Министерство образования и молодежной политики Свердловской области

ГАПОУ СО «Екатеринбургский колледж транспортного строительства»

Отчёт по программе «**Практическое занятие №8**»

Выполнила: Яковлева А.Н.

Группа: ПР-21

Преподаватель: Мирошниченко Г.В

2025

**Вариант 12**

**Задание:** Создать проект, в котором можно было бы вводить и получать информацию об определенных класса Кинотеатр.

**Входные и выходные данные основной программы**

n - количество внесенных фильмов, целое;

act - номер действия, целое;

move - номер фильма, целое.

**Описание класса Cinema:**

name - название фильма, строковый;

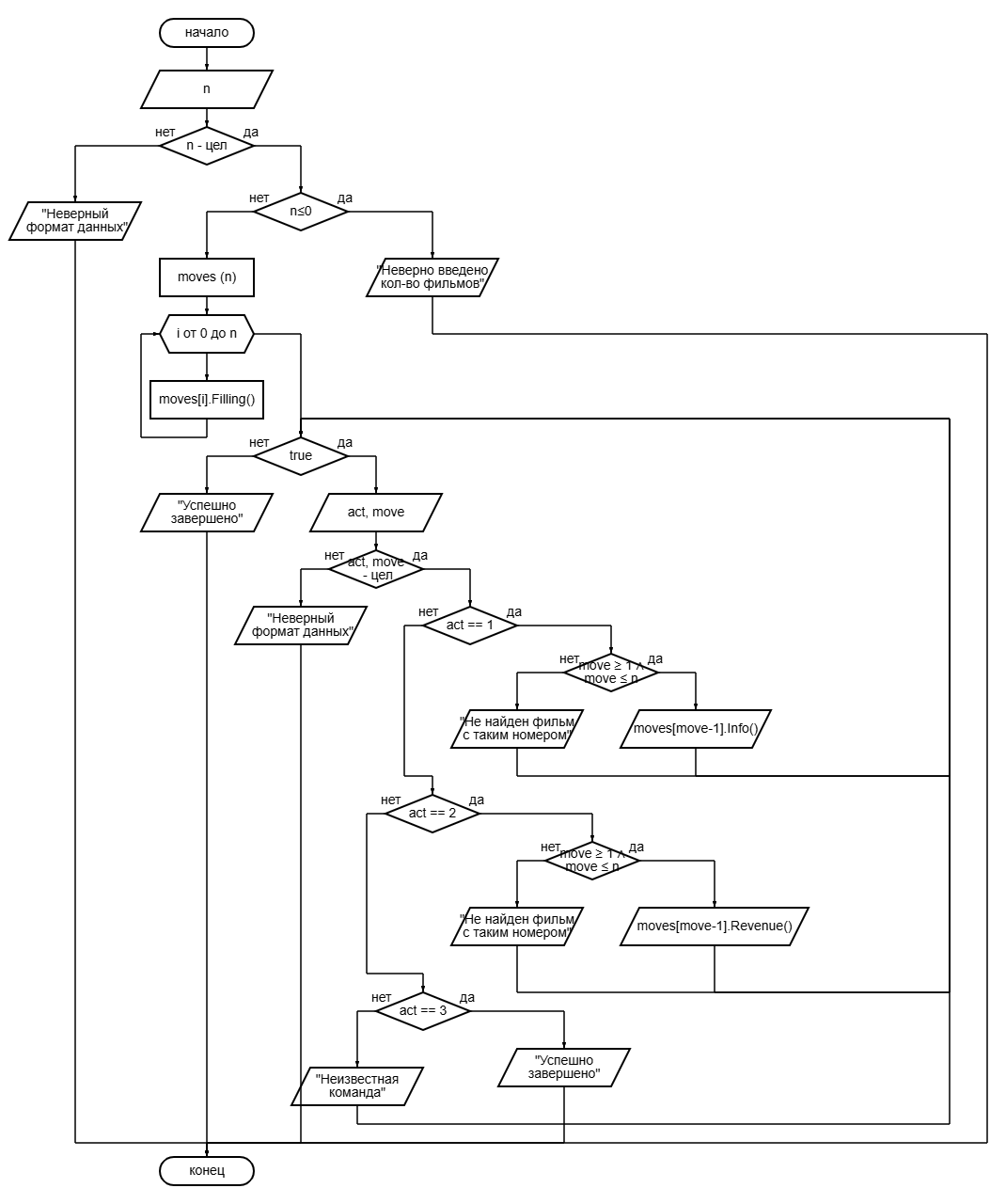
session - время начала сеанса, строковый;

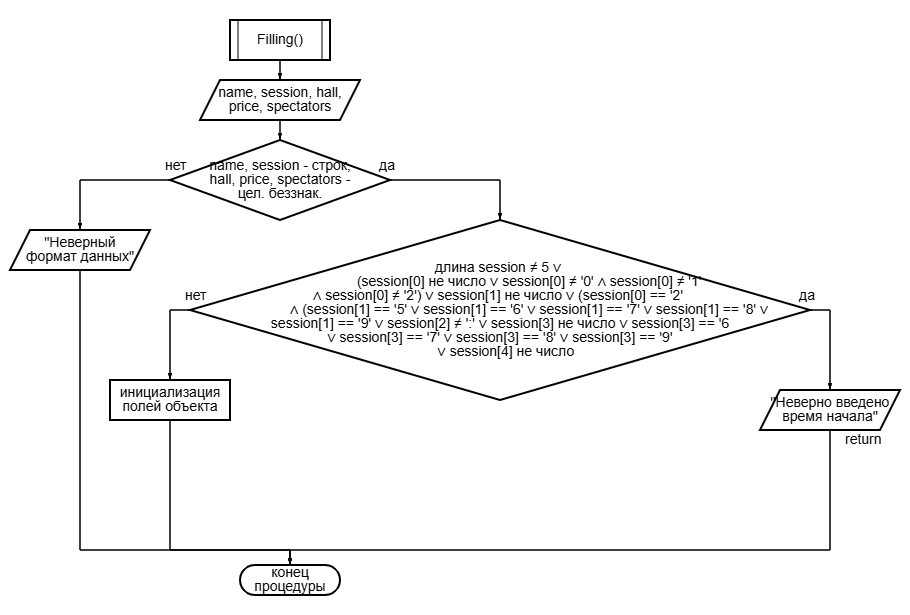
price - цена, целый беззнаковый;

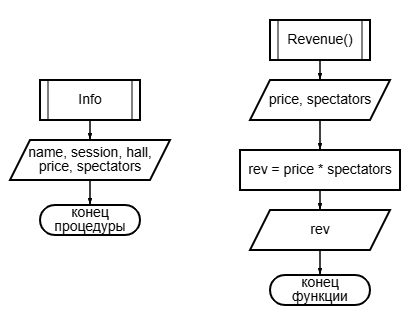
hall - номер зала, целый беззнаковый;

spectators - количество мест для зрителей, целый беззнаковый.

**Блок-схема**







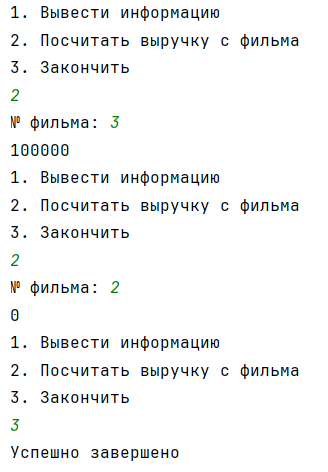
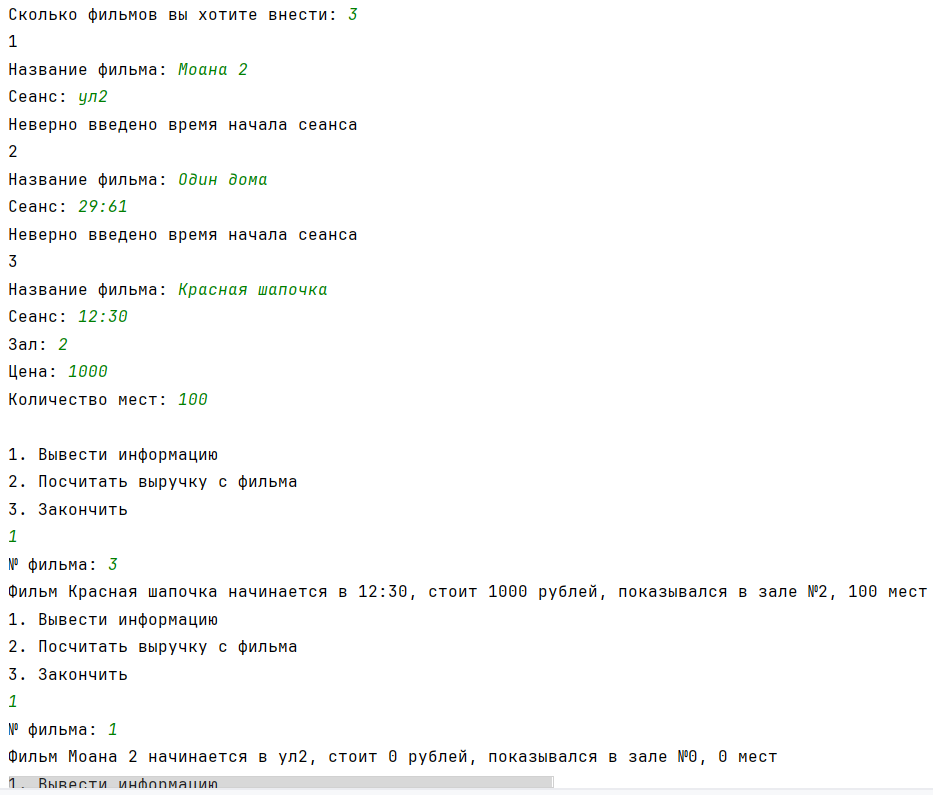
**Листинг программы**

class Cinema {  
 var name: String = "name"  
 var session: String = "00:00"  
 var price: UInt = 0u  
 var hall: UInt = 0u  
 var spectators: UInt = 0u  
 fun Filling(){  
 try {  
 *print*("Название фильма: ")  
 name = *readln*()  
 *print*("Сеанс: ")  
 session = *readln*()  
 if (session.length != 5 ||  
 (!session[0].*isDigit*() || session[0] != '0' && session[0] != '1' && session[0] != '2')  
 || !session[1].*isDigit*() || (session[0] == '2'  
 && (session[1] == '5' || session[1] == '6' || session[1] == '7' || session[1] == '8' || session[1] == '9'))  
 || session[2] != ':'  
 || !session[3].*isDigit*() || session[3] == '6' || session[3] == '7' || session[3] == '8' || session[3] == '9'  
 || !session[4].*isDigit*()  
 ) {  
 *println*("Неверно введено время начала сеанса")  
 return  
 }  
 *print*("Зал: ")  
 hall = *readln*().*toUInt*()  
 *print*("Цена: ")  
 price = *readln*().*toUInt*()  
 *print*("Количество мест: ")  
 spectators = *readln*().*toUInt*()  
 *println*()  
 }  
 catch (e: Exception) {  
 *println*("Неверный формат данных")  
 }  
 }  
 fun Info(): String{  
 return "Фильм ${name} начинается в ${session}, стоит ${price} рублей, показывался в зале №${hall}, ${spectators} мест"  
 }  
 fun Revenue() :UInt{  
 val rev = price \* spectators  
 return rev  
 }  
}

fun main() {  
 try {  
 *print*("Сколько фильмов вы хотите внести: ")  
 val n = *readln*().*toInt*()  
 when  
 {  
 n <= 0 -> *println*("Неверно введено количество фильмов")  
 else -> {  
 val moves = Array(n) **{** Cinema() **}**  
for (i in 0 *until* n)  
 {  
 *println*(i+1)  
 moves[i].Filling()  
 }  
 while (true) {  
 *println*("1. Вывести информацию")  
 *println*("2. Посчитать выручку с фильма")  
 *println*("3. Закончить")  
 val act = *readln*().*toInt*()  
 when (act){  
 1 -> {  
 *print*("№ фильма: ")  
 val move = *readln*().*toInt*()  
 when{  
 (move in 1..n) -> *println*(moves[move-1].Info())  
 else -> *println*("Не найдем фильм с таким номером")  
 }  
 }  
 2 -> {  
 *print*("№ фильма: ")  
 val move = *readln*().*toInt*()  
 when{  
 (move in 1..n) -> *println*(moves[move-1].Revenue())  
 else -> *println*("Не найдем фильм с таким номером")  
 }  
 }  
 3 -> break  
 else -> *println*("Неизвестная команда")  
 }  
 }  
 *println*("Успешно завершено")  
 }  
 }  
 }  
 catch (e: Exception)  
 {  
 *println*("Неверный формат данных")  
 }  
}

**Тестовые ситуации**





**Вывод:** Я научилась работать с классами в Kotlin.