

№1. Определить день недели по номеру.

```
fun getDayOfWeek(day: Int): String {  
    return when (day) {  
        1 -> "Понедельник"  
        2 -> "Вторник"  
        3 -> "Среда"  
        4 -> "Четверг"  
        5 -> "Пятница"  
        6 -> "Суббота"  
        7 -> "Воскресенье"  
        else -> "Некорректный номер дня"  
    }  
}  
  
fun main() {  
    println(getDayOfWeek(3)) // Вывод: Среда  
}
```

№2 Определить тип треугольника по длинам сторон\

```
fun triangleType(a: Double, b: Double, c: Double): String {  
    return when {  
        a <= 0 || b <= 0 || c <= 0 -> "Некорректные длины сторон"  
        a == b && b == c -> "Равносторонний треугольник"  
        a == b || b == c || a == c -> "Равнобедренный треугольник"  
        else -> "Разносторонний треугольник"  
    }  
}  
  
fun main() {  
    println(triangleType(3.0, 3.0, 3.0)) // Вывод: Равносторонний треугольник  
}
```

№3

Вывод оценок по числовым значениям

```

fun grade(score: Int): String {
    return when (score) {
        in 90..100 -> "A"
        in 80..89 -> "B"
        in 70..79 -> "C"
        in 60..69 -> "D"
        in 0..59 -> "F"
        else -> "Некорректная оценка"
    }
}

```

```

fun main() {
    println(grade(85)) // Вывод: B
}

```

№4 Определение времени суток

```

fun timeOfDay(hour: Int): String {
    return when (hour) {
        in 0..5 -> "Ночь"
        in 6..11 -> "Утро"
        in 12..17 -> "День"
        in 18..23 -> "Вечер"
        else -> "Некорректное время"
    }
}

```

```

fun main() {
    println(timeOfDay(14)) // Вывод: День
}

```

№5 Определить знак числа

```

fun signOfNumber(number: Int): String {
    return when {
        number > 0 -> "Положительное число"
    }
}

```

```

        number < 0 -> "Отрицательное число"
        else -> "Ноль"
    }
}

```

```

fun main() {
    println(signOfNumber(-5)) // Вывод: Отрицательное число
}

```

№6 Угадай число

```

import kotlin.random.Random

```

```

fun guessNumber(userGuess: Int, secretNumber: Int): String {
    return when {
        userGuess < secretNumber -> "Слишком низко!"
        userGuess > secretNumber -> "Слишком высоко!"
        else -> "Поздравляем! Вы угадали число!"
    }
}

```

```

fun main() {
    val secretNumber = Random.nextInt(1, 101) // Секретное число от 1 до 100
    println(guessNumber(50, secretNumber)) // Пример вызова функции
}

```

№7 Определение длины строки

```

fun stringLength(str: String): Int {
    return str.length
}

```

```

fun main() {
    println(stringLength("Hello, World!")) // Вывод: 13
}

```

№8 Определить время приготовления по типу пищи

```

fun cookingTime(foodType: String): String {
    return when (foodType.toLowerCase()) {
        "паста" -> "15 минут"
        "мясо" -> "30 минут"
        "овощи" -> "10 минут"
        else -> "Неизвестный тип пищи"
    }
}

fun main() {
    println(cookingTime("мясо")) // Вывод: 30 минут
}

```

№9 (тоже самое что и задание 7)

№10 Способы оплаты

```

fun paymentMethod(method: String): String {
    return when (method.toLowerCase()) {
        "наличные" -> "Вы выбрали оплату наличными."
        "кредитная карта" -> "Вы выбрали оплату кредитной картой."
        "paypal" -> "Вы выбрали оплату через PayPal."
        else -> "Некорректный способ оплаты."
    }
}

fun main() {
    println(paymentMethod("кредитная карта")) // Вывод: Вы выбрали оплату кредитной картой.
}

```

№11 Группа крови и возможные доноры

```

fun bloodTypeCompatibility(bloodType: String): String {
    return when (bloodType.toUpperCase()) {
        "A" -> "Можно переливать: A, AB"
    }
}

```

```

        "B" -> "Можно переливать: B, AB"
        "AB" -> "Можно переливать: AB"
        "O" -> "Можно переливать: A, B, AB, O"
        else -> "Некорректная группа крови"
    }
}

```

```

fun main() {
    println(bloodTypeCompatibility("A")) // Вывод: Можно переливать: A, AB
}

```

№12 Национальности по странам

```

fun nationalityByCountry(country: String): String {
    return when (country.toUpperCase()) {
        "США" -> "Американец/Американка"
        "РОССИЯ" -> "Русский/Русская"
        "ЯПОНИЯ" -> "Японец/Японка"
        "ГЕРМАНИЯ" -> "Немец/Немка"
        else -> "Неизвестная страна"
    }
}

```

```

fun main() {
    println(nationalityByCountry("Россия")) // Вывод: Русский/Русская
}

```

№13 Коды ошибок

```

fun errorCodeMessage(code: Int): String {
    return when (code) {
        100 -> "Ошибка сети"
        200 -> "Ошибка сервера"
        300 -> "Ошибка доступа"
        else -> "Неизвестный код ошибки"
    }
}

```

```
}
```

```
fun main() {
```

```
    println(errorMessage(200)) // Вывод: Ошибка сервера
```

```
}
```