

Design goal:

Check the missing weapon, the elements included in maneuver and the relationship between trainings and soldiers.

Weapon (**weapon_ID**, weapon_name, purchase_date, missing_date)

Assumption: Every weapon will have record if borrow or return.

Constraints: Weapon with flaws cannot store with other weapons.

Relationship: 1 training involve 1 weapon.

1 warehouse have many weapons.

1 maneuver require many weapons

Training (**training_code**, training_name, soldier_ID, weapon_ID, weapon_name, date)

Assumption: Every training has different instructor.

Soldier who missed the training without reason will fail the training test.

Training can only participate in with soldier age between 18 and 55.

Constraints: Soldier who did not pass the test three times in a row will be kick out of the military.

Relationship: 1 weapon provide for 1 training.

1 training involve 1 transportation.

Many soldiers can have many trainings.

Soldier (**soldier_ID**, soldier_name, base_code, Dept_ID, rank)

Assumption: Only people who have age between 18 to 55 can join to the military.

Constraints: People who have critical illness can not join the military.

Relationship: Many soldiers can have many trainings.

1 department have many soldiers.

1 military base have many soldier

Department (**Department_ID**, Department_name)

Assumption: Department can only maximum manager 10000 soldiers.

Relationship: 1 department have many soldiers.

1 military base have many departments.

1 maneuver participated by many departments.

Military_Base (**Base_code**, Base_name)

Assumption: Military base can only maximum manager 25 departments.

Relationship: 1 military base have many departments.

1 military base manager many warehouses.

Many military bases participate in one maneuver.

1 military base manager many soldiers.

Warehouse (**house_ID**, house_name, weapon_ID, weapon_name)

Assumption: There will always have soldier to manage the warehouse.

Weapon inside the warehouse need to maintenance every week.

Constraints: Weapon did not complete the application cannot store in warehouse.

Relationship: 1 military base have many warehouses.

1 warehouse custody many weapons.

Transportation (**Trans_ID**, Trans_name, Base_code)

Assumption: These transportations need to be maintenance every 2 days.

If the transportation dispatched, their will always have the record.

Relationship: 1 training involve 1 transportation.

1 maneuver require many transportations.

Maneuver (**Maneuver_ID**, Maneuver_name, weapon_ID, Trans_ID, Base_code, Date)

Assumption: Maneuver only hold in the big event.

Maneuver included every type of military.

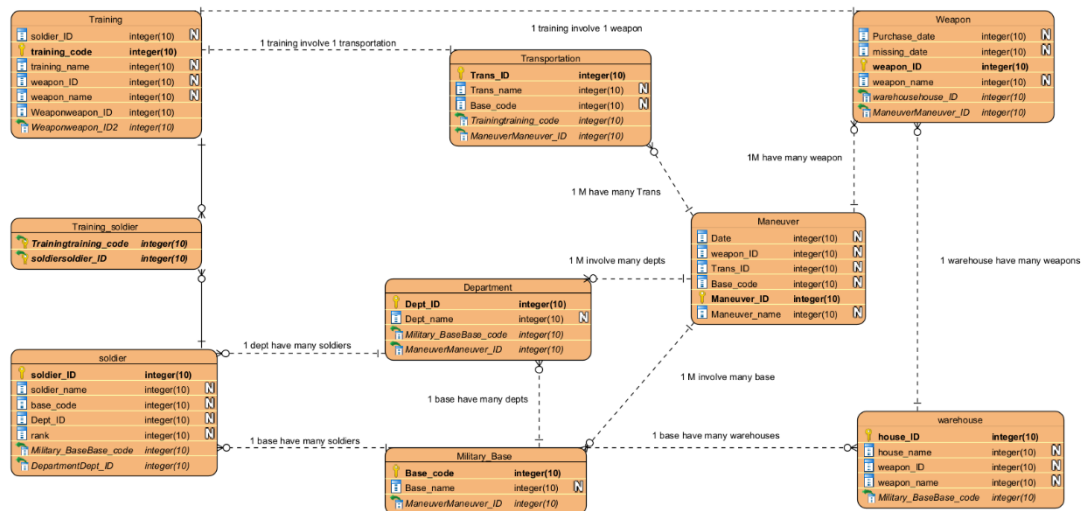
Maneuver require participated by every military base.

Relationship: 1 maneuver need many weapons.

1 maneuver require many transportations.

1 maneuver participated by many military bases.

1 maneuver participated by many departments.



Weapon (**weapon_ID**, weapon_name, purchase_date, missing_date)

This entity did not have partial dependency. (2NF)

This entity did not have transitive dependencies. (3NF)

This entity has no key attribute is transitively dependent on the key. (BCNF)

This entity has no multiple sets of multivalued dependencies. (4NF)

Training (**training_code**, training_name, soilder_ID, weapon_ID, weapon_name, date)

This entity did not have partial dependency. (2NF)

This entity has transitive dependencies (training_name → weapon_ID) & (weapon_ID → weapon_name). (Not in 3NF)

To put it into the 3 NF.

Training (**training_code**, training_name)

Training (**training_name**,soilder_ID)

Training (**training_name**, weapon_ID)

Training (**training_name**, weapon_name)

This entity has no key attribute is transitively dependent on the key. (BCNF)

This entity has multiple sets of multivalued dependencies. (Not in 4NF)

Training (**training_name**, soilder_ID) One training_name will have multiple soldier_ID.

Soldier (**soldier_ID**, soldier_name, base_code, Dept_ID, rank)

This entity did not have partial dependency. (2NF)

This entity has transitive dependencies. (Not in 3NF)

Soldier (**soldier_ID**, soldier_name)

Soldier (**soldier_name**, base_code, Dept_ID, rank)

This entity has no key attribute is transitively dependent on the key. (BCNF)

This entity has no multiple sets of multivalued dependencies. (4NF)

Department (**Department_ID**, Department_name)

This entity did not have partial dependency. (2NF)

This entity did not have transitive dependencies. (3NF)

This entity has no key attribute is transitively dependent on the key. (BCNF)

This entity has no multiple sets of multivalued dependencies. (4NF)

Military_Base (**Base_code**, Base_name)

This entity did not have partial dependency. (2NF)

This entity did not have transitive dependencies. (3NF)

This entity has no key attribute is transitively dependent on the key. (BCNF)

This entity has no multiple sets of multivalued dependencies. (4NF)

Warehouse (**house_ID**, house_name, weapon_ID, weapon_name)

This entity did not have partial dependency. (2NF)

This entity has transitive dependencies (house_name → weapon_ID) & (weapon_ID → weapon_name). (Not in 3NF)

To put it into the 3 NF.

Warehouse (**house_ID**, house_name)

Warehouse (**house_name**, weapon_ID)

Warehouse (**house_name**, weapon_name)

This entity has no key attribute is transitively dependent on the key. (BCNF)

This entity has no multiple sets of multivalued dependencies. (4NF)

Transportation (**Trans_ID**, Trans_name, Base_code)

This entity did not have partial dependency. (2NF)

This entity did not have transitive dependencies. (3NF)

This entity has no key attribute is transitively dependent on the key. (BCNF)

This entity has no multiple sets of multivalued dependencies. (4NF)

Maneuver (**Maneuver_ID**, Maneuver_name, weapon_ID, Trans_ID, Base_code, Date)

This entity did not have partial dependency. (2NF)

This entity has transitive dependencies (**Maneuver_ID** → Maneuver_name). (Not in

3NF)

To put it into the 3 NF.

Maneuver (**Maneuver_ID**, weapon_ID, Trans_ID, Base_code, Date)

Maneuver (**Maneuver_ID**, Maneuver_name)

This entity has no key attribute is transitively dependent on the key. (BCNF)

This entity has no multiple sets of multivalued dependencies. (4NF)

Create 4 table

create table if not exists soldier

```
(  
soldier_ID char(5) not null primary key,  
soldier_name varchar(10),  
base_code char(5),  
Dept_ID char(5),  
rank varchar(10)  
);
```

```
insert into soldier values('00000', 'Joe', 'A104', 'D333', 'PVT');  
insert into soldier values('00001', 'Jimmy', 'A132', 'D333', 'PVT');  
insert into soldier values('00002', 'John', 'A111', 'D335', 'PVT');  
insert into soldier values('00003', 'Ogi', 'A105', 'D363', 'SSG');  
insert into soldier values('00004', 'Simon', 'A104', 'D385', 'CSM');  
insert into soldier values('00005', 'Tom', 'A136', 'D379', 'BG');  
insert into soldier values('00006', 'Jerry', 'A111', 'D335', '1LT');  
select * from soldier;
```

	soldier_ID	soldier_name	base_code	Dept_ID	rank
▶	00000	Joe	A104	D333	PVT
	00001	Jimmy	A132	D333	PVT
	00002	John	A111	D335	PVT
	00003	Ogi	A105	D363	SSG
	00004	Simon	A104	D385	CSM
	00005	Tom	A136	D379	BG
	00006	Jerry	A111	D335	1LT
•	NULL	NULL	NULL	NULL	NULL

create table if not exists Training

```
(
training_code char(5) not null primary key,
training_name varchar(15),
soldier_ID char(5),
weapon_ID char(5),
weapon_name varchar(20)
);
```

```
insert into Training values('36677', 'grappling', '00001', '', '');
insert into Training values('22443', 'CQB', '00004', '888', 'knife');
insert into Training values('55779', 'gunshot', '00001', '555', 'gun');
insert into Training values('66884', 'shooting', '00004', '5555', 'gun');
insert into Training values('22895', 'taekwondo', '00006', '', '');
insert into Training values('77889', 'driving', '00003', '', '');
insert into Training values('66552', 'bayonet drill', '00003', '111', 'rifle');
select * from Training;
```

	training_code	training_name	soldier_ID	weapon_ID	weapon_name
▶	22443	CQB	00004	888	knife
	22895	taekwondo	00006		
	36677	grappling	00001		
	55779	gunshot	00001	555	gun
	66552	bayonet drill	00003	111	rifle
	66884	shooting	00004	5555	gun
	77889	driving	00003		
*	NULL	NULL	NULL	NULL	NULL

```
create table if not exists Weapon
(
weapon_ID char(5),
weapon_name varchar(20),
purchase_date DATE null,
missing_date DATE null
);
```

```
insert into Weapon values('111', 'rifle', '2020-03-01', '2022-03-05');
insert into Weapon values('222', 'grenade', '2020/10/29', null);
insert into Weapon values('333', 'grenade launcher', '2020/10/10', null);
insert into Weapon values('555', 'gun', '2020/07/14', '2022/02/27');
```

```

insert into Weapon values('666', 'sniper rifle', '2020/08/30', null);
insert into Weapon values('777', 'Rocket launcher', '2021/01/01', null);
insert into Weapon values('888', 'knife', '2021/01/01', null);
insert into Weapon values('1111', 'rifle', '2020-03-01', null);
insert into Weapon values('2222', 'grenade', '2020/10/29', null);
insert into Weapon values('3333', 'grenade launcher', '2020/10/10', null);
insert into Weapon values('5555', 'gun', '2020/07/14', null);
insert into Weapon values('6666', 'sniper rifle', '2020/08/30', null);
insert into Weapon values('7777', 'Rocket launcher', '2021/01/01', null);
insert into Weapon values('8888', 'knife', '2021/01/01', null);
select * from Weapon;

```

	weapon_ID	weapon_name	purchase_date	missing_date
▶	111	rifle	2020-03-01	2022-03-05
	222	grenade	2020-10-29	NULL
	333	grenade launcher	2020-10-10	NULL
	555	gun	2020-07-14	2022-02-27
	666	sniper rifle	2020-08-30	NULL
	777	Rocket launcher	2021-01-01	NULL
	888	knife	2021-01-01	NULL
	1111	rifle	2020-03-01	NULL
	2222	grenade	2020-10-29	NULL
	3333	grenade launcher	2020-10-10	NULL
	5555	gun	2020-07-14	NULL
	6666	sniper rifle	2020-08-30	NULL
	7777	Rocket launcher	2021-01-01	NULL
	8888	knife	2021-01-01	NULL

create table if not exists warehouse

```

(
house_ID char(5),
house_name varchar(20),
weapon_ID char(5),
weapon_name varchar(20)
);

```

```

insert into warehouse values('H-01', 'Topaz', '111', 'rifle');
insert into warehouse values('H-01', 'Topaz', '222', 'grenade');
insert into warehouse values('H-01', 'Topaz', '333', 'grenade launcher');
insert into warehouse values('H-01', 'Topaz', '555', 'gun');
insert into warehouse values('H-02', 'Semi', '666', 'sniper rifle');

```

```

insert into warehouse values('H-02', 'Semi', '777', 'Rocket launcher');
insert into warehouse values('H-02', 'Semi', '888', 'knife');
insert into warehouse values('H-02', 'Semi', '1111', 'rifle');
insert into warehouse values('H-04', 'Cho', '2222', 'grenade');
insert into warehouse values('H-04', 'Cho', '3333', 'grenade launcher');
insert into warehouse values('H-04', 'Cho', '5555', 'gun');
insert into warehouse values('H-04', 'Cho', '6666', 'sniper rifle');
insert into warehouse values('H-04', 'Cho', '7777', 'Rocket launcher');
insert into warehouse values('H-04', 'Cho', '8888', 'knife');
select * from warehouse;

```

	house_ID	house_name	weapon_ID	weapon_name
▶	H-01	Topaz	111	rifle
	H-01	Topaz	222	grenade
	H-01	Topaz	333	grenade launcher
	H-01	Topaz	555	gun
	H-02	Semi	666	sniper rifle
	H-02	Semi	777	Rocket launcher
	H-02	Semi	888	knife
	H-02	Semi	1111	rifle
	H-04	Cho	2222	grenade
	H-04	Cho	3333	grenade launcher
	H-04	Cho	5555	gun
	H-04	Cho	6666	sniper rifle
	H-04	Cho	7777	Rocket launcher
	H-04	Cho	8888	knife

- **select involving one/more conditions in Where Clause**
Find all name of the soldier who have training for CQB

```

select soldier_name, training_name from soldier, training
where soldier.soldier_ID = training.soldier_ID
and training_name = 'CQB';

```

	soldier_name	training_name
▶	Simon	CQB

- **select with aggregate functions (i.e., SUM,MIN,MAX,AVG,COUNT)**
Find all number of missing weapons from each warehouse

```

select house_ID, count(missing_date) from warehouse, weapon
where warehouse.weapon_ID = weapon.weapon_ID
group by house_ID;

```


	house_ID	count(missing_date)
▶	H-01	2
	H-02	0
	H-04	0

- **select with Having, Group By, Order By clause**

Find all name of the soldier who take over one training

```
select soldier_name, count(training_name) from soldier, training
where soldier.soldier_ID = training.soldier_ID
group by soldier_name having count(training_name) > 1;
```

	soldier_name	count(training_name)
▶	Jimmy	2
	Ogi	2
	Simon	2

- **Nested Select**

Find all the name of training that Jimmy takes

```
select training_name, soldier_name from soldier, training
where soldier.soldier_ID = training.soldier_ID
and soldier.soldier_ID in
(select soldier_ID from training
where soldier_name = 'Jimmy');
```

	training_name	soldier_name
▶	grappling	Jimmy
	gunshot	Jimmy

- **select involving the Union operation**

Find all name of the weapon that store in warehouse 'H-01' and 'H-04'

```
select weapon_name from warehouse
where house_ID = 'H-01'
union
select weapon_name from warehouse
where house_ID = 'H-04';
```

	weapon_name
▶	rifle
	grenade
	grenade launcher
	gun
	sniper rifle
	Rocket launcher
	knife

- **insert one tuple into a table (for 2 tables, just add 3 records for each table)**
insert three new soldiers in soldier table.

insert into soldier values ('02000', 'Andy', 'A888', 'D328', 'PVT');

insert into soldier values ('02111', 'Chou', 'A456', 'D321', 'PVT');

insert into soldier values ('02333', 'Weyne', 'A222', 'D322', 'PVT');

Before

	soldier_ID	soldier_name	base_code	Dept_ID	rank
▶	00000	Joe	A104	D333	PVT
	00001	Jimmy	A132	D333	PVT
	00002	John	A111	D335	PVT
	00003	Ogi	A105	D363	SSG
	00004	Simon	A104	D385	CSM
	00005	Tom	A136	D379	BG
	00006	Jerry	A111	D335	1LT
•	NULL	NULL	NULL	NULL	NULL

After

	soldier_ID	soldier_name	base_code	Dept_ID	rank
▶	00000	Joe	A104	D333	PVT
	00001	Jimmy	A132	D333	PVT
	00002	John	A111	D335	PVT
	00003	Ogi	A105	D363	SSG
	00004	Simon	A104	D385	CSM
	00005	Tom	A136	D379	BG
	00006	Jerry	A111	D335	1LT
	02000	Andy	A888	D328	PVT
	02111	Chou	A456	D321	PVT
	02333	Weyne	A222	D322	PVT
•	NULL	NULL	NULL	NULL	NULL

Insert three weapons in weapon table.

insert into Weapon values ('9999', 'machinegun', '2021/05/01', null);

insert into Weapon values ('9988', 'machinegun', '2021/05/01', '2022/03/25');

insert into Weapon values ('9977', 'machinegun', '2021/05/01', null);

Before

	weapon_ID	weapon_name	purchase_date	missing_date
▶	111	rifle	2020-03-01	2022-03-05
	222	grenade	2020-10-29	NULL
	333	grenade launcher	2020-10-10	NULL
	555	gun	2020-07-14	2022-02-27
	666	sniper rifle	2020-08-30	NULL
	777	Rocket launcher	2021-01-01	NULL
	888	knife	2021-01-01	NULL
	1111	rifle	2020-03-01	NULL
	2222	grenade	2020-10-29	NULL
	3333	grenade launcher	2020-10-10	NULL
	5555	gun	2020-07-14	NULL
	6666	sniper rifle	2020-08-30	NULL
	7777	Rocket launcher	2021-01-01	NULL
	8888	knife	2021-01-01	NULL

After

	weapon_ID	weapon_name	purchase_date	missing_date
▶	111	rifle	2020-03-01	2022-03-05
	222	grenade	2020-10-29	NULL
	333	grenade launcher	2020-10-10	NULL
	555	gun	2020-07-14	2022-02-27
	666	sniper rifle	2020-08-30	NULL
	777	Rocket launcher	2021-01-01	NULL
	888	knife	2021-01-01	NULL
	1111	rifle	2020-03-01	NULL
	2222	grenade	2020-10-29	NULL
	3333	grenade launcher	2020-10-10	NULL
	5555	gun	2020-07-14	NULL
	6666	sniper rifle	2020-08-30	NULL
	7777	Rocket launcher	2021-01-01	NULL
	8888	knife	2021-01-01	NULL
	9999	machinegun	2021-05-01	NULL
	9988	machinegun	2021-05-01	2022-03-25
	9977	machinegun	2021-05-01	NULL

- insert a set of tuples (by using another select statement)

Create a soldier.PVT table with soldier rank PVT.

Create table soldierPVT as

Select * from soldier where rank = 'PVT';

	soldier_ID	soldier_name	base_code	Dept_ID	rank
▶	00000	Joe	A104	D333	PVT
	00001	Jimmy	A132	D333	PVT
	00002	John	A111	D335	PVT
	02000	Andy	A888	D328	PVT
	02111	Chou	A456	D321	PVT
	02333	Weyne	A222	D322	PVT
*	NULL	NULL	NULL	NULL	NULL

- **insert involving two tables**
create a table with soldier and training

Create table soldiertraining2 as

select soldier.soldier_ID, soldier_name, rank, training_name from soldier,
training

where soldier.soldier_ID = training.soldier_ID;

	soldier_ID	soldier_name	rank	training_name
▶	00004	Simon	CSM	CQB
	00006	Jerry	1LT	taekwondo
	00001	Jimmy	PVT	grappling
	00001	Jimmy	PVT	gunshot
	00003	Ogi	SSG	bayonet drill
	00004	Simon	CSM	shooting
	00003	Ogi	SSG	driving

- **delete one tuple or a set of tuples: from one table, from multiple tables.**
Delete soldier Jerrt from soldiertraining2.

delete from soldiertraining2

where soldier_name = 'Jerry';

Before

	soldier_ID	soldier_name	rank	training_name
▶	00004	Simon	CSM	CQB
	00006	Jerry	1LT	taekwondo
	00001	Jimmy	PVT	grappling
	00001	Jimmy	PVT	gunshot
	00003	Ogi	SSG	bayonet drill
	00004	Simon	CSM	shooting
	00003	Ogi	SSG	driving

After

	soldier_ID	soldier_name	rank	training_name
▶	00004	Simon	CSM	CQB
	00001	Jimmy	PVT	grappling
	00001	Jimmy	PVT	gunshot
	00003	Ogi	SSG	bayonet drill
	00004	Simon	CSM	shooting
	00003	Ogi	SSG	driving

- **Update one tuple or a set of tuples: from one table, from multiple tables.**

```

update soldier
set rank = '1LT'
where soldier_name = 'Chou';

```

Before

	soldier_ID	soldier_name	base_code	Dept_ID	rank
▶	00000	Joe	A104	D333	PVT
	00001	Jimmy	A132	D333	PVT
	00002	John	A111	D335	PVT
	00003	Ogi	A105	D363	SSG
	00004	Simon	A104	D385	CSM
	00005	Tom	A136	D379	BG
	00006	Jerry	A111	D335	1LT
	02000	Andy	A888	D328	PVT
	02111	Chou	A456	D321	PVT
	02333	Weyne	A222	D322	PVT
★	NULL	NULL	NULL	NULL	NULL

After

	soldier_ID	soldier_name	base_code	Dept_ID	rank
▶	00000	Joe	A104	D333	PVT
	00001	Jimmy	A132	D333	PVT
	00002	John	A111	D335	PVT
	00003	Ogi	A105	D363	SSG
	00004	Simon	A104	D385	CSM
	00005	Tom	A136	D379	BG
	00006	Jerry	A111	D335	1LT
	02000	Andy	A888	D328	PVT
	02111	Chou	A456	D321	1LT
	02333	Weyne	A222	D322	PVT
★	NULL	NULL	NULL	NULL	NULL

- **Create View:** - based on one relation and more than one relation:
- operate on View (i.e., select, insert, delete, update,...)

Create a view for soldier with rank PVT and CSM of all training

```

create view SoldierPVTCSMTraining as
select soldier.soldier_ID, soldier_name, base_code, Dept_ID, rank,
training_name, weapon_name from soldier, training
where soldier.soldier_ID = training.soldier_ID
and rank = 'PVT'
union

```

```
select soldier.soldier_ID, soldier_name, base_code, Dept_ID, rank,
training_name, weapon_name from soldier, training
where soldier.soldier_ID = training.soldier_ID
and rank = 'CSM';
```

	soldier_ID	soldier_name	base_code	Dept_ID	rank	training_name	weapon_name
▶	00001	Jimmy	A132	D333	PVT	grappling	
	00001	Jimmy	A132	D333	PVT	gunshot	gun
	00004	Simon	A104	D385	CSM	CQB	knife
	00004	Simon	A104	D385	CSM	shooting	gun

Create view of all the soldier with rank not PVT

```
create view SoldiernotPVT as
select * from soldier
where rank != 'PVT' ;
```

	soldier_ID	soldier_name	base_code	Dept_ID	rank
▶	00003	Ogi	A105	D363	SSG
	00004	Simon	A104	D385	CSM
	00005	Tom	A136	D379	BG
	00006	Jerry	A111	D335	1LT
	02037	Urdy	A888	D333	CSM
	02111	Chou	A456	D321	1LT
★	NULL	NULL	NULL	NULL	NULL

- **Select**

Select all the name of soldier with rank BG

```
select * from SoldiernotPVT
where rank = 'BG';
```

	soldier_ID	soldier_name	base_code	Dept_ID	rank
▶	00005	Tom	A136	D379	BG

- **Insert**

Add new soldier Urdy in SoldiernotPVT view

```
insert into SoldiernotPVT values('02037', 'Urdy', 'A888', 'D333', 'CSM');
```

Before

	soldier_ID	soldier_name	base_code	Dept_ID	rank
▶	00003	Ogi	A105	D363	SSG
	00004	Simon	A104	D385	CSM
	00005	Tom	A136	D379	BG
	00006	Jerry	A111	D335	1LT
	02111	Chou	A456	D321	1LT

After

	soldier_ID	soldier_name	base_code	Dept_ID	rank
▶	00003	Ogi	A105	D363	SSG
	00004	Simon	A104	D385	CSM
	00005	Tom	A136	D379	BG
	00006	Jerry	A111	D335	1LT
	02037	Urdy	A888	D333	CSM
	02111	Chou	A456	D321	1LT
★	NULL	NULL	NULL	NULL	NULL

- **Update**

Update soldier Urdy's department to 'D385'

update SoldiernotPVT

set Dept_ID = 'D385'

where soldier_name = 'Urdy';

Before

	soldier_ID	soldier_name	base_code	Dept_ID	rank
▶	00003	Ogi	A105	D363	SSG
	00004	Simon	A104	D385	CSM
	00005	Tom	A136	D379	BG
	00006	Jerry	A111	D335	1LT
	02037	Urdy	A888	D333	CSM
	02111	Chou	A456	D321	1LT
★	NULL	NULL	NULL	NULL	NULL

After

	soldier_ID	soldier_name	base_code	Dept_ID	rank
▶	00003	Ogi	A105	D363	SSG
	00004	Simon	A104	D385	CSM
	00005	Tom	A136	D379	BG
	00006	Jerry	A111	D335	1LT
	02037	Urdy	A888	D385	CSM
	02111	Chou	A456	D321	1LT

- **Delete**

Delete soldier jerry from SoldiernotPVT view

delete from SoldiernotPVT
where soldier_name = 'Jerry';

Before

	soldier_ID	soldier_name	base_code	Dept_ID	rank
▶	00003	Ogi	A105	D363	SSG
	00004	Simon	A104	D385	CSM
	00005	Tom	A136	D379	BG
	00006	Jerry	A111	D335	1LT
	02037	Urdy	A888	D385	CSM
	02111	Chou	A456	D321	1LT

After

	soldier_ID	soldier_name	base_code	Dept_ID	rank
▶	00003	Ogi	A105	D363	SSG
	00004	Simon	A104	D385	CSM
	00005	Tom	A136	D379	BG
	02037	Urdy	A888	D385	CSM
	02111	Chou	A456	D321	1LT

Triggers

● enforcing referential integrity

If insert soldier did not exist cannot insert value to the training

Delimiter \$\$

create trigger addtraining before

insert on training

for each row

begin

declare temp Int;

set temp=0;

select count(*) into temp from soldier

where soldier_ID = new.soldier_ID;

if temp=0 then

insert into log values(concat('soldier',new.soldier_ID, ' is not in the system'));

insert into anytable values('Just to cause error');

end if;

end;

\$\$

```
insert into training values('55555','bomb','77777','', '');
```

```
select * from log;
```

	message
▶	soldier77777 is not in the system

```
select * from anytable;
```

	message
▶	Just to cause error

- **enforcing attribute domain constraints**

Insert soldier cannot have empty rank

Delimiter \$\$

```
create trigger insert_soldier before insert on soldier
```

```
for each row
```

```
begin
```

```
if new.rank = '' then
```

```
insert into log values(concat('soldier ',new.soldier_ID, ' rank can not be empty'));
```

```
end if;
```

```
end;
```

\$\$

```
insert into soldier values('99999', 'kevin', 'A104', 'D333', '');
```

```
select * from log;
```

	message
▶	soldier77777 is not in the system
	soldier 99999 rank can not be empty

- **creating database log**

#Insert soldier

Delimiter \$\$

```
create trigger new_soldier after insert on soldier
```

```
for each row
```

```
begin
```

```
insert into soldier_record values(concat('soldier ',new.soldier_name,' has been
```

```

added by ',current_user(), ' on
',current_date());
end;
$$
Create table soldier_record (message varchar(90));

```

```

insert into soldier values('00557', 'Chen', 'A104', 'D333', 'PVT');

```

	message
▶	soldier Chen has been added by root@localhost on 2022-03-25

#Update soldier

```

Delimiter $$
create trigger update_soldier after update on soldier
for each row
begin
insert into soldier_record values(concat('soldier ', old.soldier_name,' has been
updated by ',current_user(), ' on
',current_date()));
end;
$$

```

```

update soldier set Dept_ID = 'Y777' where soldier_name = 'Simon';

```

	message
▶	soldier Chen has been added by root@localhost on 2022-03-25
	soldier Simon has been updated by root@localhost on 2022-03-25

#Update weapon

```

Delimiter $$
create trigger update_weapon after update on weapon
for each row
begin
insert into weapon_record values(concat('weapon ', old.weapon_ID,' has been
updated by ',current_user(), ' on
',current_date()));
end;
$$

```

Create table weapon_record (message varchar(90));
update weapon set missing_date = '2022/02/05' where weapon_ID = '888';

	message
▶	weapon 888 has been updated by root@localhost on 2022-03-25

- **gathering statistics**

Count the number of soldier for each rank after Insert

Delimiter \$\$

create trigger total_rank_soldier after insert on num_soldier
for each row

begin

delete from total_rank_soldier;

insert total_rank_soldier

select rank, count(rank) from num_soldier group by rank;

end;

\$\$

create table num_soldier

(soldier_ID char(5) not null primary key,

soldier_name varchar(10),

base_code char(5),

Dept_ID char(5),

rank varchar(10));

create table total_rank_soldier(rank varchar(15), count_rank double);

insert into num_soldier values('00000', 'Joe', 'A104', 'D333', 'PVT');

insert into num_soldier values('00001', 'Jimmy', 'A132', 'D333', 'PVT');

insert into num_soldier values('00002', 'John', 'A111', 'D335', 'PVT');

insert into num_soldier values('00003', 'Ogi', 'A105', 'D363', 'SSG');

insert into num_soldier values('00004', 'Simon', 'A104', 'D385', 'CSM');

insert into num_soldier values('00005', 'Tom', 'A136', 'D379', 'BG');

insert into num_soldier values('00006', 'Jerry', 'A111', 'D335', '1LT');

insert into num_soldier values('00557', 'Chen', 'A104', 'D333', 'PVT');

	rank	count_rank
▶	1LT	1
	BG	1
	CSM	1
	PVT	4
	SSG	1

Count the number of soldiers for each rank after update

Delimiter \$\$

create trigger total_rank_soldier_update after update on num_soldier

for each row

begin

delete from total_rank_soldier;

insert total_rank_soldier

select rank, count(rank) from num_soldier group by rank;

end;

\$\$

update num_soldier

set rank = 'CSM'

where soldier_name = 'Chen';

	rank	count_rank
▶	1LT	1
	BG	1
	CSM	2
	PVT	3
	SSG	1

Coldfusion



[Click here to see Niagara Falls](#)

View Soldier List

[View Soldier List](#)

Find Soldier:

Soldier's Name:

Add Soldier:

Soldier ID: Soldier Name: Base code: Department ID: Rank:

Delete Soldier:

Soldier's Name:

Update Soldier

Soldier's Name:

[Back to Home](#)

View Soldier List

SOLDIER TABLE:

Soldier ID : 00000 Soldier Name : Joe Base Code : A104 Department ID : D333 Rank : PVT
Soldier ID : 00001 Soldier Name : Jimmy Base Code : A132 Department ID : D333 Rank : PVT
Soldier ID : 00002 Soldier Name : John Base Code : A111 Department ID : D335 Rank : PVT
Soldier ID : 00003 Soldier Name : Ogi Base Code : A105 Department ID : D363 Rank : SSG
Soldier ID : 00004 Soldier Name : Simon Base Code : A104 Department ID : D385 Rank : CSM
Soldier ID : 00005 Soldier Name : Tom Base Code : A136 Department ID : D379 Rank : BG
Soldier ID : 00006 Soldier Name : Jerry Base Code : A111 Department ID : D335 Rank : 1LT

[Back to Home](#)

Find Soldier Jimmy

Find Soldier's Record:

Soldier's ID: Soldier's Name: Base Code: Department ID: Rank:

[Back to Home](#)

Find Soldier Cindy

Find Soldier's Record:

Soldier Does not Exist

[Back to Home](#)

Delete Soldier Joe

Soldier has been deleted

[Back to Home](#)

Soldier List after delete

SOLDIER TABLE:

Soldier ID : 00001 Soldier Name : Jimmy Base Code : A132 Department ID : D333 Rank : PVT
Soldier ID : 00002 Soldier Name : John Base Code : A111 Department ID : D335 Rank : PVT
Soldier ID : 00003 Soldier Name : Ogi Base Code : A105 Department ID : D363 Rank : SSG
Soldier ID : 00004 Soldier Name : Simon Base Code : A104 Department ID : D385 Rank : CSM
Soldier ID : 00005 Soldier Name : Tom Base Code : A136 Department ID : D379 Rank : BG
Soldier ID : 00006 Soldier Name : Jerry Base Code : A111 Department ID : D335 Rank : 1LT

[Back to Home](#)

Update Jimmy from Dept_ID D333 to D634

Updating Soldier's Record:

Soldier's ID: 00001

Soldier's Name: Jimmy

Base Code: A132

Department ID: D634

Rank: PVT

Update Soldier's Record

There have some problem that cannot connect to the updateupdatesoldier file.

Update Joe

Updating Soldier's Record:

Soldier Does not Exist