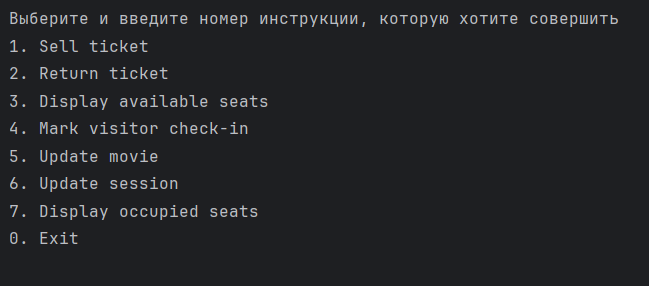
Данная программа моделирует работу кинотеатра

В классе Main мы создаём объект класса Cinema, в котором у нас хранятся данные о фильмах, сеансах и билетах.

Так же в Main мы создаём объект класс DoWork, который является наследником от интерфейса Work, данный класс и экземпляр будет помогать нам реализовывать работу кинотеатра. В данном классе мы создаём объекты классов CinemaFunctionDo и Saving. Первый реализует разные функции необходимые для работы (продать билет, вернуть билет и т.д.), а второй реализует функции сохранения данных в CSV формат.

В классе DoWork (файл Work.kt) запускается цикл, в ходе которого осуществляется взаимодействие с функциями и пользователем. Каждый раз человеку выводится инструкция, описывающая что нужно ввести, чтобы сделать конкретную операцию.



Для первой операции надо ввести дату и время сеанса (в определённом формате, указано в консоли), название фильма, ряд, место. Перенаправляет нас на функцию sellTicket.

2: надо ввести дату и время сеанса (в определённом формате, указано в консоли), название фильма, ряд, место. Перенаправляет нас на функцию returnTicket.

3: надо ввести дату и время сеанса (в определённом формате, указано в консоли), название фильма. Перенаправляет нас на функцию getAvailableSeats.

4: надо ввести дату и время сеанса (в определённом формате, указано в консоли), название фильма, ряд, место. Перенаправляет нас на функцию CheckTicket. Он меняет значение свойства checked с true на false.

5: Надо выбрать хотите ли вы добавить новый фильм или удалить какой то фильм из списка. Надо ввести название фильма. Перенаправляет нас на функцию addMovie или removeMovie. После выполнения каждой из этих функций происходит выполнение функций сохранения данных в CSV формат (saveToCSV)

6: Надо выбрать хотите ли вы добавить новый сеанс или удалить какой то сеанс из списка. Надо ввести название фильма. Перенаправляет нас на функцию addSession или removeSession. После выполнения каждой из этих функций происходит выполнение функций сохранения данных в CSV формат (saveToCSV).

7: надо ввести дату и время сеанса (в определённом формате, указано в консоли), название фильма. Перенаправляет нас на функцию getOccupiedSeats. Я выбрала данный формат хранения данных, потому что чаще всего он занимает меньше места в памяти, чем XML или JSON, а так же этот формат удобен для редактирования, удобно открыть этот файл в Exel.

0: после ввода 0 цикл и программа завершается

В каждом из этих случаев осуществляется проверка введённых данных. Если билет покупается или возвращается, то проверяется, что введенная дата должна быть больше текущего времени (чтобы мы не могли покупать или возвращать билеты на уже прошедшие сеансы)

Простите, что не успела сделать отдельный Validator, для проверки данных и из-за этого есть повторения в коде. Пожалейте бедного студента, пожалуйста.



Дополнительно так же реализовала функцию, которая позволит импортировать данные из СSV таблицы. Так же можно добавить дополнительные свойства в класс Movie, чтобы например хранить не только название, но и описание к фильмам, их длительность.

Надеюсь все было предельно понятно, если нет, задавайте вопросы. Приятной работы с программой!