

IT Academy Logo https://github.com/ciberzerone/itAcademyData_Scientist/blob/main/sprint2/imagen/logoIT.png

[Ver script de SQL https://github.com/ciberzerone/itAcademyData_Scientist/blob/main/sprint2/sql/bbdd.sql

[Ver Sprint 3 en PDF https://github.com/ciberzerone/itAcademyData_Scientist/blob/main/sprint2/pdf/sprint2.pdf

Nivel 1 https://github.com/ciberzerone/itAcademyData_Scientist/blob/main/sprint2/imagen/nivel1.PNG

Ejercicio 1:

El esquema creado incluye dos tablas principales: `company` y `transaction`. A continuación, se describen estas tablas y las variables que contienen, así como las relaciones entre ellas.

Diseño de la Tabla `credit_card`

Esta tabla almacena información sobre las tarjetas de créditos involucradas en las transacciones. Contiene los siguientes campos:

Columnas:

- **id**: Identificador único de la tarjeta de crédito.
- **iban**: Número IBAN de la tarjeta.
- **pan**: Número PAN de la tarjeta.
- **pin**: PIN de la tarjeta.
- **cvv**: Código CVV de la tarjeta.
- **expiring_date**: Fecha de vencimiento de la tarjeta.

Claves y Relaciones:

- **id**: Clave primaria.

Sql de tabla `credit_card`

```
```sql -- Creacion la tabla creditcard CREATE TABLE IF NOT EXISTS
creditcard (id VARCHAR(15) PRIMARY KEY, iban VARCHAR(34), pan
VARCHAR(19), pin VARCHAR(4), cvv VARCHAR(4), expiring_date DATE ;
```
```

Introducir la data en la `credit_card`

Al insertar los datos, arroja un error por el formato de los datos `expiringdate`, una solución podría ser utilizar `varchar`, pero se utilizó

STRTO_DATE(cadena, formato que convierte los datos de una cadena a un formato de fecha

```
sql -- convertirá la cadena '10/30/22' en la fecha 2022-10-30.
('CcU-2938', 'TR301950312213576817638661', '5424465566813633',
'3257', '984', STR_TO_DATE('10/30/22', '%m/%d/%y', ... los demas
registros ...
```

Identificar de forma única cada tarjeta en la credit_card

```
```sql id VARCHAR(15 PRIMARY KEY,
```
```

Establecer relación con la tabla transaction

```
```sql ALTER TABLE transaction ADD CONSTRAINT fkcreditcard FOREIGN
KEY (creditcardid REFERENCES credit_card(id;
```
```

Diagrama del Esquema

A continuación se presenta un diagrama que ilustra la relación entre las tablas user, credit_card, company y transaction:

Diagrama del Esquemahttps://github.com/ciberzerone/itAcademyData_Scientist/blob/main/sprint3/imagen/eje01ER.PNG

Ejercicio 2: Actualizar número de cuenta del usuario con ID CcU-2938.

**Consulta 1: La información que debe mostrarse para este registro es:
R323456312213576817699999.**

Sql es Mostrar ID CcU-2938

```
sql USE transactions; SELECT * FROM credit_card WHERE id =
'CcU-2938';
```

Imagen de Mostrar ID CcU-2938:

Mostrar ID CcU-2938 <https://github.com/ciberzerone/itAcademyDataScientist/blob/main/sprint3/imagen/eje02actualizar01.PNG>

Sql Actualizacion del campo Iban del id CcU-2938

```
sql USE transactions; UPDATE credit_card SET iban =  
'R323456312213576817699999' WHERE id = 'CcU-2938';
```

Imagen de sql de la Actualizacion del campo Iban del id CcU-2938:

sql de la Actualizacion del campo Iban del id CcU-2938 <https://github.com/ciberzerone/itAcademyDataScientist/blob/main/sprint3/imagen/eje02actualizar02.PNG>

Sql Mostrar campo Iban del id CcU-2938 actualizado

```
sql SELECT * FROM credit_card WHERE id = 'CcU-2938';
```

Imagen de Sql Mostrar campo Iban del id CcU-2938 actualizado:

Sql Mostrar campo Iban del id CcU-2938 actualizado <https://github.com/ciberzerone/itAcademyDataScientist/blob/main/sprint3/imagen/eje02actualizar03.PNG>

Ejercicio 3: En la tabla transaction ingresar un nuevo usuario.

Ingresar los siguientes datos:

Imagen de Datos a ingresar:

Mostrar ID CcU-2938 <https://github.com/ciberzerone/itAcademyDataScientist/blob/main/sprint3/imagen/eje03insertar01.PNG>

Sql de Insert de los datos en la tabla 'transaction' pero a la vez debe ingresarse datos en tablas: 'company' 'user' 'credit_card'.

```
`` sql -- Deshabilitar las comprobaciones de claves foráneas temporalmente  
SET FOREIGNKEYCHECKS = 0;
```

```
-- Insertar un nuevo registro en la tabla 'company' INSERT INTO company
(id, company_name, phone, email, country, website VALUES ('b-9999', 'Ey
tacademy', '1234567890', 'contact@eit.com', 'CountryName', 'http://
example.com');

-- Insertar un nuevo registro en la tabla 'user' INSERT INTO user (id, name,
surname, phone, email, birthdate, country, city, postalcode, address VALUES
(9999, 'John', 'Doe', '9876543210', 'john.doe@example.com', '1990-01-01',
'barca', 'barcasas', '12345', '123 Main St';

-- Insertar un nuevo registro en la tabla 'creditcard' INSERT INTO
creditcard (id, iban, pan, pin, cvv, expiringdate VALUES ('CcU-9999',
'TR301950312213576817638661', '5424465566813633', '3257', '984',
STRTO_DATE('10/30/22', '%m/%d/%y');

-- Insertar un nuevo registro en la tabla 'transaction' INSERT INTO
transaction (id, creditcardid, companyid, userid, lat, longitude, timestamp,
amount, declined VALUES ('108B1D1D-5B23-A76C-55EF-C568E49A99DD',
'CcU-9999', 'b-9999', 9999, 829.999, -117.999, NOW(), 111.11, 0;

-- Habilitar las comprobaciones de claves foráneas SET
FOREIGNKEYCHECKS = 1;

...
```

Imagen de Mostrar ID CcU-2938:

Mostrar ID CcU-2938 <https://github.com/ciberzerone/itAcademyDataScientist/blob/main/sprint3/imagen/eje03insertar02.PNG>

Explicar código:

- **SET FOREIGNKEYCHECKS = 0;** Para evitar el Error Code: 1452. Cannot add or update a child row: a foreign key constraint fails (transactions.transaction, CONSTRAINT transaction_ibfk_1 FOREIGN KEY (company_id) REFERENCES company (id*)/ se debe usar SET FOREIGNKEYCHECKS = 0; que evita evitar errores de restricciones de claves foráneas durante las inserciones.

Ejercicio 4: Desde recursos humanos te solicitan eliminar la columna pan de la tabla credit_card. Recuerda mostrar el cambio realizado.

Eliminar la columna pan de la tabla credit_card:

Sql para eliminar

```
sql ALTER TABLE credit_card DROP COLUMN pan;
```

Imagen Eliminar columna pan:

Mostrar ID CcU-2938 <https://github.com/ciberzerone/itAcademyDataScientist/blob/main/sprint3/imagen/eje04eliminar01.PNG>

Explicar codigo:

Para modificar la tabla 'credit_card' se utilizar 'alter' y para eliminar la columna 'pan' se utiliza drop

Mostrar el cambio realizado:

Sql para mostrar los cambios en la tabla credit_card

```
```sql  
DESCRIBE credit_card;
```
```

Imagen de mostrando los cambios realizados en la tabla credit_card:

Mostrar ID CcU-2938 <https://github.com/ciberzerone/itAcademyDataScientist/blob/main/sprint3/imagen/eje04eliminar02.PNG>

Explicar codigo:

```
-- para mostrar describe credit_card nos muestra la estructura
```