

Assignment 2

MyMyUNSW Schema

Last updated: **Sunday 20th March 11:26pm**

Most recent changes are shown in **red** ... older changes are shown in **brown**.

[\[Assignment Spec\]](#) [\[Database Design\]](#) **[\[SQL Schema\]](#)** [\[Testing\]](#) [\[Sample Outputs\]](#) [\[Fixes+Updates\]](#)

database/schema.sql

```
-- COMP3311 22T1 Ass2 ... schema for IMDB data

create domain Counter integer check (value >= 0);
-- these are ideal domain definitions
-- create domain Minutes integer check (value > 0);
-- create domain YearType integer check (value > 1800);
-- unfortunately, the data in IMDB doesn't agree
-- so we use these instead
create domain Minutes integer check (value >= 0);
create domain YearType integer check (value >= 0);

create table Movies (
    id            integer,
    title         text not null,
    orig_title    text,
    start_year    YearType not null,
    end_year      YearType,
    runtime       Minutes,
    rating        float,
    nvotes        Counter,
    primary key (id)
);

create table Movie_genres (
    movie_id      integer, -- not null because PK
    genre         text,    -- not null because PK
    foreign key (movie_id) references Movies(id),
    primary key (movie_id, genre)
);

create table Aliases (
    id            integer,
    movie_id      integer not null,
    ordering       Counter not null,
    local_title   text not null,
    region        char(4),
    language      char(4),
    extra_info    text,
    foreign key (movie_id) references Movies(id),
    primary key (id)
);

create table Names (
```

```
        id            integer,
        name          text not null,
        birth_year    YearType, -- ideally, not null
        death_year    YearType,
        primary key (id)
    );

create table Principals (
    movie_id    integer, -- not null because PK
    ordering    Counter, -- not null because PK
    name_id     integer not null,
    foreign key (movie_id) references Movies(id),
    foreign key (name_id) references Names(id),
    primary key (movie_id, ordering)
);

create table Acting_roles (
    movie_id    integer, -- not null because PK
    name_id     integer, -- not null because PK
    played      text,     -- not null because PK
    foreign key (movie_id) references Movies(id),
    foreign key (name_id) references Names(id),
    primary key (movie_id,name_id,played)
);

create table Crew_roles (
    movie_id    integer, -- not null because PK
    name_id     integer, -- not null because PK
    role        text,     -- not null because PK
    foreign key (movie_id) references Movies(id),
    foreign key (name_id) references Names(id),
    primary key (movie_id,name_id,role)
);
```