

# Movie Rating System

## Closures

### 1. Exercise 1: Creating a Movie

- Create a function `createMovie` that receives a string argument representing the movie title. The function should return an object with:
  - `title` property: This should store the movie title.
  - `addRating` method: This should take a number as a rating (from 1 to 5) and add it to the movie's ratings. Only up to 10 ratings should be added per movie. If more ratings are attempted to be added, it should print a console error message "Maximum number of ratings reached!".
  - `getAverageRating` method: This should return the average of all ratings added to the movie so far. If no ratings were added, it should return 0.

### 2. Exercise 2: Hiding Movie Details

- Create a function `createMovieWithDetails` that receives three arguments: the movie's title (string), the director's name (string), and the release year (number). The function should return an object with:
  - `getDetails` method: This should return an object with the movie's title, director, and release year.
  - `addRating` method: This should work exactly like the `addRating` method from Exercise 1.
  - `getAverageRating` method: This should work exactly like the `getAverageRating` method from Exercise 1.

### 3. Exercise 3: Creating a Movie Database

- Create a function `createMovieDatabase` that returns an object representing a movie database. The object should provide methods to:
  - `addMovie` : This should create a new movie (using the `createMovie` function from Exercise 1), and add it to the database. The title of the movie should be received as an argument.
  - `findMovieByTitle` : This should receive a movie title as an argument and return the movie object with that title from the database.

- `getAverageRatingOfAllMovies` : This should return the average of all the movies' average ratings in the database. If no movies are present in the database, it should return 0.