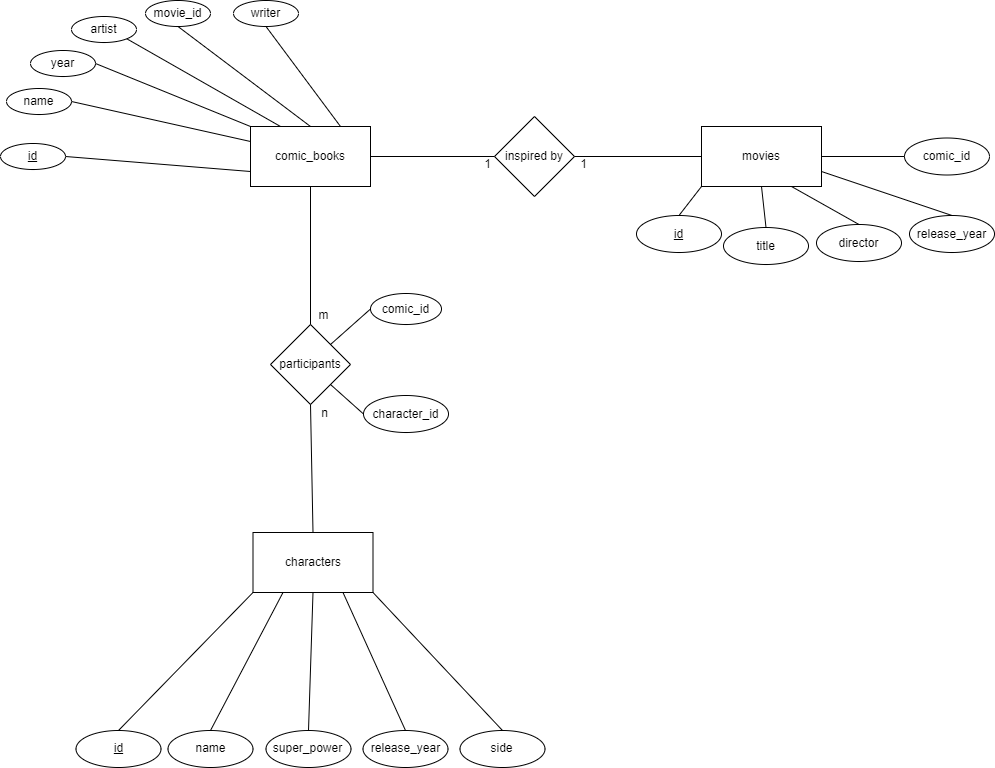
# Programming Assignment 2 Report

|  |  |
| --- | --- |
| Student(s): | Irmantas Venckus – [iv222dn@student.lnu.se](mailto:iv222dn@student.lnu.se)  Hussein Lakis – hl223fk@student.lnu.se |

## Project Idea

For our assignment we have designed a database, which stores information about comic books its characters and the movies that are based on them. The targeted audience are the comic book/ movie fans which are interested in finding out more information about their favorite movie/comic book. Examples of usage: finding out how many movies does a particular hero has, finding out what kind of characters are mentioned in a comic book.

## Schema Design



The E/R diagram contains 3 main entities comic\_books, movies and characters. The main entity comic\_books is connected by 2 relationships. The first one is connection with characters it is connected by a relationship “participant”, which links each comic book with characters that are defined in it by their id numbers. The described relationship has a many to many relation, because a single comic book can contain multiple characters and a single character can be featured in multiple books. The second relationship is “inspired by” which is a one to one relationship. Though in reality it is a many to many relationship and we are aware of it, but since it is a limited database and our used examples do not contain these scenarios we decided to use a one to one relation.

because there can be multiple movies inspired by a single comic book and a single movie can be inspired by multiple comic books. Unlike the participants

## SQL Queries

Here you present and discuss the most interesting queries. Make sure you have 5 of them at least and check the specification in the assignment sheet. One example is found below:

Q: **List the name, last name and job title of the employee from a given city.**

The following query is a multirelation query and uses *JOIN.* We pass the argument of the city name (marked with ? in the query) and the query should give us all the employees of the corresponding shop. We join table *Employees* on table *Shops* by matching the *Shops.ID* to the foreign key *Employees.shopID*

SELECT firstname, lastname, jobtitle

FROM Employees

JOIN Shops ON Employees.shopID = Shops.ID

WHERE city=?;

## Discussion and Resources

Here you can write anything you might think it is important and provide the link to the required resources. For example:

We had issues with the missing and inconsistent data. We decided to remove/insert NULLS in the missing/corrupted attributes/tuples…...

The project uses xyz library, please check readme.txt for installation details.

Source code: [github/... link]

Video demonstration: [youtube/vimeo/... link]

# Changelog

|  |  |  |
| --- | --- | --- |
| Person | Task | Date |
| Ilir | Setting-up server environment and Git repository | 2018-08-20 |
| Ilir | Implemented module for loading the data | 2018-08-20 |
| Maria | Designing and implementing the home-page | 2018-08-20 |
| Maria | Implementing Ouath authentication for Twitter | 2018-08-21 |
| Maria | Documented my changes/contributions in the assignment report | 2018-08-21 |
| Ilir | Documented my changes/contributions in the assignment report | 2018-08-21 |
|  |  |  |