

Практика 7

1. Написать консольный калькулятор

```
fun calculator() {
    println("Введите первое число:")
    val a = readln().toDoubleOrNull() ?: run {
        println("Ошибка")
        return
    }

    println("Введите операцию (+, -, *, /):")
    val op = readln().trim()

    println("Введите второе число:")
    val b = readln().toDoubleOrNull() ?: run {
        println("Ошибка")
        return
    }

    when (op) {
        "+" -> println("Результат: ${a + b}")
        "-" -> println("Результат: ${a - b}")
        "*" -> println("Результат: ${a * b}")
        "/" -> if (b != 0.0) println("Результат: $ {a / b}") else
println("Деление на ноль")
        else -> println("Неверная операция")
    }
}

fun main() {
    calculator()
}
```

2. Найти палиндром слова

```
fun isPalindrome(word: String): Boolean {
    val lowerCaseWord = word.toLowerCase()
    return lowerCaseWord == lowerCaseWord.reversed()
}

fun main() {
    print("Введите слово:")
    val word = readLine() ?: ""

    if (isPalindrome(word)) {
        println("'${word}' - палиндром")
    } else {
        println("'${word}' - не палиндром")
    }
}
```

3. Напишите функцию, которая принимает количество побед, ничейных игр и поражений и возвращает количество очков, которая набрала команда.

```
fun calculatePoints(wins: Int, draws: Int, losses: Int): Int {
    return wins * 3 + draws * 1 + losses * 0
}

fun main() {
    print("Введите количество побед: ")
}
```

```

val wins = readLine()?.toIntOrNull() ?: 0

print("Введите количество ничьих: ")
val draws = readLine()?.toIntOrNull() ?: 0

print("Введите количество поражений: ")
val losses = readLine()?.toIntOrNull() ?: 0

val totalPoints = calculatePoints(wins, draws, losses)
println("Всего очков: $totalPoints")

print("Введите числа через пробел:")
val numbers = readLine()?.split(" ")?.mapNotNull { it.toIntOrNull() } ?:
emptyList()

if (numbers.isEmpty()) {
    val smallestNumber = numbers.minOrNull()
    println("Самое маленькое число: $smallestNumber")
} else {
    println("Список чисел пуст.")
}
print("Введите первое число: ")
val num1 = readLine()?.toDoubleOrNull() ?: return

print("Введите второе число: ")
val num2 = readLine()?.toDoubleOrNull() ?: return

println(num1 == num2)
}

```

4. Карточная игра 21

```

fun main() {
    playBlackjack()
}

fun playBlackjack() {
    fun generateCard() = listOf(2, 3, 4, 5, 6, 7, 8, 9, 10, 10, 10, 10,
11).random()
    fun calculateSum(cards: MutableList<Int>): Int {
        var sum = cards.sum()
        var aces = cards.count { it == 11 }
        while (sum > 21 && aces > 0) {
            sum -= 10
            aces--
        }
        return sum
    }

    val playerCards = mutableListOf(generateCard(), generateCard())
    val dealerCards = mutableListOf(generateCard(), generateCard())

    var p: Int
    do {
        p = calculateSum(playerCards)
        println("\nВаши карты: ${playerCards.joinToString()} (Сумма: $p)")
        if (p >= 21) break

        print("Взять карту? (да/нет): ")
        when (readln().trim().lowercase()) {
            "да" -> playerCards.add(generateCard())

            "нет" -> break
        }
    }
}

```

```
        else -> println("Ошибка")
    }
} while (true)

var d = calculateSum(dealerCards)
while (d < 17) {
    dealerCards.add(generateCard())
    d = calculateSum(dealerCards)
}

println("\n--- Итоги ---")
println("Ваши карты: ${playerCards.joinToString()} (Сумма: $p)")
println("Карты дилера: ${dealerCards.joinToString()} (Сумма:$d)")

when {
    p > 21 -> println("Вы перебрали. Проигрыш.")
    d > 21 -> println("Дилер перебрал! Вы победили")
    p == d -> println("Ничья")
    p > d -> println("Вы победили")
    else -> println("Дилер победил")
}
}
```