, 'Harry Potter', getdate() );
INSERT INTO Nominees VALUES (10081, 2104, 'Kevin Malone',getdate() );
INSERT INTO Nominees VALUES (10091, 1101, 'Gerard Butler',getdate() );
INSERT INTO Nominees VALUES (10101, 2102, 'Ayerton Senna', getdate() );
INSERT INTO Nominees VALUES (10111, 1102, 'Big Show',getdate() );
INSERT INTO Nominees VALUES (10121, 1202, 'Walter White',getdate() );
INSERT INTO Nominees VALUES (10131, 2202, 'Jesse Pinkman', getdate() );
INSERT INTO Nominees VALUES (10141, 1104, 'Alan Donald', getdate() );
INSERT INTO Nominees VALUES (10151, 1303, 'Ben Affleck', getdate() );
INSERT INTO Nominees VALUES (10161, 1203, 'kevin Spacey', getdate() );
INSERT INTO Nominees VALUES (10171, 2203, 'Pam Beesley', getdate() );
INSERT INTO Nominees VALUES (10181, 1103, 'Frank Underwood',getdate() );
INSERT INTO Nominees VALUES (10191, 2301, 'Michael Hussey',getdate() );
INSERT INTO Nominees VALUES (10201, 2204, 'Gereard Pique', getdate() );

```

```

INSERT INTO Nominees VALUES (10211, 1204, 'Lou Vincent',getdate() );
INSERT INTO Nominees VALUES (10221, 1304, 'Kelly kapur', getdate() );
INSERT INTO Nominees VALUES (10231, 2304, 'Ted Murphy',getdate() );
INSERT INTO Nominees VALUES (10241, 2303, 'Ned Stark',getdate() );

--Bank Details table
INSERT INTO Bank_Details VALUES (1301,10011,123456,42342,23423,54545, getdate() );
INSERT INTO Bank_Details VALUES (2201,10031,433420,39234,4324324,435, getdate() );
INSERT INTO Bank_Details VALUES (1201,10021,1232334,2345,13434,1331, getdate() );
INSERT INTO Bank_Details VALUES (2101,10041,1234256,452342,2234243,4524, getdate() );
INSERT INTO Bank_Details VALUES (1302,10051,432432,45256442,762234243,4587624, getdate() );
INSERT INTO Bank_Details VALUES (2103,10061,1232334,2345,13434,1331, getdate() );
INSERT INTO Bank_Details VALUES (2104,10081,65436,5876442,76223,45874, getdate() );
INSERT INTO Bank_Details VALUES (2302,10071,2321,546546,23321,43242, getdate() );
INSERT INTO Bank_Details VALUES (1101,10091,463420,3954,42345,65434, getdate() );
INSERT INTO Bank_Details VALUES (2102,10101,4420,954,4345,6524, getdate() );
INSERT INTO Bank_Details VALUES (1102,10111,23451,86754,8767,87987, getdate() );
INSERT INTO Bank_Details VALUES (1202,10121,23451,8674,867,877, getdate() );
INSERT INTO Bank_Details VALUES (2202,10131,251,989,008,78977, getdate() );
INSERT INTO Bank_Details VALUES (1104,10141,51,99,908,2344, getdate() );
INSERT INTO Bank_Details VALUES (1303,10151,7671,9789,908788,287344, getdate() );
INSERT INTO Bank_Details VALUES (1203,10161,1203,1203,1203,1203, getdate() );
INSERT INTO Bank_Details VALUES (1204,10211,345341,344324,534,3456, getdate() );
INSERT INTO Bank_Details VALUES (1304,10221,23435,21212,645,3432, getdate() );
INSERT INTO Bank_Details VALUES (2304,10231,254,341224,34,45534, getdate() );
INSERT INTO Bank_Details VALUES (1203,10151,54345,32412,1234,768534, getdate() );
INSERT INTO Bank_Details VALUES (2204,10201,2204,2204,2204,2204, getdate() );
INSERT INTO Bank_Details VALUES (2303,10241,2303,2303,2303,2303, getdate() );
INSERT INTO Bank_Details VALUES (2203,10171,54345,32412,1234,768534, getdate() );
INSERT INTO Bank_Details VALUES (1103,10181,1103,1103,1103,1103, getdate() );
INSERT INTO Bank_Details VALUES (2301,10191,10191,10191,10191,10191, getdate() );

```

IV. Pay Slip Generator Query: Multi Table Merge

This table joins multiple table and fetches values to generate the Payslip for a given veteran and a given pay cycle:

```

insert into pay_slip
select (Year(getdate()) * month(getdate())) * A.Veteran_ID ) as PaySlipID,
       Year(getdate()) as Year_on_slip, month(getdate()) as Month_on_slip,
       A.Veteran_ID , A.PenID, A.AllowanceID,
       case when A.Martyred_Flag =1 or a.DoD is not null then 0 else B.Pension_Amount end as
Pension_Gross , C.Amount as Allowance_Amount ,(case when A.Martyred_Flag =1 or a.DoD is not null then
0 else B.Pension_Amount end + C.Amount) as Total_Gross ,
       0.3*(case when A.Martyred_Flag =1 or a.DoD is not null then 0 else B.Pension_Amount end +
C.Amount) as Deductions,0.7*(case when A.Martyred_Flag =1 or a.DoD is not null then 0 else
B.Pension_Amount end +C.Amount) as Total_net ,
       case when A.Martyred_Flag =1 or a.DoD is not null then 'Nominee_Account' else
'Veteran_Account' end as Payment_account_type,
       case when A.Martyred_Flag =0 or a.DoD is null then D.self_acc_no else D.Nominee_acc_no end
as Payment_Account_Number
from MVPPI_Veterans A
left join Pension_Slab B
on A.PenID= B.PenID
left join Allowance_Slab C
on A.AllowanceID=C.Allowance_ID
left join Bank_Details D
on A.Veteran_ID=D.Service_id
left join MOPI E
on A.Veteran_ID=E.Service_id

```

7. Queries to Answer Major Data Questions

Report 1 : Payslip Report (To be Generated on the 7th of every month)

```
select (Year(getdate()) * month(getdate()) * A.Veteran_ID ) as PaySlipID,
       Year(getdate()) as Year_on_slip, month(getdate()) as Month_on_slip,
       A.Veteran_ID , A.PenID, A.AllowanceID,
       case when A.Martyred_Flag =1 or a.DoD is not null then 0 else B.Pension_Amount end as
Pension_Gross , C.Amount as Allowance_Amount ,(case when A.Martyred_Flag =1 or a.DoD is not null then
0 else B.Pension_Amount end + C.Amount) as Total_Gross ,
       0.3*(case when A.Martyred_Flag =1 or a.DoD is not null then 0 else B.Pension_Amount end +
C.Amount) as Deductions,0.7*(case when A.Martyred_Flag =1 or a.DoD is not null then 0 else
B.Pension_Amount end +C.Amount) as Total_net ,
       case when A.Martyred_Flag =1 or a.DoD is not null then 'Nominee_Account' else
'Veteran_Account' end as Payment_account_type,
       case when A.Martyred_Flag =0 or a.DoD is null then D.self_acc_no else D.Nominee_acc_no end
as Payment_Account_Number
from MVPPI_Veterans A
left join Pension_Slab B
on A.PenID= B.PenID
left join Allowance_Slab C
on A.AllowanceID=C.Allowance_ID
left join Bank_Details D
on A.Veteran_ID=D.Service_id
left join MOPI E
on A.Veteran_ID=E.Service_id
```

Report 2 : Total salary drawn till date by a veteran till date

```
SELECT Veteran_ID,COUNT(DISTINCT Month_on_slip) AS Number_of_pay_cycles,SUM(Total_net) AS TOTAL_SALARY
into Total_drawn_salary
FROM PAY_SLIP
GROUP BY Veteran_ID
```

Report 3 : Number of Servicemen in each of the departments

```
--Report 3
select Department_Serving, count(*) as Number_of_servicemen
into Head_Count_per_Dept
from mopi
group by department_serving
```

Report 4 : Number of Martyrs in Each of the Deployed locations

```
--Report 4
select Commission_location, count(*) as Number_Of_Martyrs
into Number_Of_martyrs
from MOPI
where martyred_flag =1
group by Commission_location
```

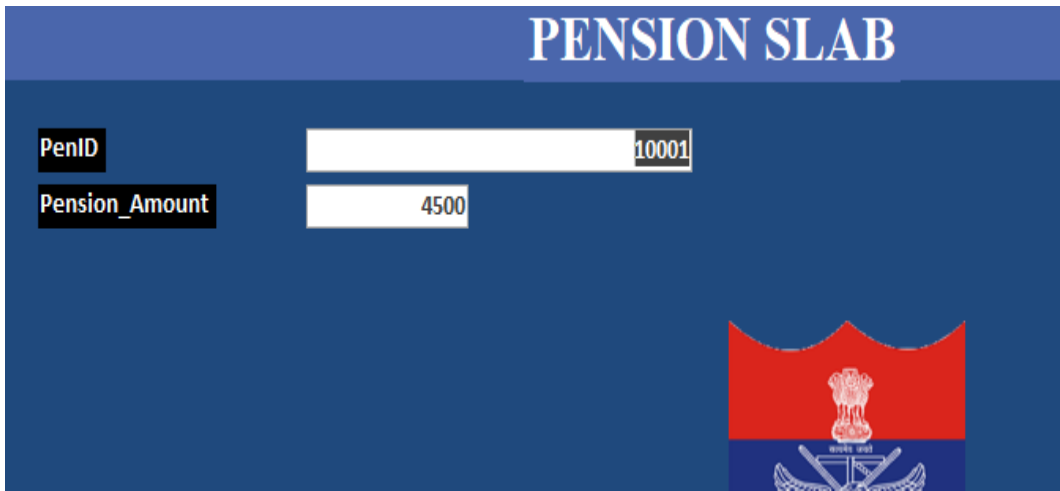
Report 5 : Number of Nominee Payees and Verteran Payees

```
select Payment_account_type,(count(*)/count(distinct Month_on_slip)) as Number_Of_Nominee_Account
into Number_Of_Nominee_Accounts
from pay_slip group by Payment_account_type
```

8. Interfaces

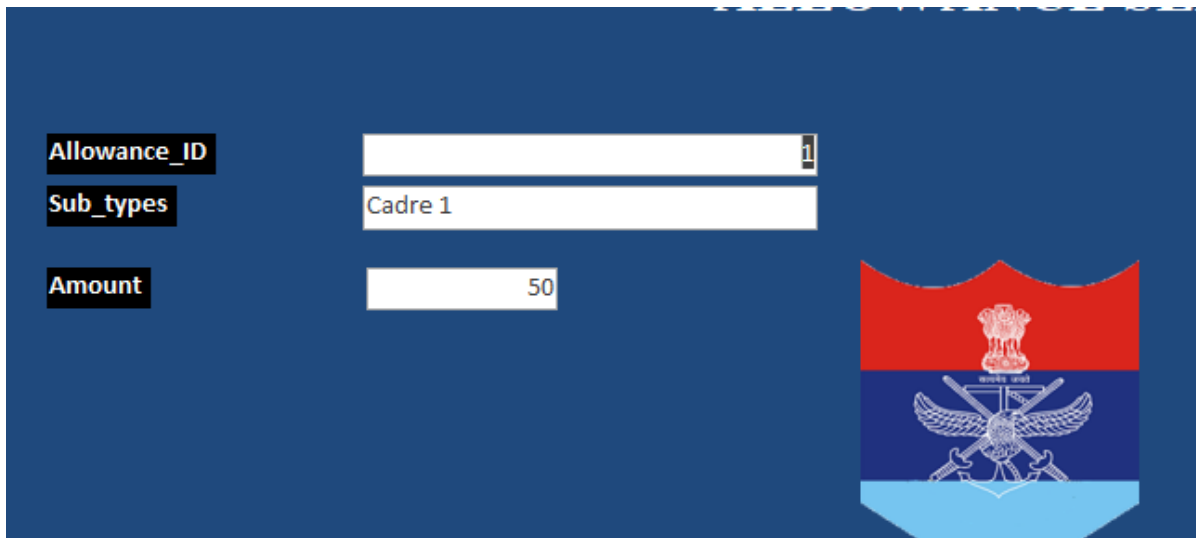
I. Forms: Total Number of Tables : 07 Total Number of Forms : 07

- 1) **Pension Slab Form:** Allows the government's finance department to change the values in the pension slab.



The screenshot shows a web form titled "PENSION SLAB" in a blue header. Below the header, there are two input fields: "PenID" with the value "10001" and "Pension_Amount" with the value "4500". The form has a dark blue background. In the bottom right corner, there is a logo featuring a red shield with a white emblem and a blue banner below it.

- 2) **Allowance Slab Form:** Allows the government to change the values in the allowance slab



The screenshot shows a web form titled "ALLOWANCE SLAB" in a blue header. Below the header, there are three input fields: "Allowance_ID" with the value "1", "Sub_types" with the value "Cadre 1", and "Amount" with the value "50". The form has a dark blue background. In the bottom right corner, there is a logo featuring a red shield with a white emblem and a blue banner below it.

- 3) **Officer's Data Base:** Allows MOPI to add new officers, edit their information and add retirement/Martyr flags.

MOPI can add new officers, edit current officer information and add Retirement/Martyr status and their dates in the data base. They can choose Allowance ID from the drop down options which is linked to the Allowance Slab.

Combo Box : Allowance Slab

MILITARY OFFICERS PAY-ROLL INSTITUTE DATA BASE

Service ID	1101	Date of Birth (YYYY-MM-DD)	1978-06-08
Serviceman_Name	Mark Butler	Gender (1: Male)	1
Department_Serving	Medical	Marital_Status (1: Married)	1
Commission_years	23	Retirement_Status (1: Retired)	1
Commission_location	Afghanistan	Martyr_Status (1: Martyred)	0
Retirement/Martyr_Date (YYYY-MM-DD)		2009-08-03	
PayId		1007	
Allowance_ID (Drop Down)		7	
last_update		2019-12-03	

Record: 1 of 24 No Filter Search

4) Veteran's Data Base Form: Allows MVPPI to view the current veterans.

The data base is automatically updated after every insert row in the MOPI or every update in the retirement_date in the MOPI table.

The MVPPI_Veterans can update Date of death for retired veterans as for DEAD and RETIRED veterans, the pension should go to the Nominee accounts and no the Veteran's Bank accounts

Veterans Information

Veteran_ID	1101
Name	Mark Butler
Department_Served	Medical
Commission_years	23
Commission_location	Afghanistan
DoB	1978-06-08
DoD	
Martyred_Flag	0
ServiceID	1101
PenID	10005
AllowanceID	7

last_update 2019-12-03

- 5) **Pay_Slip_Generator Form:** Used to Generate the payslip of a given Veteran for a given month and year.

Triggers a report for that particular veteran's payslip.

Combo Boxes : VeteranID, Month and Year

Once the veteran enters his Id ,chooses a pay date and presses the button, it is going to generate a report which is discussed in the Report section.

The screenshot shows a web form titled "PAY SLIP GENERATOR" in a large, bold, serif font. Below the title, there are three dropdown menus. The first is labeled "Veteran_ID" and has the value "1102" selected. The second is labeled "Month_on_slip" and has the value "11" selected. The third is labeled "Year_on_slip" and has the value "2019" selected. Below these dropdowns is a large, light blue rectangular button with a small icon in the center. The background of the form is dark blue with a faint American flag pattern and a central emblem.

6) **Details Change request form:**

Allows a veteran/Nominee to request changes to his particular form. Can choose their service Id and Nominee Id to edit the details.After the Veteran/Nominee enters his credentials and clicks the button, form 7 will be triggered.

Combo Boxes : ServieId and NomiID

The screenshot shows a web form titled "Choose the ServiceID Nominee combination to view/edit" in a bold, serif font. Below the title, there are two dropdown menus. The first is labeled "Service ID" and has the value "2103" selected. The second is labeled "NomiID" and is currently empty. Below these dropdowns is a large, light blue rectangular button with a small icon in the center. The background of the form is dark blue.

- 7) **Nominee and Banking Information:** Allow MOPI to access and edit all the Nominees and Banking details. Here the bank details has been embedded as a subform.

MOPI can access all the records in these tables. The uses can access only their information through form 6.

Nominee and Bank Details						
NomID		10021				
Service_ID		1201				
Name		Michael Scorsesse				
SUBFORM Bank Details		Self_acc_no	Nominee_acc_no	Self_IFSC_Code	Nominee_IFSC_Code	last_update
		1232334	2345 13434	1331		2019-12-03
	*					
Record: 1 of 1 No Filter Search						

II. Reports

1. Calculate the Pension Amount, Allowance amount and the Total net pay after deductions for each of the Veterans for a given month. Maintain a record of all the payslips issued for all veterans.
2. Calculate number of personnel in each of the serving in each of the departments
3. To calculate the total pension drawn till date (Sum total of all Payslips amounts) for all the veterans.
4. To calculate number of payslips issued to nominee accounts and issued to veteran's accounts.
5. Calculate the officers Martyred in each of the Post Locations (Countries).

Report 1

The PaySlip report is generated through the Payslip Generator Form where a veteran furnishes his details and chooses a Pay cycle date for which he wants his payslip to be generated.

The Button bellow is embedded with a report which will generate the payslip for the given veteran and a given Month and Year.

PAY SLIP GENERATOR

Veteran_ID

Month_on_slip

Year_on_slip

PAY-SLIP REPORT

PaySlipID

Veteran_ID

Year

Month

PenID

AllowanceID

Payment_account_type

Payment_Account_Numb

Pension_Gross

Allowance_Amount

Total_Gross

Deductions

Total_net

Report 2

To get the Number of Servicemen in each departments

Head_Count_per_Dept	
Department_Serving	Number_of_servicemen
Armory	8
Medical	8
Stratergy	8

Wednesday, December 4, 2019

Page 1 of 1

Report 3

To get the Number of Nominee Payees vs Number of veteran Payees

Number_Of_Nominee_Accounts	
Payment_account_type	Number_Of_Nominee_Account
Nominee_Account	7
Veteran_Account	6

Report 4

To Show the total pension drawn (ALL TIME) by each of the veterans

TOTAL_DRAWN_Penison	
Veteran_ID	TOTAL_Pension_Drawn_Alltime
1101	5670
1102	5460
1201	9870
1204	630
1301	9870
1303	1365
1304	1470
2101	1365
2103	1365
2201	9870
2203	1470
2302	9870
2304	735

Wednesday, December 4, 2019

Page 1 of 1

Report 5

To keep track of Number of Martyrs in each of the service locations

Number_Of_Martyrs	
Commission_location	Number_Of_Martyrs
Afghanistan	3
Iraq	4

Wednesday, December 4, 2019

Page 1 of 1

9. TRRIGERS: Testing and Demo

I. Trigger 1:

Inserting veterans into the MVPPI data base from the MOPI Data base

```
create trigger trigger1
on MOPI
for insert --triggered when a new enry is made in the MOPI data base
as
if @@ROWCOUNT>=1
begin
Insert into MVPPI_Veterans
select i.Service_ID, i.S_name
,i.Department_Serving,i.Commission_years,i.Commission_location,i.DoB,
case when i.Martyred_flag = 1 then i.Retirement_DateorMartyr_Date else NULL end,
i.Martyred_flag, i.Service_ID,
case when i.Commission_years > 30 then 10007
when i.Commission_years <=30 and i.Commission_years > 25 then 10006
when i.Commission_years <=25 and i.Commission_years > 20 then 10005
when i.Commission_years <=20 and i.Commission_years > 15 then 10004
when i.Commission_years <=15 and i.Commission_years > 10 then 10003
when i.Commission_years <=10 and i.Commission_years > 5 then 10002
else 10001 end,
i.AllowanceID,i.last_update
from inserted i
where i.retired_flag=1 or i. martyred_flag =1
end
```

Before Execution:

Before running the insert into MOPI command

Veteran_ID	Name	Department_Served	Commission_years	Commission_location	DoB	DoD	Martyred_Flag	ServiceID	PenID	AllowanceID	last_update
------------	------	-------------------	------------------	---------------------	-----	-----	---------------	-----------	-------	-------------	-------------

After Execution:

The MVPPI_Veterans Table after the 'Insert into MOPI' command is run

	Veteran_ID	Name	Department_Served	Commission_years	Commission_location	DoB	DoD	Martyred_Flag	ServiceID	PenID	AllowanceID	last_update
1	1101	Mark Butler	Medical	23	Afghanistan	1978-06-...	NULL	0	1101	10005	7	2019-12-11
2	1102	Lucas Giligan	Medical	20	Afghanistan	1980-08-...	NULL	0	1102	10004	6	2019-12-11
3	1201	Jade Scorsese	Strategy	32	Afghanistan	1965-03-...	NULL	0	1201	10007	7	2019-12-11
4	1204	Drake Vincent	Strategy	2	Afghanistan	1995-01-...	2014-05-...	1	1204	10001	8	2019-12-11
5	1301	Priyanka Talpa...	Amory	35	Afghanistan	1962-11-...	NULL	0	1301	10007	7	2019-12-11
6	1303	Kiara Affleck	Amory	15	Afghanistan	1986-10-...	2015-01-...	1	1303	10003	13	2019-12-11
7	1304	Adam Kapur	Amory	10	Afghanistan	1995-01-...	2014-05-...	1	1304	10002	11	2019-12-11
8	2101	Girisha Matunga	Medical	35	Iraq	1967-06-...	2019-12-...	1	2101	10007	13	2019-12-11
9	2103	Narcissa Malfoy	Medical	30	Iraq	1968-05-...	2012-01-...	1	2103	10006	13	2019-12-11
10	2201	Mahesh Kaushal	Strategy	32	Iraq	1965-03-...	NULL	0	2201	10007	7	2019-12-11
11	2203	Jim Beesley	Strategy	22	Iraq	1988-01-...	2016-05-...	1	2203	10005	11	2019-12-11
12	2302	Albert Black P...	Amory	34	Iraq	1968-05-...	NULL	0	2302	10007	7	2019-12-11
13	2304	Paul Murphy	Amory	5	Iraq	1997-04-...	2019-05-...	1	2304	10001	9	2019-12-11

II. TRIGGER 2

Updating the retirement date and retirement_flag or martyr_flag for a serviceman in the MOPI table should push that particular serviceman into the Veteran's table as he has retired.

```

drop trigger trigger2
create trigger trigger2
on MOPI
for update
as
if update(Retirement_DateorMartyr_Date)
begin
delete MVPPI_Veterans
insert into MVPPI_Veterans
select i.Service_ID, i.S_name
,i.Department_Serving,i.Commission_years,i.Commission_location,i.DoB,
case when i.Martyred_flag = 1 then i.Retirement_DateorMartyr_Date else NULL end,
i.Martyred_flag, i.Service_ID,
case when i.Commission_years > 30 then 10007
when i.Commission_years <=30 and i.Commission_years > 25 then 10006
when i.Commission_years <=25 and i.Commission_years > 20 then 10005
when i.Commission_years <=20 and i.Commission_years > 15 then 10004
when i.Commission_years <=15 and i.Commission_years > 10 then 10003
when i.Commission_years <=10 and i.Commission_years > 5 then 10002
else 10001 end,
i.AllowanceID,i.last_update
from MOPI i
where i.retired_flag=1 or i.martyred_flag =1
end

```

The UPDATE command

```

update MOPI
set Retired_Flag=1 , Retirement_DateorMartyr_Date= '12-11-2019'
where MOPI.Service_ID= 2301

```

Before Execution:

MVPPI Veterans Table

	Veteran_ID	Name	Department_Served	Commission_years	Commission_location	DoB	DoD	Martyred_Flag	ServiceID	PenID	AllowanceID	last_update
1	1101	Mark Butler	Medical	23	Afghanistan	1978-06-...	NULL	0	1101	10005	7	2019-12-11
2	1102	Lucas Giligan	Medical	20	Afghanistan	1980-08-...	NULL	0	1102	10004	6	2019-12-11
3	1201	Jade Scorcesse	Strategy	32	Afghanistan	1965-03-...	NULL	0	1201	10007	7	2019-12-11
4	1204	Drake Vincent	Strategy	2	Afghanistan	1995-01-...	2014-05-...	1	1204	10001	8	2019-12-11
5	1301	Priyanka Talpa...	Amory	35	Afghanistan	1962-11-...	NULL	0	1301	10007	7	2019-12-11
6	1303	Kiara Affleck	Amory	15	Afghanistan	1986-10-...	2015-01-...	1	1303	10003	13	2019-12-11
7	1304	Adam Kapur	Amory	10	Afghanistan	1995-01-...	2014-05-...	1	1304	10002	11	2019-12-11
8	2101	Girisha Matunga	Medical	35	Iraq	1967-06-...	2019-12-...	1	2101	10007	13	2019-12-11
9	2103	Narcissa Malfoy	Medical	30	Iraq	1968-05-...	2012-01-...	1	2103	10006	13	2019-12-11
10	2201	Mahesh Kaushal	Strategy	32	Iraq	1965-03-...	NULL	0	2201	10007	7	2019-12-11
11	2203	Jim Beesley	Strategy	22	Iraq	1988-01-...	2016-05-...	1	2203	10005	11	2019-12-11
12	2302	Albert Black P...	Amory	34	Iraq	1968-05-...	NULL	0	2302	10007	7	2019-12-11
13	2304	Paul Murphy	Amory	5	Iraq	1997-04-...	2019-05-...	1	2304	10001	9	2019-12-11

After Execution:

	Veteran_ID	Name	Department_Served	Commission_years	Commission_location	DoB	DoD	Martyred_Flag	ServiceID	PenID	AllowanceID	last_update
1	1101	Mark Butler	Medical	23	Afghanistan	1978-06-...	NULL	0	1101	10005	7	2019-12-11
2	1102	Lucas Giligan	Medical	20	Afghanistan	1980-08-...	NULL	0	1102	10004	6	2019-12-11
3	1201	Jade Scorcesse	Strategy	32	Afghanistan	1965-03-...	NULL	0	1201	10007	7	2019-12-11
4	1204	Drake Vincent	Strategy	2	Afghanistan	1995-01-...	2014-05-...	1	1204	10001	8	2019-12-11
5	1301	Priyanka Talpa...	Amory	35	Afghanistan	1962-11-...	NULL	0	1301	10007	7	2019-12-11
6	1303	Kiara Affleck	Amory	15	Afghanistan	1986-10-...	2015-01-...	1	1303	10003	13	2019-12-11
7	1304	Adam Kapur	Amory	10	Afghanistan	1995-01-...	2014-05-...	1	1304	10002	11	2019-12-11
8	2101	Girisha Matunga	Medical	35	Iraq	1967-06-...	2019-12-...	1	2101	10007	13	2019-12-11
9	2103	Narcissa Malfoy	Medical	30	Iraq	1968-05-...	2012-01-...	1	2103	10006	13	2019-12-11
10	2201	Mahesh Kaushal	Strategy	32	Iraq	1965-03-...	NULL	0	2201	10007	7	2019-12-11
11	2203	Jim Beesley	Strategy	22	Iraq	1988-01-...	2016-05-...	1	2203	10005	11	2019-12-11
12	2301	Alfred Ebone	Amory	12	Iraq	1992-11-...	NULL	0	2301	10003	5	2019-12-11
13	2302	Albert Black P...	Amory	34	Iraq	1968-05-...	NULL	0	2302	10007	7	2019-12-11
14	2304	Paul Murphy	Amory	5	Iraq	1997-04-...	2019-05-...	1	2304	10001	9	2019-12-11

III. Triggers Demo in Access

In this demo, a row of new data is inserted into the MOPI Data base. The retirement flag is set to 1 and the retirement date is keyed in. As a result of the declared trigger, this entrant into the MOPI data base will be added to the MVPPI Data base too.

Initially, the MVPPI veteran's data base has 13 records. After the execution of the new insert command, the final number of records in the MVPPI_Verteran's data base becomes 14. (**Attached Separately too, if the link doesn't work**)



Access - Database4_Database- C:_Users_Krishna_Documents_Database4.accdb (Access 2007 - 2016 file format) 2019-12-05 00-50-06.mp4