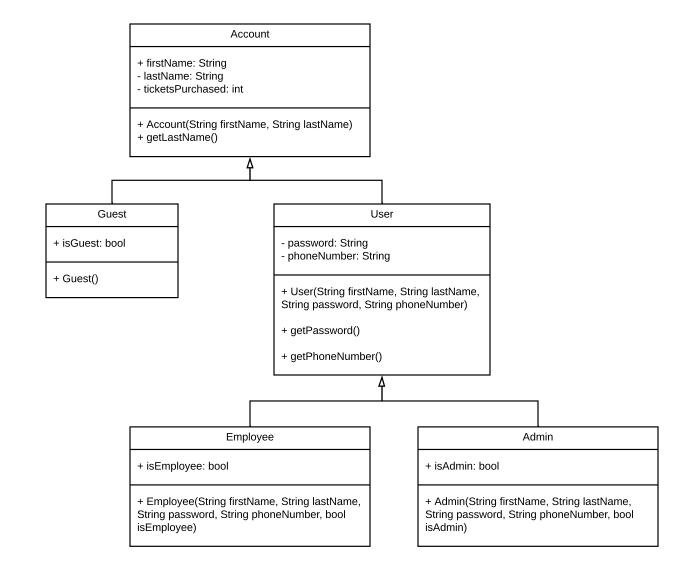
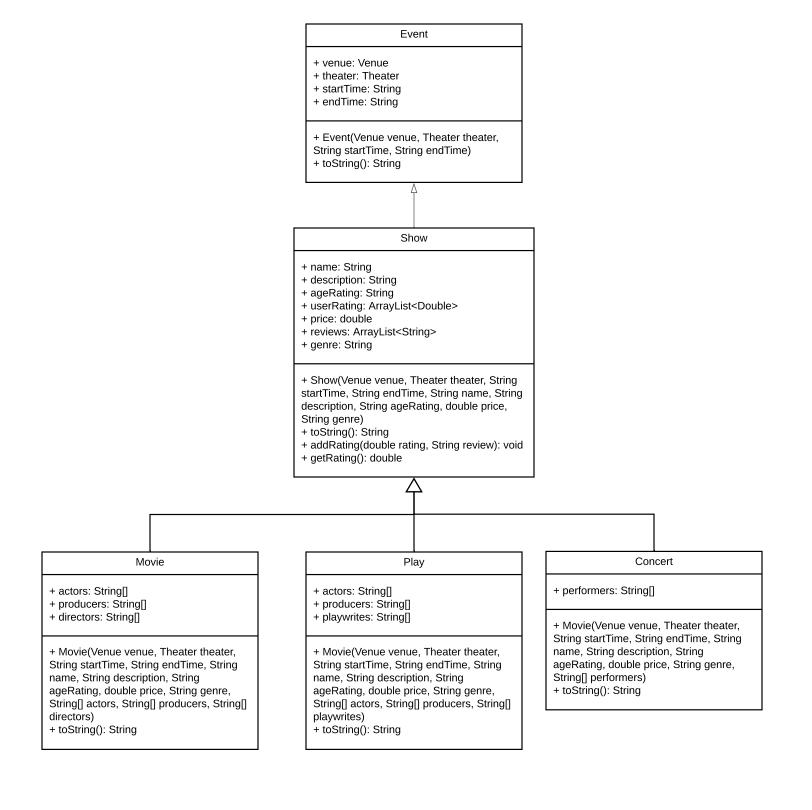
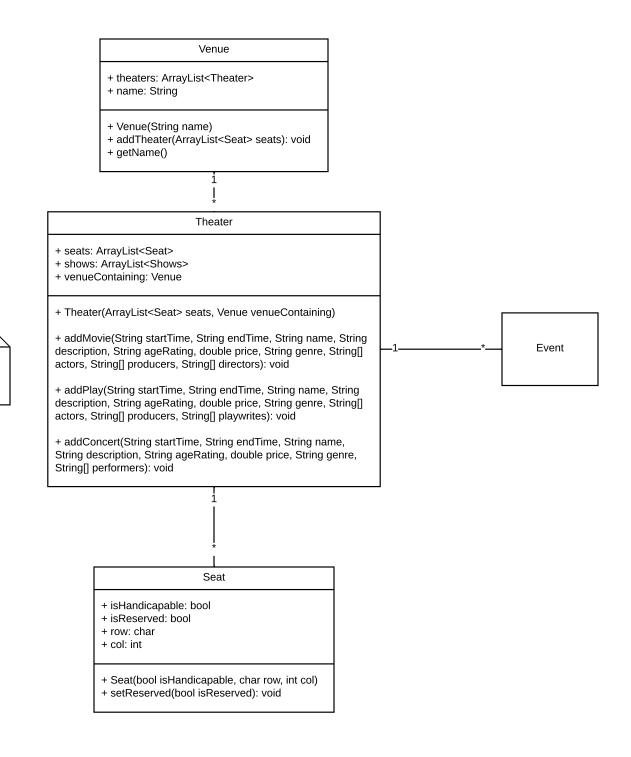


The guest will be an account with:
-firstName "Guest"
-lastName "User"
-isGuest True

There will only be one guest account so there is no need to allow the constructor to take in other names







When adding an Event, the method will use 'this.venueContaining' for the Venue and 'this' for the Theater

TheaterSystem

- accounts: ArrayList<Account>+ currentAccount: Account+ venues: ArrayList<Venue>+ userAccountType: String

- + TheaterSystem()
- + TheaterSystem(ArrayList<User> accounts, ArrayList<Venue> venues)
- + login(String firstName, String lastName, String password): String
- +updateAccountType(): void
- $+\ create Account (String\ first Name,\ String\ last Name,\ String\ phone Number,$

String password, bool isEmployee, bool isAdmin): String

- + searchShowName(String showName): ArrayList<Show>
- + searchGenre(String genre): ArrayList<Show>
- + searchRating(double rating): ArrayList<Show>
- + searchAgeRating(String ageRating): ArrayList<Show>
- + addShow(Theater theater): void
- + seatSearch(Theater theater, char row, int col): Seat
- + nextSeatSearch(Theater theater): Seat
- + purchaseTicket(Show show): String
- + purchaseTicket(Show show, char row, int col): String
- + refundTicket(Show show): String
- + printTicket(Show show, Seat seat): String
- + leaveReview(double rating, String review, Show show): void
- + deleteReview(double rating, String review, Show show): void
- + addEvent(Theater theater, Show show): void
- + displaySeats(Theater theater): String
- + saveToJSON(): void
- + loadFromJSON(): void

TheaterSystemUI

- + tsUI: TheaterSystemUI
- + ts: TheaterSystem
- + key: Scanner
- + quit: boolean

+ main(String[] args):void

- checkAccountType(String account): void
- login(): void
- createAccount(): void
- displayGuest(): void
- displayUser(): void
- displayEmployee(): void
- displayAdmin(): void
- searchByName(): void
- searchByGenre(): void
- searchByAgeRating(): void
- searchByUserRating(): void
- addEvent(): void
- removeEvent(): void
- displayEventList(): void
- displaySeats(Theater theater): void
- purchaseTicket(): void
- getRefund(): void
- leaveReview(): void
- removeReview(): void

JSONHandler

- + loadAccounts(): ArrayList<User>
- + loadVenues(): ArrayList<Venue>
- + saveAccounts(ArrayList<User>): void
- + saveVenues(ArrayList<Venue>): void
- + getVenueJSON(Venue venue): JSONObject
- + getTheaterJSON(Theater theater): JSONObject
- + getShowJSON(Show show): JSONObject
- + getSeatJSON(Seat seat): JSONObject
- + getAccountJSON(User user): JSONObject

JSONConstant is just an abstract class that holds constants used by the JSONHandler