Library Class:

Just a collection of movies and watchlists. The client can add(remove) movies and watchlists to(from) a library.

Watchlist Class:

Identified by a name, has a collection of movies. Both attributes are private without getters. addMovie(Movie m) does not make a copy of the Movie parameter as Movie is immutable and any legal changes that happen to the movie should also reflect to the watchlist that movie is in. getStudios and getLanguages both return a newly created list with the attribute references uncopied, which is fine since studio and language attributes of class Movie are Strings, hence immutable.

Movie Class:

Identified by a String filePath attribute (unchangeable), has many attributes. Language, studio and format along with the filePath are final. A default title will be assigned to a Movie at instantiation, the private attribute can later be changed via class method. The constructor assigns the default title via the filename given by the filePath; it also assigns the format of the movie. Custom information can be added/updated or removed one at a time via appropriate methods that take 2 Strings as arguments (key, value pair). There are getters for title, format and custom info. checkValidity also works as a getter and used in almost every method to unsure the attribute is up-to-date. Used design by contract to ensure some parameter specifications.

Format Enum:

An enum for the accepted movie formats.

Diagram

Description automatically generated

Object Diagram of a little simplified version of my Client code