

[5 POINTS] TASK 1: RELATIONAL LOGIC MODELLING

Location (Postcode, address, city)

PK: Postcode

(3) Etc→ We made another table for location and we add the city field to make more realistic the location table.

Employee (alias_employee, DNI*, Full_name*, postcode)

PK: Alias_employee

NN: DNI

NN: Full_name

(1) ATT derivative → alias_employee has been changed from the original model

Hacker(Alias_hacker, dni*, email, Full_name*, phone, Employee_alias, postcode)

PK: Alias_hacker

FK: Employee_alias→ employee (Alias_employee) D:C

NNV: Alias_employee

FK: Postcode→ location (postcode) D:C

NN: DNI

NN: Full_name

(1) ATT derivative → alias_hacker has been changed from the original model

Employee_Collaborate_Employee (Day, Employee_alias)

PK: Day

FK: Employee_alias→ employee (Alias) D:C

AT_Type(Code_type, name, type_description)

PK: Code_type

Attack(Index_at, type_code, timestamp_at, risk, alias_employee)

PK: {Index_at, type_code}

FK: type_code → at_type (code_type) D:C

FK: alias_employee→ Employee (alias_employee)

(3) Etc→ Risk is gonna be enum: high ,medium, low.

Target (ip_target, service, name*, cost)

PK: (Ip_target, service)

UK: name

(3) Etc→ Cost has to be less or equal to 1000000

Effect(at_index, ip_target, eff_description, severity, Code_effect*)

PK: { Index_at, ip_target}

FK: ip_target → target (ip_target,Service) D:C

FK: at_index → attack (index_at) D:C

UK: Code_effect

(3) Etc→ Severity its a % tinyint range less or equal to 100

[2 POINTS] TASK 2: DDL PHYSICAL MODELLING. METADATA CREATION

Specify the necessary statements to incorporate the previous tables to a MySQL DB. Remember to respect a correct order for a sequential operation.

```
DROP DATABASE IF EXISTS security;
CREATE DATABASE security;
USE security;
CREATE TABLE Location (
  Postcode Varchar (255) PRIMARY KEY,
  Address Varchar(255),
  City Varchar(255)
);
CREATE TABLE Employee (
  Alias_employee varchar(255) PRIMARY KEY,
  Dni_employee varchar(255) UNIQUE,
  Full_name varchar(255),
  Location_postcode varchar(255),
  CONSTRAINT emp_loc_fk FOREIGN KEY (location_postcode) REFERENCES Location (Postcode)
  ON UPDATE CASCADE
  ON DELETE CASCADE
);
CREATE TABLE Hacker (
  alias_hacker Varchar (255) PRIMARY KEY,
  name varchar (255),
  email varchar (255),
  alias_employee varchar (255),
  location_postcode varchar(255),
  FOREIGN KEY (location_postcode) REFERENCES Location (postcode)
  ON UPDATE CASCADE
  ON DELETE CASCADE,
  FOREIGN KEY (alias_employee) REFERENCES Employee (Alias_employee)
  ON UPDATE CASCADE
  ON DELETE CASCADE
);
CREATE TABLE Employee_Collaboration (
  employe_alias VARCHAR(255) PRIMARY KEY,
  day_col DATETIME NOT NULL,
  CONSTRAINT day_emp_fk FOREIGN KEY (employe_alias) REFERENCES Employee (Alias_employee)
);
CREATE TABLE Attack_type (
  code_type VARCHAR(255) PRIMARY KEY,
  name VARCHAR(255),
  type_description VARCHAR(255)
);
CREATE TABLE Attack (
  index_at VARCHAR (255),
  type_code VARCHAR (255),
  alias_employee VARCHAR (255),
  timestamp_at DATETIME,
  risk VARCHAR(255),
  CONSTRAINT atc_rsk_ck CHECK (risk IN ('low', 'medium', 'high')),
  CONSTRAINT atc_typ_pk PRIMARY KEY (index_at, type_code),
  CONSTRAINT atc_emp_fk FOREIGN KEY (alias_employee) REFERENCES Employee (Alias_employee),
  CONSTRAINT atc_typ_fk FOREIGN KEY (type_code) REFERENCES Attack_type (code_type)
  ON UPDATE CASCADE
  ON DELETE CASCADE
);
CREATE TABLE Target (
  ip_target VARCHAR(255),
  name_target VARCHAR(255),
  service_target VARCHAR(255),
```

```
cost INT UNSIGNED NOT NULL,  
CONSTRAINT tar_bot_pk PRIMARY KEY (ip_target, service_target)  
);
```

```
CREATE TABLE Effect(  
at_index VARCHAR(255),  
code_effect VARCHAR(255) UNIQUE,  
CONSTRAINT eff_att_pk PRIMARY KEY (at_index, target_ip),  
eff_description VARCHAR (255),  
severity TINYINT UNSIGNED NOT NULL,  
CONSTRAINT ck_eff_per CHECK (severity >=100),  
target_ip VARCHAR (255),  
CONSTRAINT eff_ati_fk FOREIGN KEY (at_index) REFERENCES Attack (index_at)  
ON UPDATE CASCADE  
ON DELETE CASCADE,  
CONSTRAINT eff_tar_fk FOREIGN KEY (target_ip) REFERENCES Target (ip_target)  
ON UPDATE CASCADE  
ON DELETE CASCADE  
);
```

Using vstudio code i create an sql file and importing to mariadb with source in case of an error creating de database we can go to the file modify it and launch it inot mariadb.

[2 POINTS] TASK 3: DDL PHYSICAL MODELLING.**METADATA MODIFICATION**

Specify the necessary statements to perform these modifications in MySQL. Remember to respect a correct order for a sequential operation.

- Modification 1 (Now ATTACKs can be registered without being detected by anyone, participation (1,1) turns (0,1) on Detects.)

Now Detects its 0,1 so we don't need the column of employees, we have to delete the constrain first and later the column,

```
ALTER TABLE Attack DROP CONSTRAINT atc_emp_fk;
ALTER TABLE Attack DROP alias_employee;
```

Modification 2 (The cost cannot be less than 0 or greater than 1000000.)

We need to add a constrain to make a check on cost <=1000000

```
ALTER TABLE Target
ADD CONSTRAINT tar_cos_ck CHECK (cost
<=1000000);
```

So we can add this query and says query ok but if really want to see if the constraint is okay u have to check the information schema and select the table_constrains.

```
MariaDB [security]> ALTER TABLE Effect ADD CONSTRAINT ck_eff_per CHECK (severity <=100);
Query OK, 0 rows affected (0.032 sec)
Records: 0 Duplicates: 0 Warnings: 0
```

def	security	PRIMARY	security	attack
def	security	atc_rsk_ck	security	attack
def	security	atc_emp_fk	security	attack
def	security	atc_typ_fk	security	attack
def	security	PRIMARY	security	attack_type
def	security	PRIMARY	security	effect
def	security	code_effect	security	effect
def	security	ck_eff_per	security	effect
def	security	eff_ati_fk	security	effect
def	security	eff_tar_fk	security	effect
def	security	PRIMARY	security	employee
def	security	Dni_employee	security	employee
def	security	emp_loc_fk	security	employee
def	security	PRIMARY	security	employee_collaboration
def	security	day_emp_fk	security	employee_collaboration
def	security	PRIMARY	security	hacker
def	security	hacker_ibfk_1	security	hacker
def	security	hacker_ibfk_2	security	hacker
def	security	PRIMARY	security	location
def	security	PRIMARY	security	target
def	security	tar_cos_ck	security	target

Modification 3 (Default value for timestamp_at is the current timestamp when that record is created)

We need to change the datatype of the field timestamp_at, it was DATETIME and now we can put timestamp
 ALTER TABLE Attack MODIFY timestamp_at TIMESTAMP;

```
MariaDB [security]> describe Attack;
```

Field	Type	Null	Key	Default	Extra
index_at	varchar(255)	NO	PRI	NULL	
type_code	varchar(255)	NO	PRI	NULL	
timestamp_at	timestamp	NO		current_timestamp()	on update current_timestamp()
risk	varchar(255)	YES		NULL	

4 rows in set (0.010 sec)

Modification 4 (Now the email address of the EMPLOYEES is also stored.)

Now we need to add an email column to the employee table:

ALTER TABLE Employee ADD email varchar(255);

```
MariaDB [security]> ALTER TABLE Employee ADD email varchar(255);
Query OK, 0 rows affected (0.006 sec)
Records: 0 Duplicates: 0 Warnings: 0

MariaDB [security]> describe Employee;
+-----+-----+-----+-----+-----+-----+
| Field          | Type          | Null | Key | Default | Extra |
+-----+-----+-----+-----+-----+-----+
| Alias_employee | varchar(255)  | NO   | PRI | NULL    |       |
| Dni_employee   | varchar(255)  | YES  | UNI | NULL    |       |
| Full_name      | varchar(255)  | YES  |     | NULL    |       |
| Location_postcode | varchar(255) | YES  | MUL | NULL    |       |
| email          | varchar(255)  | YES  |     | NULL    |       |
+-----+-----+-----+-----+-----+-----+
5 rows in set (0.009 sec)
```

Modification 5 (Deleting an ATTACK will erase its corresponding EFFECTS and the information on which TARGETs it was directed at (not the TARGETs themselves))

No entendia bien este punto es decir creo que hice la tabla de Effect pensando en esto porque cuando he echo el delete del ataque en el ultimo punto se ha borrado todo menos la informacion del target, puede ser que haya interpretado mal la tabla de efectos en el punto anterior y este mal maquetada de esa forma, pero al borrar el ataque con indice 35 no se ha borrado la informacion del objetivo.

```
MariaDB [security]> DELETE FROM Attack WHERE index_at='35';
Query OK, 1 row affected (0.005 sec)

MariaDB [security]> select * from Attack;
Empty set (0.000 sec)

MariaDB [security]> select * from Effect;
Empty set (0.000 sec)

MariaDB [security]> select * from Target;
+-----+-----+-----+-----+
| ip_target | name_target | service_target | cost |
+-----+-----+-----+-----+
| 213.0.87.46 | oracle      | Web Server      | 25000 |
+-----+-----+-----+-----+
1 row in set (0.000 sec)
```

Specify the necessary statements to perform these data operations in MySQL. Remember to respect a correct order for a sequential operation.

- Operation 1 (Create 2 AT_TYPES with this information: code_type "PH951" name "Phishing" type_description "A fraudulent attempt to obtain sensitive information by disguising oneself as a trustworthy entity". code_type "DD468" name "DDoS" type_description "A distributed denial-of-service attack that overwhelms a target with a large amount of traffic from multiple sources".)

/// We use

```
INSERT INTO Attack_type (code_type, name, type_description)
```

```
VALUES ('PH951', 'Phishing', 'A fraudulent attempt to obtain sensitive information by disguising oneself as a trustworthy entity');
```

```
INSERT INTO Attack_type ('DD468', 'Ddos', 'A distributed-denial-of-service attack that overwhelms a target with a large amount of traffic from multiple sources');
```

```
MariaDB [security]> SELECT * FROM Attack_type;
+-----+-----+-----+
| code_type | name | type_description |
+-----+-----+-----+
| DD468 | Ddos | A distributed-denial-of-service attack that overwhelms the target with a large amount of traffic from multiple sources |
| PH951 | Phising | A fraudulent attempt to obtain sensitive information by disguising oneself as a trustworthy entity |
+-----+-----+-----+
2 rows in set (0.000 sec)
```

- Operation 2 (There has been a "DDoS" ATTACK with index_at "35" on the TARGET with ip_target "213.0.87.46" and service "Web Server". The only EFFECT caused by the ATTACK has been a 30 minutes of web service denial with a severity of 25%.)

/// We have to insert the attack info, target info and effect info on all fields some can be null but others cant.

```
INSERT INTO Attack VALUES ('35', 'DD468', '2023-12-26 20:50:34', 'medium');
```

```
INSERT INTO Target VALUES ('213.0.87.46', 'oracle', 'Web Server', '25000');
```

```
INSERT INTO Effect VALUES ('35', 'E123', '30 minutes of web server down', '25', '213.0.87.46');
```

```
MariaDB [security]> INSERT INTO Attack VALUES ('35', 'DD468', '2023-12-26 20:50:34', 'medium');
Query OK, 1 row affected (0.004 sec)
```

```
MariaDB [security]> select * from Attack
-> ;
+-----+-----+-----+-----+
| index_at | type_code | timestamp_at | risk |
+-----+-----+-----+-----+
| 35 | DD468 | 2023-12-26 20:50:34 | medium |
+-----+-----+-----+-----+
1 row in set (0.000 sec)
```

```
MariaDB [security]> INSERT INTO Target VALUES ('213.0.87.46', 'oracle', 'Web Server', '25000 ');
Query OK, 1 row affected (0.005 sec)
```

```
MariaDB [security]> select * from Target;
+-----+-----+-----+-----+
| ip_target | name_target | service_target | cost |
+-----+-----+-----+-----+
| 213.0.87.46 | oracle | Web Server | 25000 |
+-----+-----+-----+-----+
1 row in set (0.001 sec)
```

```
MariaDB [security]> INSERT INTO Effect VALUES ('35', 'E123', '30 minutes of web server down', '25', '213.0.87.46 ');
Query OK, 1 row affected (0.005 sec)
```

```
MariaDB [security]> select * from Effect;
+-----+-----+-----+-----+-----+
| at_index | code_effect | eff_description | severity | target_ip |
+-----+-----+-----+-----+-----+
| 35 | E123 | 30 minutes of web server down | 25 | 213.0.87.46 |
+-----+-----+-----+-----+-----+
1 row in set (0.000 sec)
```

UPDATE the severity of that EFFECT to 30%.

Now we use the command:

UPDATE Effects SET at_index='35' WHERE severity='30' ;

```
MariaDB [security]> UPDATE Effect SET severity='30' WHERE at_index='35 ' ;
Query OK, 1 row affected (0.001 sec)
Rows matched: 1  Changed: 1  Warnings: 0

MariaDB [security]> select * from Effect;
+-----+-----+-----+-----+-----+
| at_index | code_effect | eff_description          | severity | target_ip |
+-----+-----+-----+-----+-----+
| 35      | E123      | 30 minutes of web server down | 30      | 213.0.87.46 |
+-----+-----+-----+-----+-----+
1 row in set (0.000 sec)
```

DELETE that ATTACK

Now we use the command:

DELETE FROM Attack WHERE index_at='35';

```
MariaDB [security]> DELETE FROM Attack WHERE index_at='35';
Query OK, 1 row affected (0.005 sec)

MariaDB [security]> select * from Attack;
Empty set (0.000 sec)

MariaDB [security]> select * from Effect;
Empty set (0.000 sec)

MariaDB [security]> select * from Target;
+-----+-----+-----+-----+
| ip_target | name_target | service_target | cost |
+-----+-----+-----+-----+
| 213.0.87.46 | oracle      | Web Server      | 25000 |
+-----+-----+-----+-----+
1 row in set (0.000 sec)
```