

## **СЦЕНАРИЙ ВИДЕО: "КОНСОЛЬНАЯ ИГРА 'КАМЕНЬ-НОЖНИЦЫ-БУМАГА' НА JAVA С НУЛЯ"**

### **ЧАСТЬ 1: ВВЕДЕНИЕ (3 минуты)**

**0:00–3:00**

"Привет, друзья! Сегодня мы создадим с нуля классическую игру 'Камень-Ножницы-Бумага' на Java — идеальный проект для начинающих, который поможет вам разобраться в основах программирования, работе с условиями, случайными числами и пользовательским вводом."

"Эта игра — отличный способ прокачать навыки Java, научиться работать с логическими операторами, циклами и создавать интерактивные консольные приложения."

Наша игра будет уметь:

- Принимать выбор игрока через консольный ввод
- Генерировать случайный выбор компьютера
- Определять победителя по правилам игры
- Вести счет побед, поражений и ничьих
- Работать в бесконечном цикле для многократных раундов
- Обрабатывать некорректный ввод пользователя

"Всё это мы напишем в одном классе, чтобы вы поняли каждый шаг кода и основы структурного программирования!"

### **ЧАСТЬ 2: ТЕХНОЛОГИИ И ИНСТРУМЕНТЫ (3 минуты)**

**3:00–6:00**

"Мы используем только стандартные возможности Java, без сторонних библиотек. Всё максимально просто и нативно."

Что применим:

- **Scanner** — для взаимодействия с пользователем и получения выбора
- **Random** — для генерации случайного хода компьютера
- **Массивы String[]** — для удобного отображения ходов
- **Циклы while** — для основного игрового цикла и проверки ввода
- **Условные операторы if-else** — для определения победителя
- **Логические операторы** — для комбинации условий
- **Обработка исключений** — для корректной работы с пользовательским вводом

"Мы создадим одну компактную программу, которая демонстрирует все ключевые концепции начального программирования на Java."

### **ЧАСТЬ 3: ЧЕМУ ВЫ НАУЧИТЕСЬ (3 минуты)**

**6:00–9:00**

"После просмотра видео вы получите не только работающую игру, но и глубокое понимание фундаментальных концепций Java."

Вы поймёте:

- Как работать с пользовательским вводом через Scanner
- Как генерировать случайные числа для игровой логики
- Как создавать и использовать массивы для данных
- Как строить сложные логические условия
- Как организовывать циклы для непрерывной работы программы
- Как обрабатывать и валидировать пользовательский ввод

"Это идеальный стартовый проект для практики алгоритмического мышления и основ Java!"

### **ЧАСТЬ 4: ФУНКЦИОНАЛ ИГРЫ (3 минуты)**

**9:00–12:00**

"Мы создадим полноценную игровую систему с подсчетом очков и удобным интерфейсом!"

Что будет у нашей программы:

- **Интуитивное меню выбора** — простой ввод 0, 1 или 2 для выбора хода
- **Честная игра** — компьютер делает случайный выбор
- **Прозрачная логика** — четкие правила определения победителя
- **Система счета** — отслеживание статистики всех игр
- **Защита от ошибок** — обработка некорректного ввода
- **Непрерывная работа** — возможность играть много раундов подряд
- **Итоговая статистика** — определение общего победителя серии

"Мы создадим систему, где каждый раунд — это независимое событие, а вся серия — это осмысленная история с финальным результатом!"

### **ЧАСТЬ 5: НАПИСАНИЕ КОДА (15 минут)**

**12:00–27:00**

"Всё максимально понятно и структурировано. Мы не просто пишем код, мы разбираем его логику и алгоритмы."

### **Блок 1: Настройка и инициализация**

- Создаём класс Main и метод main
- Инициализируем Scanner для ввода и Random для генерации ходов
- Создаём массив choices для отображения ходов
- Объявляем переменные для счета

### **Блок 2: Пользовательский интерфейс и ввод данных**

- Выводим приветствие и правила игры
- Реализуем цикл для корректного ввода выбора игрока
- Добавляем проверку hasNextInt() для защиты от некорректного ввода
- Обрабатываем ввод чисел вне диапазона 0-2

### **Блок 3: Игровая логика**

- Генерируем ход компьютера с помощью random.nextInt(3)
- Сравниваем ходы по правилам игры
- Реализуем логику определения победителя через if-else
- Обновляем счетчик побед, поражений и ничьих

### **Блок 4: Цикл игры и управление**

- Организуем главный игровой цикл с флагом playAgain
- Реализуем запрос на повторную игру
- Добавляем обработку различных вариантов ответа (да/нет)
- Выводим промежуточную и финальную статистику

"Каждый блок мы будем писать и сразу же тестировать, чтобы вы видели результат каждого этапа!"

## **ЧАСТЬ 6: ЗАВЕРШЕНИЕ (3 минуты)**

**27:00–30:00**

"Подводим итоги: Мы научились создавать интерактивные консольные приложения на Java, работать с пользовательским вводом, случайными числами и сложной логикой."

- Реализовали полноценную, функциональную игру с системой счета
- На практике разобрали работу с Scanner, Random и циклами

- Создали надежную систему обработки пользовательского ввода
- Построили четкую логику определения победителя

"Этот проект — отличная основа для дальнейшего развития. Вы можете добавить уровни сложности, расширенную версию с Ящерицей и Споком, сохранение статистики в файл и многое другое!"

"Если тебе понравился проект, поставь лайк и подпишись, впереди будет ещё больше практических проектов на Java!"

"С вами был [твоё имя], до встречи в следующем видео, где мы создадим новый полезный инструмент на Java!"

---

### **КЛЮЧЕВЫЕ АКЦЕНТЫ СЦЕНАРИЯ:**

#### **Демонстрация базовых концепций:**

- **Работа с вводом/выводом** — Scanner для ввода, System.out для вывода
- **Случайные числа** — Random для генерации ходов компьютера
- **Логические операции** — сложные условия для определения победителя
- **Циклы и условия** — управление потоком выполнения программы

#### **Практическая польза:**

- Реальная работающая игра с дружелюбным интерфейсом
- Код, который можно сразу использовать и расширять
- Фундамент для более сложных игровых проектов

#### **Образовательная ценность:**

- Понимание работы с пользовательским вводом
- Навыки построения логических условий
- Принципы организации игровых циклов
- Практика обработки ошибок и валидации данных