

Зад. 1 Определить день недели по номеру

```
1.kt x
1 fun getDayOfWeek(dayNumber: Int): String {
2     return when (dayNumber) {
3         1 -> "Понедельник"
4         2 -> "Вторник"
5         3 -> "Среда"
6         4 -> "Четверг"
7         5 -> "Пятница"
8         6 -> "Суббота"
9         7 -> "Воскресенье"
10        else -> "Некорректный номер дня"
11    }
12 }
13
14 fun main() {
15     val dayNumber = 3 // Укажите номер дня
16     val dayOfWeek = getDayOfWeek(dayNumber)
17     println(dayOfWeek)
18 }
```

```
C:\Users\Student\.jdk\openjdk-22.0.2\bin\jav
Среда

Process finished with exit code 0
```

Зад. 2 Определить тип треугольника по длинам сторон

```
2.kt x
1 fun determineTriangleType(a: Double, b: Double, c: Double): String {
2     return when {
3         a <= 0 || b <= 0 || c <= 0 -> "Стороны должны быть положительными"
4         a + b <= c || a + c <= b || b + c <= a -> "Это не треугольник"
5         a == b && b == c -> "Равносторонний треугольник"
6         a == b || b == c || a == c -> "Равнобедренный треугольник"
7         else -> "Разносторонний треугольник"
8     }
9 }
10
11 fun main() {
12     val sideA = 3.0 // Укажите длину стороны A
13     val sideB = 4.0 // Укажите длину стороны B
14     val sideC = 5.0 // Укажите длину стороны C
15
16     val triangleType = determineTriangleType(sideA, sideB, sideC)
17     println(triangleType)
18 }
```

```
C:\Users\Student\.jdk\openjdk-22.0.2
Разносторонний треугольник
```

```
Process finished with exit code 0
```

Зад. 3 Вывод оценок по числовым значениям

```
3.kt x
1 fun getGrade(score: Int): String {
2     return when (score) {
3         in 90 ≤ .. ≤ 100 -> "Отлично"
4         in 75 ≤ .. ≤ 89 -> "Хорошо"
5         in 60 ≤ .. ≤ 74 -> "Удовлетворительно"
6         in 0 ≤ .. ≤ 59 -> "Неудовлетворительно"
7         else -> "Некорректный балл"
8     }
9 }
10
11 fun main() {
12     val score = 85 // Укажите ваше числовое значение
13
14     val grade = getGrade(score)
15     println(grade)
16 }
```

```
C:\Users\Student\.jdk\openjdk-22.0.  
Хорошо  
  
Process finished with exit code 0
```

Зад. 4 Определение времени суток

```
4.kt ×  
1 fun determineTimeOfDay(hour: Int): String {  
2     return when (hour) {  
3         in 0 ≤ .. ≤ 5 -> "Ночь"  
4         in 6 ≤ .. ≤ 11 -> "Утро"  
5         in 12 ≤ .. ≤ 17 -> "День"  
6         in 18 ≤ .. ≤ 23 -> "Вечер"  
7         else -> "Некорректное время"  
8     }  
9 }  
10  
11 fun main() {  
12     val hour = 14 // Укажите час в 24-часовом формате  
13  
14     val timeOfDay = determineTimeOfDay(hour)  
15     println(timeOfDay)  
16 }
```

```
C:\Users\Student\.jdk\openjdk-22.0.2  
День  
  
Process finished with exit code 0
```

Зад. 5 Определить знак числа

```
5.kt x
1  fun determineSign(number: Int): String {
2      return when {
3          number > 0 -> "Положительное"
4          number < 0 -> "Отрицательное"
5          else -> "Ноль"
6      }
7  }
8
9  fun main() {
10     val number = -5 // Укажите число для проверки
11
12     val sign = determineSign(number)
13     println("Знак числа $number: $sign")
14 }
```

```
C:\Users\Student\.jdk\openjdk-22.0.1
Знак числа -5: Отрицательное

Process finished with exit code 0
```

Зад. 6 Угадай число

```
6.kt x
1  import kotlin.random.Random
2
3  fun main() {
4      val numberToGuess = Random.nextInt( from: 1, until: 101)
5      var userGuess: Int?
6      var attempts = 0
7
8      println("Я загадал число от 1 до 100. Попробуй его угадать!")
9
10     do {
11         println("Введите ваше предположение:")
12         userGuess = readLine()?.toIntOrNull()
13
14         if (userGuess == null) {
15             println("Пожалуйста, введите действительное число.")
16             continue
17         }
18
19         attempts++
20
21         when {
22             userGuess < numberToGuess -> println("Мое число больше.")
23             userGuess > numberToGuess -> println("Мое число меньше.")
24             else -> println("Поздравляю! Вы угадали число $numberToGuess за $attempts попыток.")
25         }
26     } while (userGuess != numberToGuess)
27 }
```

```
C:\Users\Student\.jdk\openjdk-22.0.2\bin\java.exe "  
Я загадал число от 1 до 100. Попробуй его угадать!  
Введите ваше предположение:  
2  
Поздравляю! Вы угадали число 2 за 1 попыток.  
  
Process finished with exit code 0
```

Зад. 7 Определение длины строки

```
7.kt x  
1 fun main() {  
2     val myString = "Привет, мир!"  
3     val length = myString.length  
4  
5     println("Длина строки: $length")  
6 }
```

```
C:\Users\Student\.jdk\openjdk-22.0.2\bin\java.exe "  
Длина строки: 12  
  
Process finished with exit code 0
```

Зад. 8 Определить время приготовления по типу пищи

```
8.kt x  
1 fun main() {  
2     val cookingTimes = mapOf(  
3         "паста" to 10,  
4         "мясо" to 30,  
5         "овощи" to 15,  
6         "рыба" to 20,  
7         "рис" to 25  
8     )  
9  
10    val foodType = "мясо"  
11    val time = cookingTimes[foodType]  
12  
13    if (time != null) {  
14        println("Время приготовления $foodType: $time минут")  
15    } else {  
16        println("Информация о времени приготовления для $foodType не найдена")  
17    }  
18 }
```

```
C:\Users\Student\.jdk\openjdk-22.0.2
Время приготовления мяса: 30 минут

Process finished with exit code 0
```

Зад. 9 Определение длины строки

```
9.kt x
1 fun main() {
2     val str = "Привет, мир!"
3     val length = str.length
4     println("Длина строки: $length")
5 }
```

```
C:\Users\Student\.jdk\openjdk-22.0.2
Длина строки: 12

Process finished with exit code 0
```

Зад. 10 Способы оплаты: Напишите программу, которая принимает способ оплаты (наличные, кредитная карта, PayPal) и выдает соответствующее сообщение.

```
10.kt x
1 fun main() {
2     println("Введите способ оплаты (наличные, кредитная карта, PayPal): ")
3     val paymentMethod = readLine()
4
5     when (paymentMethod?.toLowerCase()) {
6         "наличные" -> println("Вы выбрали оплату наличными.")
7         "кредитная карта" -> println("Вы выбрали оплату кредитной картой.")
8         "paypal" -> println("Вы выбрали оплату через PayPal.")
9         else -> println("Неизвестный способ оплаты. Пожалуйста, выберите наличные, кредитную карту или PayPal.")
10    }
11 }
```

```
C:\Users\Student\.jdk\openjdk-22.0.2\bin\java.exe "-javaagent
Введите способ оплаты (наличные, кредитная карта, PayPal):
Наличные
Вы выбрали оплату наличными.

Process finished with exit code 0
```

Зад. 11 Группа крови: вводится группа крови (А, В, АВ, О) и выводится, какие типы крови можно применять для переливания.

```
11.kt x
1 fun main() {
2     println("Введите вашу группу крови (A, B, AB, 0): ")
3     val bloodType = readLine()?.toUpperCase()
4
5     val compatibleBloodTypes = when (bloodType) {
6         "A" -> "A, 0"
7         "B" -> "B, 0"
8         "AB" -> "A, B, AB, 0"
9         "0" -> "0"
10        else -> "Некорректная группа крови"
11    }
12
13    println("Совместимые группы крови для переливания: $compatibleBloodTypes")
14 }
```

```
C:\Users\Student\.jdk\openjdk-22.0.2\bin\java.exe
Введите вашу группу крови (A, B, AB, 0):
A
Совместимые группы крови для переливания: A, 0
Process finished with exit code 0
```

Зад. 12 Национальности: Напишите программу, которая по странам (США, Россия, Япония и т.д.) выводит информацию о соответствующей национальности.

```
12.kt x
1 fun main() {
2     println("Введите название страны (США, Россия, Япония и т.д.): ")
3     val country = readLine()?.trim()
4
5     val nationality = when (country) {
6         "США" -> "американец/американка"
7         "Россия" -> "русский/русская"
8         "Япония" -> "японец/японка"
9         "Германия" -> "немец/немка"
10        "Франция" -> "француз/француженка"
11        "Италия" -> "итальянец/итальянка"
12        "Китай" -> "китаец/китаянка"
13        else -> "Национальность не найдена"
14    }
15
16    println("Национальность для страны $country: $nationality")
17 }
```

```
C:\Users\Student\.jdk\openjdk-22.0.2\bin\java.exe "-j
Введите название страны (США, Россия, Япония и т.д.):
Россия
Национальность для страны Россия: русский/русская

Process finished with exit code 0
```

Зад. 13 Коды ошибок: Программа принимает код ошибки (100, 200, 300) и выводит сообщение об ошибке (например, "Ошибка сети", "Ошибка сервера" и т.д.).

```
13.kt x
1 fun main() {
2     println("Введите код ошибки (100, 200, 300): ")
3     val errorCode = readLine()?.trim()?.toIntOrNull()
4
5     val errorMessage = when (errorCode) {
6         100 -> "Ошибка сети"
7         200 -> "Ошибка сервера"
8         300 -> "Ошибка клиента"
9         else -> "Неизвестный код ошибки"
10    }
11
12    println("Сообщение об ошибке: $errorMessage")
13 }
```

```
C:\Users\Student\.jdk\openjdk-22.0.2\bin\java.exe "-j
Введите код ошибки (100, 200, 300):
300
Сообщение об ошибке: Ошибка клиента

Process finished with exit code 0
```