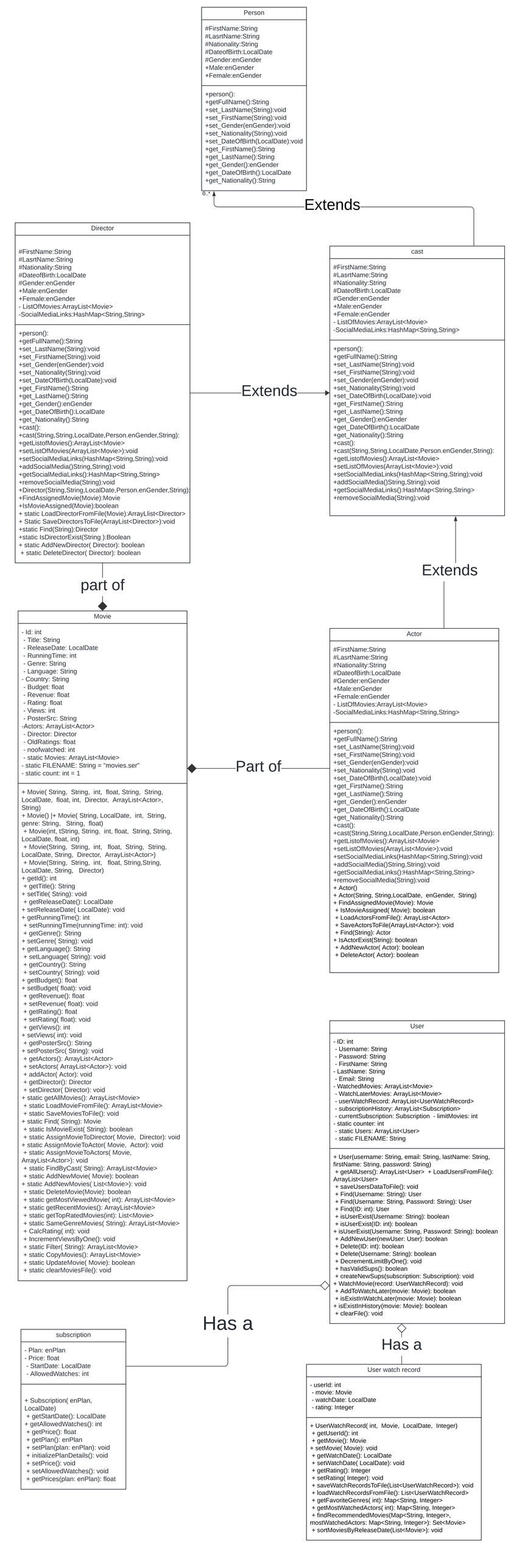
Person
#FirstName:String
#LasrtName:String
#Nationality:String
#DateofBirth:LocalDate
#Gender:enGender
+Male:enGender
+Female:enGender
+person():
+getFullName():String
+set_LastName(String):void
+set_FirstName(String):void
+set_Gender(enGender):void
+set_Nationality(String):void
+set_DateOfBirth(LocalDate):void
+get_FirstName():String
+get_LastName():String
+get_Gender():enGender
+get_DateOfBirth():LocalDate
+get_Nationality():String
0..*
Extends
Director
#FirstName:String
#LasrtName:String
#Nationality:String
#DateofBirth:LocalDate
#Gender:enGender
+Male:enGender
+Female:enGender
- ListOfMovies:ArrayList<Movie>
-SocialMediaLinks:HashMap<String,String>
+person():
+getFullName():String
+set_LastName(String):void
+set_FirstName(String):void
+set_Gender(enGender):void
+set_Nationality(String):void
+set_DateOfBirth(LocalDate):void
+get_FirstName():String
+get_LastName():String
+get_Gender():enGender
+get_DateOfBirth():LocalDate
+get_Nationality():String
+cast():
+cast(String,String,LocalDate,Person.enGender,String):
+getListofMovies():ArrayList<Movie>
+setListOfMovies(ArrayList<Movie>):void
+setSocialMediaLinks(HashMap<String,String):void
+addSocialMedia()String,String):void
+getSocialMediaLinks():HashMap<String,String>
+removeSocialMedia(String):void
+Director(String,String,LocalDate,Person.enGender,String):
+FindAssignedMovie(Movie):Movie
+IsMovieAssigned(Movie):boolean
+ static LoadDirectorFromFile(Movie):ArrayLlist<Director>
+ Static SaveDirectorsToFile(ArrayList<Director>):void
+static Find(String):Director
+static IsDirectorExist(String):Boolean
+ static AddNewDirector(Director): boolean
+ static DeleteDirector(Director): boolean
Extends
cast
#FirstName:String
#LasrtName:String
#Nationality:String
#DateofBirth:LocalDate
#Gender:enGender
+Male:enGender
+Female:enGender
- ListOfMovies:ArrayList<Movie>
-SocialMediaLinks:HashMap<String,String>
+person():
+getFullName():String
+set_LastName(String):void
+set_FirstName(String):void
+set_Gender(enGender):void
+set_Nationality(String):void
+set_DateOfBirth(LocalDate):void
+get_FirstName():String
+get_LastName():String
+get_Gender():enGender
+get_DateOfBirth():LocalDate
+get_Nationality():String
+cast():
+cast(String,String,LocalDate,Person.enGender,String):
+getListofMovies():ArrayList<Movie>
+setListOfMovies(ArrayList<Movie>):void
+setSocialMediaLinks(HashMap<String,String):void
+addSocialMedia()String,String):void
+getSocialMediaLinks():HashMap<String,String>
+removeSocialMedia(String):void
part of
Extends
Movie
- Id: int
- Title: String
- ReleaseDate: LocalDate
- RunningTime: int
- Genre: String
- Language: String
- Country: String
- Budget: float
- Revenue: float
- Rating: float
- Views: int
- PosterSrc: String
-Actors: ArrayList<Actor>
- Director: Director
- OldRatings: float
- noofwatched: int
- static Movies: ArrayList<Movie>
- static FILENAME: String = "movies.ser"
- static count: int = 1
+ Movie(String, String, int, float, String, String, LocalDate, float, int, Director, ArrayList<Actor>, String)
+ Movie() |+ Movie(String, LocalDate, int, String, genre: String, String, float)
+ Movie(int, tString, String, int, float, String, String, LocalDate, float, int)
+ Movie(String, String, int, float, String, String, LocalDate, String, Director, ArrayList<Actor>)
+ Movie(String, String, int, float, String,String, LocalDate, String, Director)
+ getId(): int
+ getTitle(): String
+ setTitle(String): void
+ getReleaseDate(): LocalDate
+ setReleaseDate(LocalDate): void
+ getRunningTime(): int
+ setRunningTime(runningTime: int): void
+ getGenre(): String
+ setGenre(String): void
+ getLanguage(): String
+ setLanguage(String): void
+ getCountry(): String
+ setCountry(String): void
+ getBudget(): float
+ setBudget(float): void
+ getRevenue(): float
+ setRevenue(float): void
+ getRating(): float
+ setRating(float): void
+ getViews(): int
+ setViews(int): void
+ getPosterSrc(): String
+ setPosterSrc(String): void
+ getActors(): ArrayList<Actor>
+ setActors(ArrayList<Actor>): void
+ addActor(Actor): void
+ getDirector(): Director
+ setDirector(Director): void
+ static getAllMovies(): ArrayList<Movie>
+ static LoadMovieFromFile(): ArrayList<Movie>
+ static SaveMoviesToFile(): void
+ static Find(String): Movie
+ static IsMovieExist(String): boolean
+ static AssignMovieToDirector(Movie, Director): void
+ static AssignMovieToActor(Movie, Actor): void
+ static AssignMovieToActors(Movie, ArrayList<Actor>): void
+ static FindByCast(String): ArrayList<Movie>
+ static AddNewMovie(Movie): boolean
+ static AddNewMovies(List<Movie>): void
+ static DeleteMovie(Movie): boolean
+ static getMostViewedMovie(int): ArrayList<Movie>
+ static getRecentMovies(): ArrayList<Movie>
+ static getTopRatedMovies(int): List<Movie>
+ static SameGenreMovies(String): ArrayList<Movie>
+ CalcRating(int): void
+ IncrementViewsByOne(): void
+ static Filter(String): ArrayList<Movie>
+ static CopyMovies(): ArrayList<Movie>
+ static UpdateMovie(Movie): boolean
+ static clearMoviesFile(): void
Part of
Actor
#FirstName:String
#LasrtName:String
#Nationality:String
#DateofBirth:LocalDate
#Gender:enGender
+Male:enGender
+Female:enGender
- ListOfMovies:ArrayList<Movie>
-SocialMediaLinks:HashMap<String,String>
+person():
+getFullName():String
+set_LastName(String):void
+set_FirstName(String):void
+set_Gender(enGender):void
+set_Nationality(String):void
+set_DateOfBirth(LocalDate):void
+get_FirstName():String
+get_LastName():String
+get_Gender():enGender
+get_DateOfBirth():LocalDate
+get_Nationality():String
+cast():
+cast(String,String,LocalDate,Person.enGender,String):
+getListofMovies():ArrayList<Movie>
+setListOfMovies(ArrayList<Movie>):void
+setSocialMediaLinks(HashMap<String,String):void
+addSocialMedia()String,String):void
+getSocialMediaLinks():HashMap<String,String>
+removeSocialMedia(String):void
+ Actor()
+ Actor(String, String,LocalDate, enGender, String)
+ FindAssignedMovie(Movie): Movie
+ IsMovieAssigned(Movie): boolean
+ LoadActorsFromFile(): ArrayList<Actor>
+ SaveActorsToFile(ArrayList<Actor>): void
+ Find(String): Actor
+ IsActorExist(String): boolean
+ AddNewActor(Actor): boolean
+ DeleteActor(Actor): boolean
User
- ID: int
- Username: String
- Password: String
- FirstName: String
- LastName: String
- Email: String
- WatchedMovies: ArrayList<Movie>
- WatchLaterMovies: ArrayList<Movie>
- userWatchRecord: ArrayList<UserWatchRecord>
- subscriptionHistory: ArrayList<Subscription>
- currentSubscription: Subscription - limitMovies: int
- static counter: int
- static Users: ArrayList<User>
- static FILENAME: String
+ User(username: String, email: String, lastName: String, firstName: String, password: String)
+ getAllUsers(): ArrayList<User> + LoadUsersFromFile(): ArrayList<User>
+ saveUsersDataToFile(): void
+ Find(Username: String): User
+ Find(Username: String, Password: String): User
+ Find(ID: int): User
+ isUserExist(Username: String): boolean
+ isUserExist(ID: int): boolean
+ isUserExist(Username: String, Password: String): boolean
+ AddNewUser(newUser: User): boolean
+ Delete(ID: int): boolean
+ Delete(Username: String): boolean
+ DecrementLimitByOne(): void
+ hasValidSups(): boolean
+ createNewSups(subscription: Subscription): void
+ WatchMovie(record: UserWatchRecord): void
+ AddToWatchLater(movie: Movie): boolean
+ isExistInWatchLater(movie: Movie): boolean
+ isExistInHistory(movie: Movie): boolean
+ clearFile(): void
Has a
Has a
subscription
- Plan: enPlan
- Price: float
- StartDate: LocalDate
- AllowedWatches: int
+ Subscription(enPlan, LocalDate)
+ getStartDate(): LocalDate
+ getAllowedWatches(): int
+ getPrice(): float
+ getPlan(): enPlan
+ setPlan(plan: enPlan): void
+ initializePlanDetails(): void
+ setPrice(): void
+ setAllowedWatches(): void
+ getPrices(plan: enPlan): float
User watch record
- userId: int
- movie: Movie
- watchDate: LocalDate
- rating: Integer
+ UserWatchRecord(int, Movie, LocalDate, Integer)
+ getUserId(): int
+ getMovie(): Movie
+ setMovie(Movie): void
+ getWatchDate(): LocalDate
+ setWatchDate(LocalDate): void
+ getRating(): Integer
+ setRating(Integer): void
+ saveWatchRecordsToFile(List<UserWatchRecord>): void
+ loadWatchRecordsFromFile(): List<UserWatchRecord>
+ getFavoriteGenres(int): Map<String, Integer>
+ getMostWatchedActors(int): Map<String, Integer>
+ findRecommendedMovies(Map<String, Integer>, mostWatchedActors: Map<String, Integer>): Set<Movie>
+ sortMoviesByReleaseDate(List<Movie>): void