## Министерство образования и молодежной политики Свердловской области



ГАПОУ СО «Екатеринбургский колледж транспортного строительства»

Отчёт по программе «**Класс enum в Kotlin**»

Выполнил: Сергеева Агата Андреевна

Группа: ПР-21

Преподаватель: Мирошниченко Г.В

2024 г.

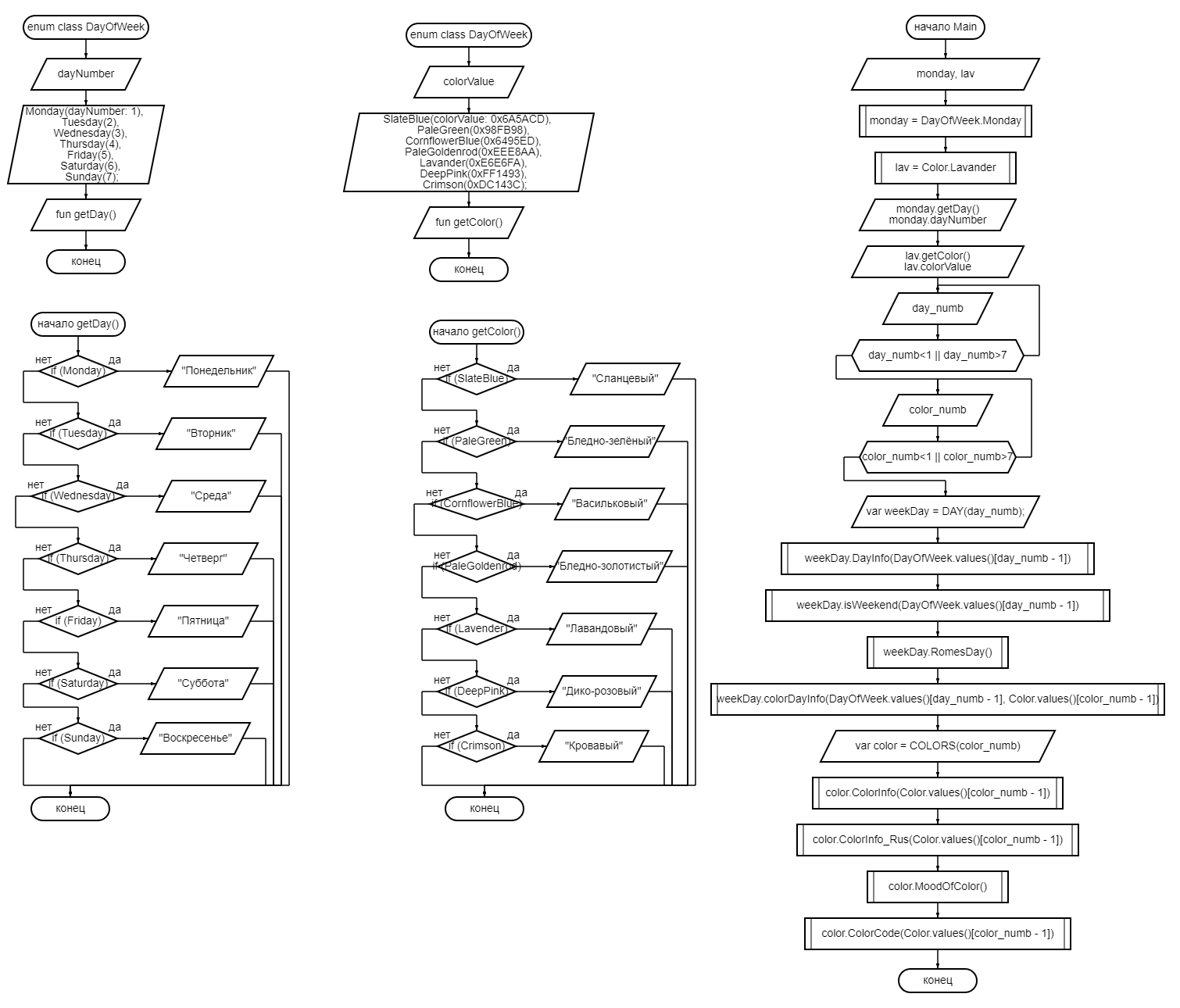
**Задание:**

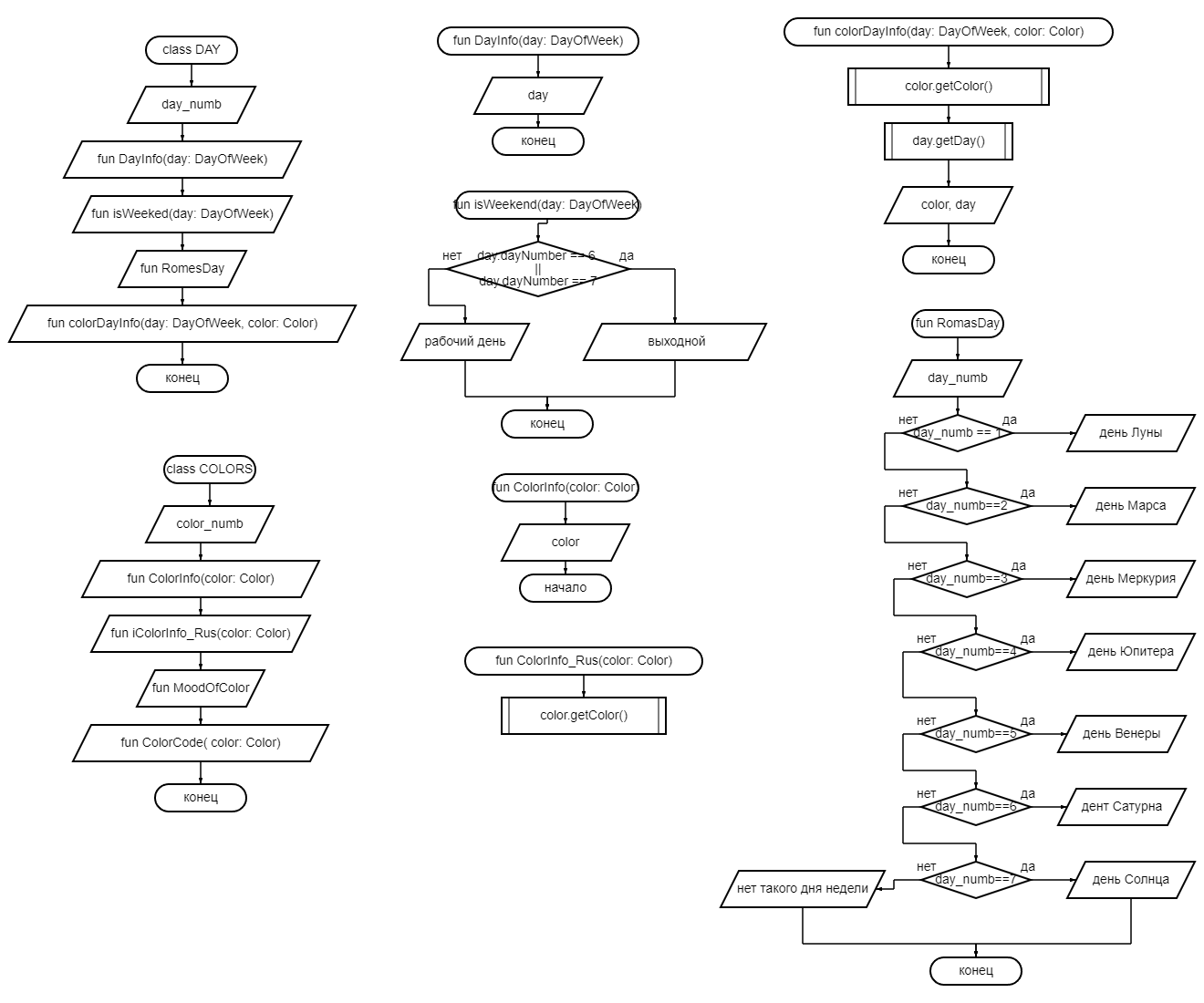
С помощью класса enum задайте дни недели и цвета и используя вспомогательные функции, выдайте значения на экран в виде строки и числа. Создайте 2 класса и в каждом из них по 4 метода

**Входные данные:** day\_numb – номер дня недели, int; color\_numb –номер цвета, int.

**Выходные данные:** различные сообщения, в зависимости от введённого номера дня и номера цвета, string

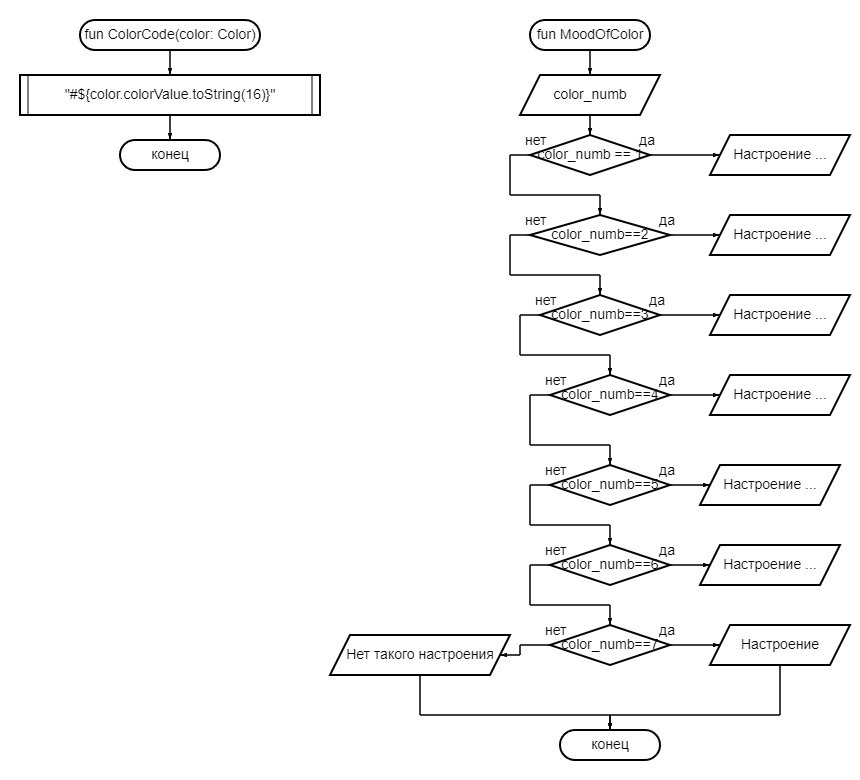
**Блок-схема:**





конец

конец



**Листинг программы:**

Enum class DayOfWeek:

enum class DayOfWeek(val dayNumber: Int) {  
 *Monday*(1),  
 *Tuesday*(2),  
 *Wednesday*(3),  
 *Thursday*(4),  
 *Friday*(5),  
 *Saturday*(6),  
 *Sunday*(7);  
  
 fun getDay(): String {  
 return when (this) {  
 *Monday* -> "Понедельник"  
 *Tuesday* -> "Вторник"  
 *Wednesday* -> "Среда"  
 *Thursday* -> "Четверг"  
 *Friday* -> "Пятница"  
 *Saturday* -> "Суббота"  
 *Sunday* -> "Вокресенье"  
 }  
 }  
}

Enum class Color:

enum class Color (val colorValue: Int) {  
 *SlateBlue*(0x6A5ACD),  
 *PaleGreen*(0x98FB98),  
 *CornflowerBlue*(0x6495ED),  
 *PaleGoldenrod*(0xEEE8AA),  
 *Lavander*(0xE6E6FA),  
 *DeepPink*(0xFF1493),  
 *Crimson*(0xDC143C);  
  
 fun getColor(): String {  
 return when (this) {  
 *SlateBlue* -> "Сланцевый"  
 *PaleGreen* -> "Бледно-зелёный"  
 *CornflowerBlue* -> "Васильковый"  
 *PaleGoldenrod* -> "Бледно-золотистый"  
 *Lavander* -> "Лавандовый"  
 *DeepPink* -> "Дико-розовый"  
 *Crimson* -> "Кровавый"  
 }  
 }  
}

class DAY:

class DAY (var day\_numb: Int) {  
 fun DayInfo(day: DayOfWeek) {  
 *println*("День недели (ин инглиш): $day")  
 }  
 fun isWeekend(day: DayOfWeek) {  
 if (day.dayNumber == 6 || day.dayNumber == 7) {  
 *println*("${day.getDay()} - это выходной")  
 }

else {  
 *println*("${day.getDay()} - это рабочий день недели")  
 }  
 }  
 fun RomesDay(): String {  
 when(day\_numb)  
 {  
 1 -> return "день Луны"  
 2 -> return "день Марса"  
 3 -> return "день Меркурия"  
 4 -> return "день Юпитера"  
 5 -> return "день Венеры"  
 6 -> return "день Сатурна"  
 7 -> return "день Солнца"  
 else -> return "Нет такого дня недели ни у нас, ни у римлян"  
 }  
 }  
 fun colorDayInfo(day: DayOfWeek, color: Color) {  
 *println*("А если бы это был монстр с моей игре, то его бы звали: ${color.getColor()} ${day.getDay()}. Красиво, не правда ли?")  
 }  
}

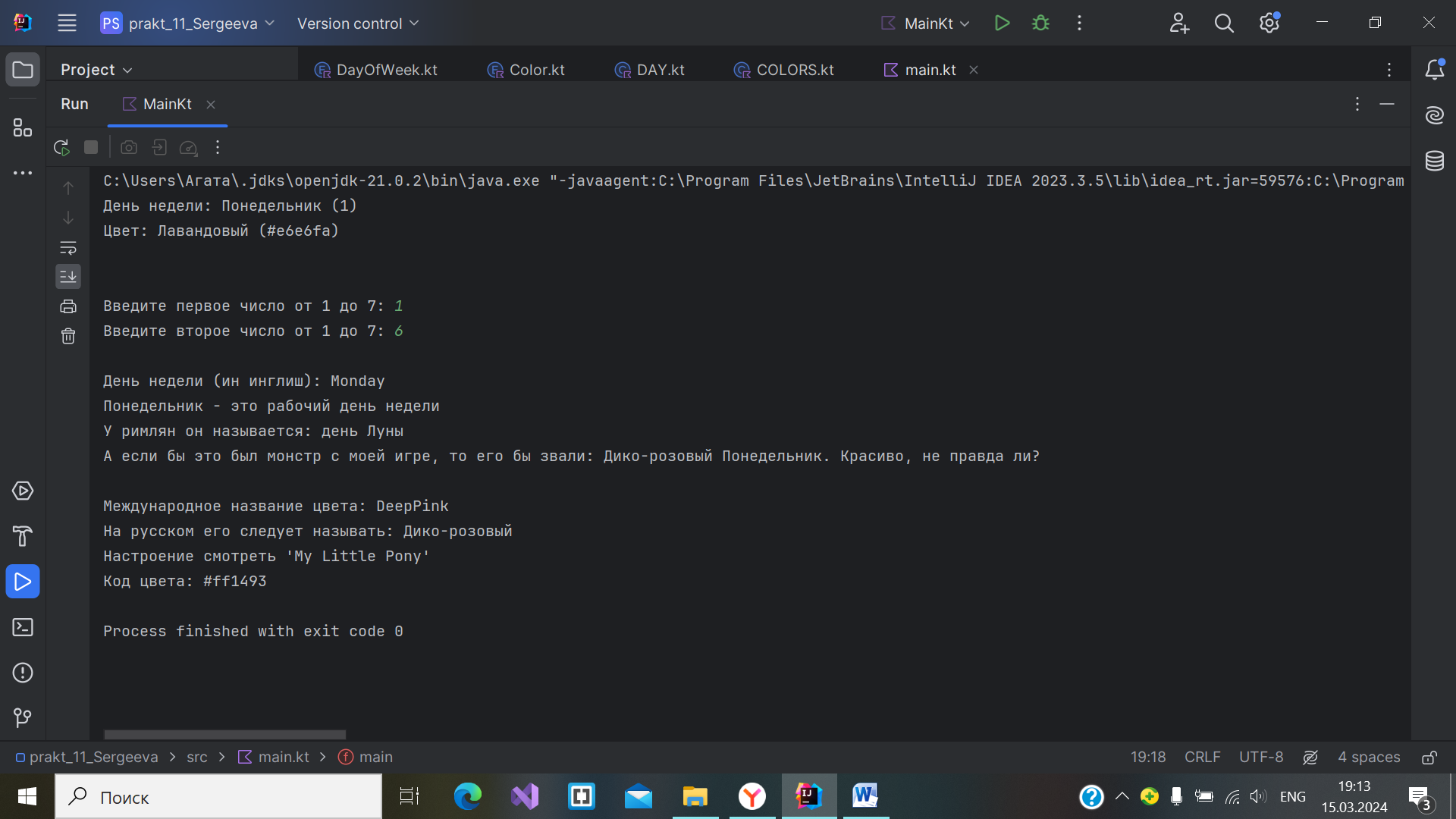
class COLORS:

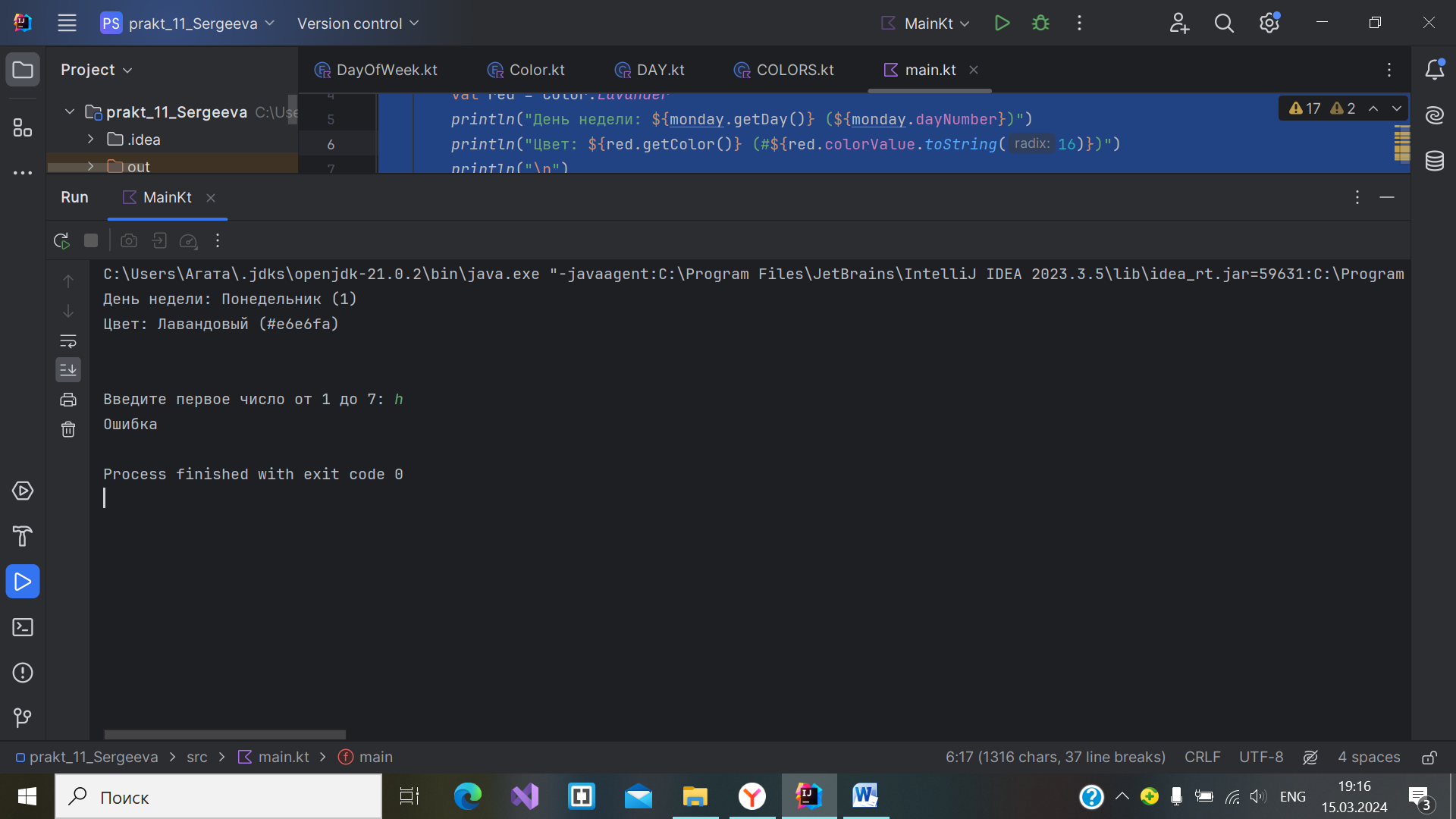
class COLORS (var color\_numb: Int){  
 fun ColorInfo(color: Color) {  
 *println*("Международное название цвета: $color")  
 }  
 fun ColorInfo\_Rus(color: Color) {  
 *println*("На русском его следует называть: ${color.getColor()}")  
 }  
 fun MoodOfColor():String{  
 when(color\_numb)  
 {  
 1 -> return "Настроение танцевать в поле цветов"  
 2 -> return "Настроение готовить чай"  
 3 -> return "Настроение перечитать 'Мастер и Маргарита'"  
 4 -> return "Настроение обновить монстров в игре по птпм"  
 5 -> return "Настроение спать"  
 6 -> return "Настроение смотреть 'My Little Pony'"  
 7 -> return "Настроение ехидно смеяться"  
 else -> return "Нет такого настроения"  
 }  
 }  
 fun ColorCode(color: Color) {  
 *println*("Код цвета: #${color.colorValue.*toString*(16)}")  
 }  
}

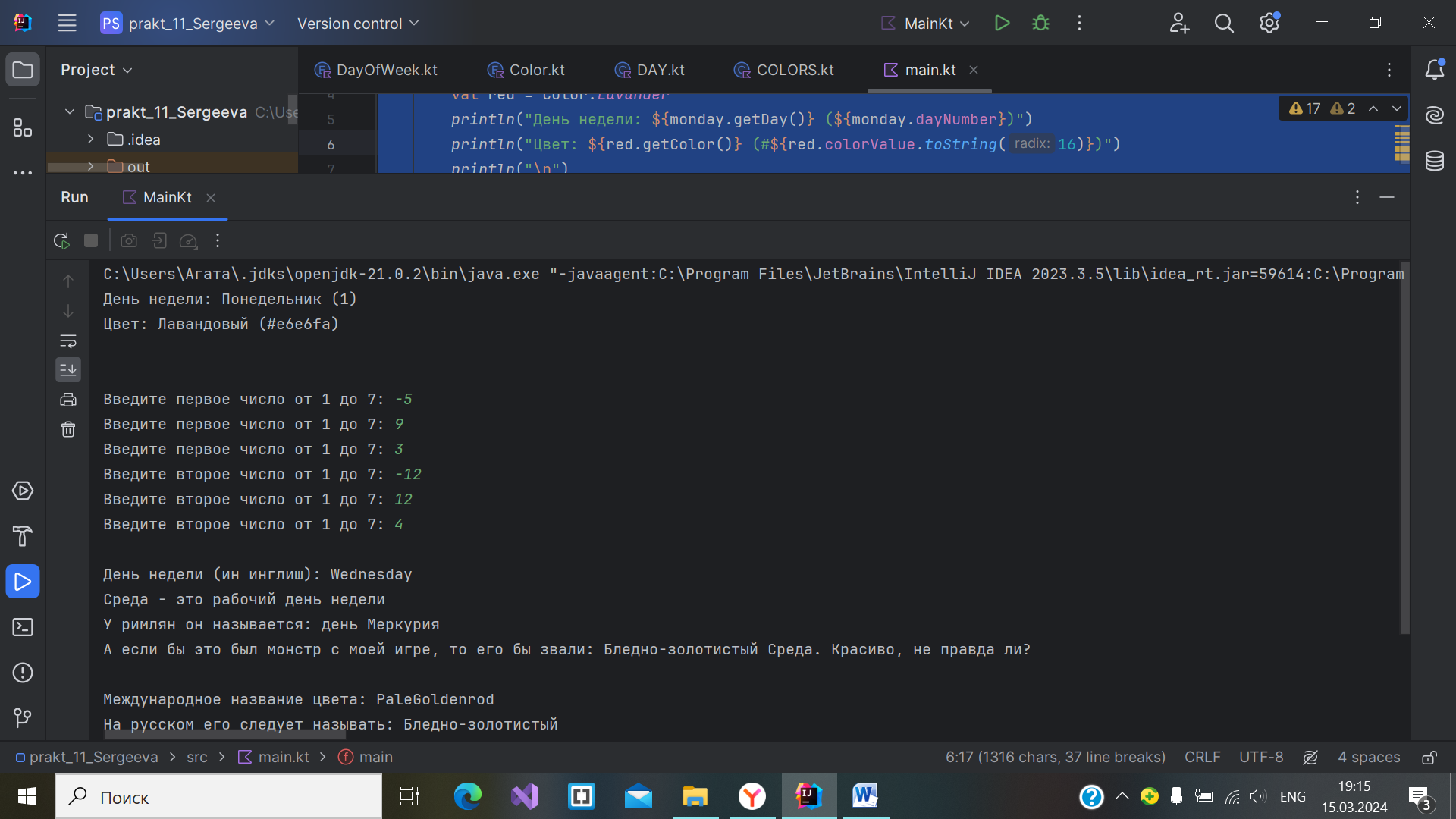
Main:

fun main() {  
 try {  
 var monday = DayOfWeek.*Monday* val red = Color.*Lavander  
 println*("День недели: ${monday.getDay()} (${monday.dayNumber})")  
 *println*("Цвет: ${red.getColor()} (#${red.colorValue.*toString*(16)})")  
 *println*("\n")  
  
 var day\_numb = 0;  
 var color\_numb = 0;  
 do {  
 *print*("Введите первое число от 1 до 7: ")  
 day\_numb = *readln*()!!.*toInt*()  
 } while (day\_numb < 1 || day\_numb > 7)  
 do {  
 *print*("Введите второе число от 1 до 7: ")  
 color\_numb = *readln*()!!.*toInt*()  
 } while (color\_numb < 1 || color\_numb > 7)  
 *println*("")  
  
 var weekDay = DAY(day\_numb);  
 weekDay.DayInfo(DayOfWeek.values()[day\_numb - 1])  
 weekDay.isWeekend(DayOfWeek.values()[day\_numb - 1])  
 *println*("У римлян он называется: ${weekDay.RomesDay()}")  
 weekDay.colorDayInfo(DayOfWeek.values()[day\_numb - 1], Color.values()[color\_numb - 1])  
 *println*("")  
  
 var color = COLORS(color\_numb)  
 color.ColorInfo(Color.values()[color\_numb - 1])  
 color.ColorInfo\_Rus(Color.values()[color\_numb - 1])  
 *println*(color.MoodOfColor())  
 color.ColorCode(Color.values()[color\_numb - 1])  
 }  
 catch (e : Exception)  
 {  
 *println* ("Ошибка")  
 }  
}

**Тестовые ситуации:**







**Вывод:** изучили класс enum в Kotlin