# Database Coursework 1 Group Report
# Group TWELVE

## Sakila

Group Members

| | |
|---|---|
| Kok Yong En | 10346769 |
| Lim Hui Xing | 20014709 |
| Lim Kelly | 20024694 |
| Sin Kai Ling | 10341812 |
| Tan Song Ning | 10281222 |
| Tan Yi-Wei | 20033760 |

## Introduction

The dataset Sakila is a group of data related to a film rental shop. The dataset is given with its entities and attributes defined. The relation for all the entities with their attributes have been provided as well.

## Reason behind the design of database

Database allows us to store information, and provides us efficiency in manipulating the data, which includes organizing, retrieving, updating the data.

One of the major reasons behind this database design is to reduce redundancy and inconsistency. Without a proper design of database, duplicated information could exist in several places (files). Redundancy will lead to higher cost of storage and access cost. If the various copies of information does not share the same information, inconsistencies arise.

Besides that, with the relationships between relations clearly identified (primary keys and foreign keys), integrity problems could be avoided. The connections between the tables enables us to put constraints on to the information. For our database Sakila, foreign keys that exist in the table `payment`, are `SET NULL` on delete, but the foreign keys in other tables such as `film, customers, address, payment` etc. `CASCADE` on delete and on update. This is because the records on payment are meant to be kept even if the referenced entities no longer exist.

## Connection between the Tables

The primary keys and the foreign keys for the dataset are identified.

| Candidate keys | Primary key in the relation | Foreign key in the relation |
|---|---|---|
| actor_id | actor | film_actor |
| address_id | address | customer, store, staff |
| city_id | city | address |
| country_id | country | city |
| customer_id | customer | payment, rental |

| | | |
|---|---|---|
| film_id | film | film_actor, inventory |
| language_id | language | film |
| inventory_id | inventory | rental |
| payment_id | payment | |
| rental_id | rental | payment |
| store_id | store | customer, inventory, staff |
| staff_id | staff | payment, rental |

## Data types

Primary keys are usually in the form of ID. The most convenient data type for ID is integer types. Thus, for numerics data with lesser than 3 digits, the data type `tinyint` is used. For all the fields that are related to numerics which have lesser than 5 digits, the data type `smallint` is used. For numerics data with lesser than 8 digits, the data type `mediumint` is used. For numerics data with lesser than 11 digits, the data type `int` is used. Fields like 'active' which have boolean values like True(1) or False(0) use data type `tinyint` as well.

The fields which are related to names, for instance, customer names, street names, city names, country names, category names or usernames uses the data type `varchar`. Other fields such as passwords, email and descriptions for the film also use the data type `varchar`. Varchar means a variable character field which can hold characters and numerics. However, the field description is assigned the data type `text`. This is due to descriptions are usually lengthy and that data type can hold up to 65535 characters.

Fields which are associated with prices such as amount and cost uses the data type `decimal`. These fields are floating point numbers are the precision up until two decimal points.

Fields which records the time and date about the creation time or its last update use the data type `timestamp`. This data type supports the format of YYYY-MM-DD HH:MM:SS.

The field ‘rating’ in the table ‘film’ uses the data type `enum`, as the ratings for each film can only be ‘G', 'PG', 'PG-13', 'R', or 'NC-17'. Only one option in the set can be used.

The field ‘special features’ in the table ‘film’ uses the datatype `Set`, as special features can be one or more options from “Trailers, Deleted Scenes, Commentaries, Behind the Scenes”.

## Relationships

Every film is acted by multiple actors, every actor is involved in many films. Thus, entity actor and entity film have a many-to-many relationship.

There are several copies of one same film in the inventory, each of the copies are given a unique ID. Thus, film and inventory share a one-to-many relationship.

There are many films which use the same language, but each film uses one language only. Language and film share a one-to-many relationship.

Each category has multiple film, every film has only one category. Category and film has a one-to-many relationship.

Every customer has only one address. Every staff has one address. Every store has one address. Every address is only owned by one entity. Thus, customer, staff, store share a one-to-one relationship with address.

Each city contains many addresses, each addresses only exist in one city. City and address have a one-to-many relationship.
Each country contains many cities, every city is situated in one country only. Thus country and city have a one-to-many relationship.

Each customer has made one or more payments, each payment can only be made by one customer.Customer and payment have a one-to-many relationship.

Each customer has made many rentals, each rental is only made by one customer. Customer and rental have a one-to-many relationship.

Every copy of film in the inventory can be rented one or multiple times. Each rental ID is associated with one copy of the film in the inventory only(inventory ID). Thus, inventory and rental have a one-to-many relationship.

Every rental has one or many payments. Each payment is made for one rental only. Rental and payment have a one-to-many relationship.

Every store is visited by many customers, every customer visits only one store. Store and customer have a one-to-one relationship.

Every store has multiple inventory ID, but each inventory ID is kept in one store only. Store and inventory have a one-to-many relationship.

Every store has many staff, every staff works in one store only. Store and staff have a one-to-many relationship.

Every staff has managed many payments, each payment is handled by one staff only. Staff and payment have a one-to-many relationship.

Every staff has managed many rentals, every rental is handled by one staff only. Staff and rental have a one-to-many relationship.

## Errors Found and Solutions

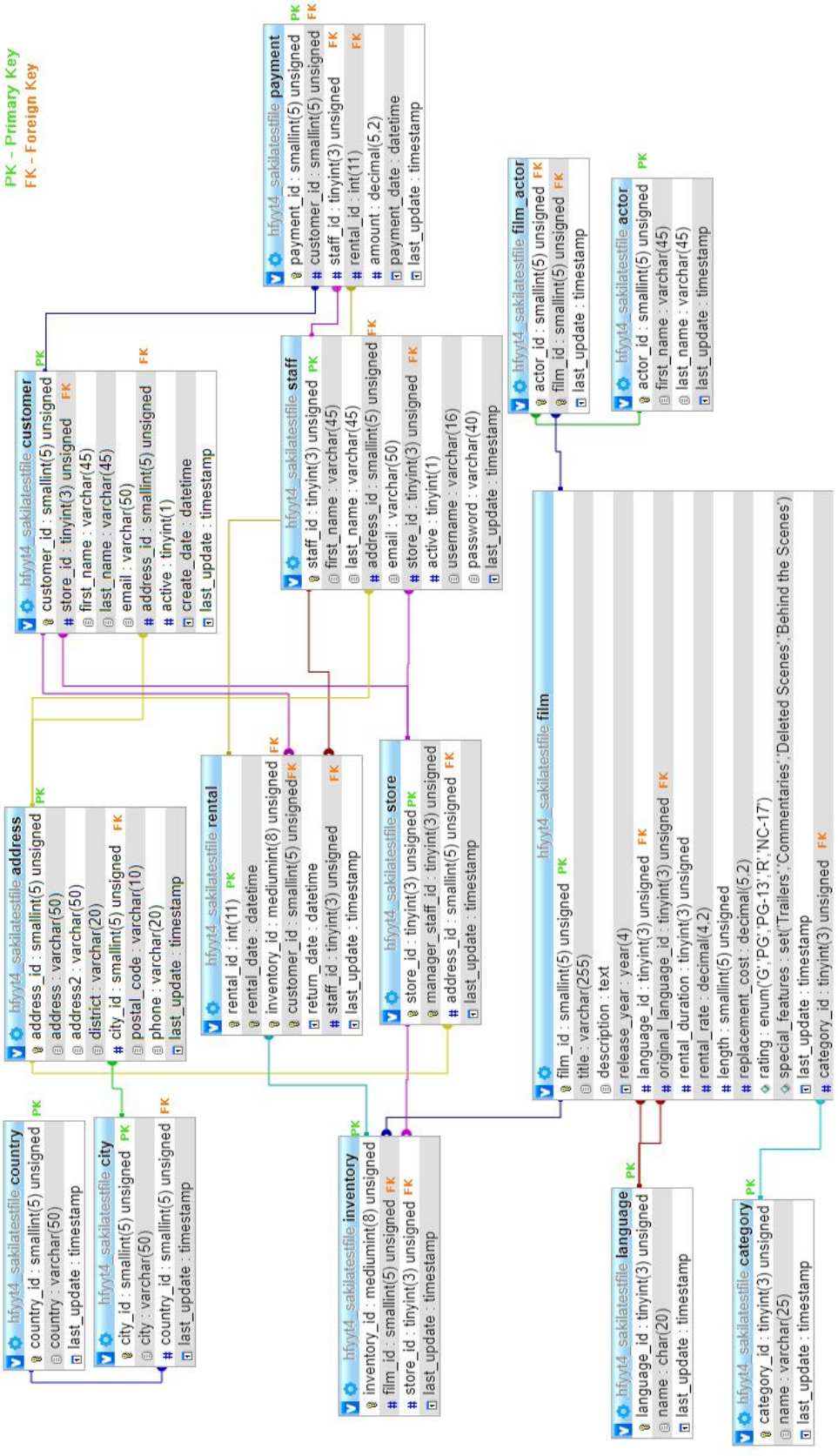
In staff relation, the table contains meaningless sequence of characters and numbers. The table also include a picture column which is not a supported data type in database management system. Therefore, all the unnecessary datas are removed from the relation.

There are data redundancies in table ‘film_text’ and table ‘category’, so both the tables are removed and a new column category_id is added into the table ‘film’.

## Completeness

There are some fields in the data set which are null, which are ‘Address2’ from ‘address’ table, ‘original language id’ from from ‘film’ table. It is assumed that there is some missing data from the data set.

## ER Diagram


PK - Primary Key
FK - Foreign Key
hfyt4_sakilatestfile payment
payment_id : smallint(5) unsigned
customer_id : smallint(5) unsigned
staff_id : tinyint(3) unsigned
rental_id : int(11)
amount : decimal(5,2)
payment_date : datetime
last_update : timestamp
hfyt4_sakilatestfile customer
customer_id : smallint(5) unsigned
store_id : tinyint(3) unsigned
first_name : varchar(45)
last_name : varchar(45)
email : varchar(50)
address_id : smallint(5) unsigned
active : tinyint(1)
create_date : datetime
last_update : timestamp
hfyt4_sakilatestfile staff
staff_id : tinyint(3) unsigned
first_name : varchar(45)
last_name : varchar(45)
address_id : smallint(5) unsigned
email : varchar(50)
store_id : tinyint(3) unsigned
active : tinyint(1)
username : varchar(16)
password : varchar(40)
last_update : timestamp
hfyt4_sakilatestfile film_actor
actor_id : smallint(5) unsigned
film_id : smallint(5) unsigned
last_update : timestamp
hfyt4_sakilatestfile actor
actor_id : smallint(5) unsigned
first_name : varchar(45)
last_name : varchar(45)
last_update : timestamp
hfyt4_sakilatestfile address
address_id : smallint(5) unsigned
address : varchar(50)
address2 : varchar(50)
district : varchar(20)
city_id : smallint(5) unsigned
postal_code : varchar(10)
phone : varchar(20)
last_update : timestamp
hfyt4_sakilatestfile rental
rental_id : int(11)
rental_date : datetime
inventory_id : mediumint(8) unsigned
customer_id : smallint(5) unsigned
return_date : datetime
staff_id : tinyint(3) unsigned
last_update : timestamp
hfyt4_sakilatestfile store
store_id : tinyint(3) unsigned
manager_staff_id : tinyint(3) unsigned
address_id : smallint(5) unsigned
last_update : timestamp
hfyt4_sakilatestfile film
film_id : smallint(5) unsigned
title : varchar(255)
description : text
release_year : year(4)
language_id : tinyint(3) unsigned
original_language_id : tinyint(3) unsigned
rental_duration : tinyint(3) unsigned
rental_rate : decimal(4,2)
length : smallint(5) unsigned
replacement_cost : decimal(5,2)
rating : enum('G','PG','PG-13','R','NC-17')
special_features : set('Trailers','Commentaries','Deleted Scenes','Behind the Scenes')
last_update : timestamp
category_id : tinyint(3) unsigned
hfyt4_sakilatestfile country
country_id : smallint(5) unsigned
country : varchar(50)
last_update : timestamp
hfyt4_sakilatestfile city
city_id : smallint(5) unsigned
city : varchar(50)
country_id : smallint(5) unsigned
last_update : timestamp
hfyt4_sakilatestfile inventory
inventory_id : mediumint(8) unsigned
film_id : smallint(5) unsigned
store_id : tinyint(3) unsigned
last_update : timestamp
hfyt4_sakilatestfile language
language_id : tinyint(3) unsigned
name : char(20)
last_update : timestamp
hfyt4_sakilatestfile category
category_id : tinyint(3) unsigned
name : varchar(25)
last_update : timestamp

**SQL Statements**

**Delete**

```
DELETE FROM actor WHERE actor_id='$actor_id'
DELETE FROM address WHERE address_id='$address_id'
DELETE FROM category WHERE category_id='$category_id'
DELETE FROM city WHERE city_id='$city_id'
DELETE FROM country WHERE country_id='$country_id'
DELETE FROM customer WHERE customer_id='$customer_id'
DELETE FROM film WHERE film_id='$film_id'
DELETE FROM film_actor WHERE actor_id='$actor_id' AND
film_id='$film_id'
DELETE FROM inventory WHERE inventory_id='$inventory_id'
DELETE FROM language WHERE language_id='$language_id'
DELETE FROM payment WHERE payment_id='$payment_id'
DELETE FROM rental WHERE rental_id='$rental_id'
DELETE FROM staff WHERE staff_id='$staff_id'
DELETE FROM store WHERE store_id='$store_id'
```

**Insert**

```
INSERT INTO actor
VALUES(NULL,'$first_name','$last_name',CURRENT_TIMESTAMP)
INSERT INTO address
VALUES(NULL,'$address','$address2','$district','$city_id','$postal_code','
$phone',CURRENT_TIMESTAMP)
INSERT INTO category
VALUES(NULL,'$name',CURRENT_TIMESTAMP)
INSERT INTO city
VALUES(NULL,'$city','$country_id',CURRENT_TIMESTAMP)
INSERT INTO country
VALUES(NULL,'$country',CURRENT_TIMESTAMP)
INSERT INTO customer
VALUES(NULL,'$store_id','$first_name','$last_name','$email','$address_
id','$active',CURRENT_TIMESTAMP,CURRENT_TIMESTAMP)
```

```
INSERT INTO film
VALUES(NULL,'$title','$description','$release_year','$language_id','$original_language_id','$rental_duration','$rental_rate','$length','$replacement_cost','$rating','$special_features',CURRENT_TIMESTAMP,'$category_id')
INSERT INTO film_actor
VALUES('$actor_id','$film_id',CURRENT_TIMESTAMP)
INSERT INTO inventory
VALUES(NULL,'$film_id','$store_id',CURRENT_TIMESTAMP)
INSERT INTO language
VALUES(NULL,'$name',CURRENT_TIMESTAMP)
INSERT INTO payment
VALUES(NULL,'$customer_id','$staff_id','$rental_id','$amount',CURRENT_TIMESTAMP,CURRENT_TIMESTAMP)
INSERT INTO rental
VALUES(NULL,'$rental_date','$inventory_id','$customer_id','$return_date','$staff_id',CURRENT_TIMESTAMP)
INSERT INTO staff
VALUES(NULL,'$first_name','$last_name','$address_id',
'$email','$store_id','$active','$username','$password',CURRENT_TIMESTAMP)
INSERT INTO store
VALUES(NULL,'$manager_staff_id','$address_id',CURRENT_TIMESTAMP)
```

**Select**

```
SELECT * FROM actor
SELECT * FROM address
SELECT * FROM category
SELECT * FROM city
SELECT * FROM country
SELECT * FROM customer
SELECT * FROM film
SELECT * FROM film_actor
SELECT * FROM inventory
SELECT * FROM language
```

SELECT * FROM payment
SELECT * FROM rental
SELECT * FROM staff
SELECT * FROM store

**Update**

UPDATE actor SET first_name='$first_name',last_name='$last_name' WHERE actor_id='$actor_id'
UPDATE address SET address='$address',address2='$address2', district='$district',city_id='$city_id',postal_code='$postal_code',phone='$phone', last_update=CURRENT_TIMESTAMP WHERE address_id='$address_id'
UPDATE category SET name='$name',last_update=CURRENT_TIMESTAMP WHERE category_id='$category_id'
UPDATE city SET city='$city',country_id='$country_id', last_update=CURRENT_TIMESTAMP WHERE city_id='$city_id'
UPDATE country SET country='$country', last_update=CURRENT_TIMESTAMP WHERE country_id='$country_id'
UPDATE customer SET store_id='$store_id',first_name='$first_name',last_name='$last_name',email='$email',address_id='$address_id',active='$active',create_date='$create_date',last_update=CURRENT_TIMESTAMP WHERE customer_id='$customer_id'
UPDATE film SET title='$title',description='$description', release_year='$release_year',language_id='$language_id', original_language_id='$original_language_id',rental_duration='$rental_duration', rental_rate='$rental_rate',length='$length', replacement_cost='$replacement_cost',rating='$rating',special_features='$special_features',last_update=CURRENT_TIMESTAMP,category_id='$category_id' WHERE film_id='$film_id'
UPDATE film_actor SET actor_id='$actor_id',film_id='$film_id', last_update=CURRENT_TIMESTAMP WHERE actor_id=$prevactor_id AND film_id='$prevfilm_id'
UPDATE inventory SET film_id='$film_id',store_id='$store_id',

```
last_update=CURRENT_TIMESTAMP WHERE
inventory_id='$inventory_id'
UPDATE language SET name='$name',
last_update=CURRENT_TIMESTAMP WHERE
language_id='$language_id'
UPDATE payment SET customer_id='$customer_id',staff_id='$staff_id',
rental_id='$rental_id',amount='$amount',payment_date='$payment_date'
,last_update=CURRENT_TIMESTAMP WHERE
payment_id='$payment_id'
UPDATE rental SET
rental_date='$rental_date',inventory_id='$inventory_id',
customer_id='$customer_id',return_date='$return_date',staff_id='$staff_i
d',last_update=CURRENT_TIMESTAMP WHERE rental_id='$rental_id'
UPDATE staff SET first_name='$first_name',last_name='$last_name',
address_id='$address_id',email='$email',store_id='$store_id',active='$ac
tive',username='$username',password='$password',last_update=CURR
ENT_TIMESTAMP WHERE staff_id='$staff_id'
UPDATE store SET
manager_staff_id='$manager_staff_id',address_id='$address_id',
last_update=CURRENT_TIMESTAMP WHERE store_id='$store_id'
```