

ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ УЧРЕЖДЕНИЕ
ДОПОЛНИТЕЛЬНОГО ОБРАЗОВАНИЯ
«ДОМ ЮНОШЕСКОГО ТЕХНИЧЕСКОГО ТВОРЧЕСТВА»
ЦЕНТР ЦИФРОВОГО ОБРАЗОВАНИЯ ДЕТЕЙ «IT-КУБ» г. КЫШТЫМА

Методическая разработка по направлению Scratch

Тема: Создание интерактивного квеста-тренажёра «Кибер-детектив: Тайны безопасной сети»

Тип урока: Комбинированный (теория + проектная деятельность)

Направленность: техническая

Возраст обучающихся: 8–12 лет

Электронный образовательный ресурс: Scratch

Автор-составитель:

Терентьева Каролина Максимовна

педагог дополнительного образования

г. Кыштым

2026 год

Цель: Формирование навыков безопасного поведения в Интернете через создание игрового программного продукта на языке Scratch.

Задачи:

1. Образовательные: изучить понятия «фишинг», «сложный пароль», «цифровой след».
2. Развивающие: развить алгоритмическое мышление, навыки логического построения диалогов в Scratch.
3. Воспитательные: пропаганда этичного общения в сети.

1. Аннотация и инновационность

1.1. Аннотация

Данная методическая разработка представляет собой проект занятия по визуальному программированию в среде Scratch, интегрированный с модулем по информационной безопасности.

В основе работы лежит принцип «обучение через создание». Вместо пассивного потребления информации о правилах поведения в сети, учащиеся выступают в роли проектировщиков цифрового контента. Результатом занятия является интерактивный квест-тренажер «Кибер-детектив», который может быть использован в дальнейшем как самостоятельное электронное пособие для сверстников и родителей. Работа направлена на формирование не только технической грамотности, но и осознанного отношения к своим действиям в цифровом пространстве.

1.2. Инновационность и новизна

Инновационная составляющая работы заключается в объединении двух образовательных векторов: развития алгоритмического мышления и кибер-воспитания.

Ключевые элементы новизны:

1. Геймификация просвещения: Правила безопасности трансформируются из скучного перечня «нельзя» в механику игры (система очков безопасности, уровни сложности, визуальные эффекты).
2. Инструментальный подход: Использование Scratch-расширений (например, «Текст в речь» или «Видеораспознавание») для создания иммерсивного опыта. Проект перестает быть статичным текстом и становится «живым» советником.
3. Практико-ориентированная проверка: Созданный детьми тренажер проходит этап «взаимного тестирования», что позволяет учащимся взглянуть на проблемы безопасности глазами как разработчика (защитника), так и пользователя (потенциальной жертвы).
4. Масштабируемость: Разработка предусматривает создание открытого цифрового продукта, который можно опубликовать в сообществе Scratch, делая его доступным для глобального просвещения школьников по всему миру.

Новизна идеи заключается в переходе от декларативного метода (лекции) к деятельностному: ребенок сам программирует «логику злоумышленника» и «защиту героя», что гарантирует более глубокое усвоение принципов кибергигиены.

2. План занятия (45–90 минут)

Этап I. Мотивационно-ориентировочный (10 минут)

1. Деятельность педагога: задает проблемный вопрос: «Представьте, что интернет — это огромный город. В нем есть библиотеки, парки, но есть и темные переулки. Как не попасть в ловушку?» Демонстрирует короткий ролик или комикс о типичных ошибках (переход по ссылке «Вы выиграли iPhone»).
2. Деятельность учащихся: участвуют в дискуссии, делятся своим опытом (встречали ли они спам или странные сообщения).
3. Результат: Формирование интереса к теме и понимание цели — создать игрового помощника для других ребят.

Этап II. Информационно-аналитический (15 минут)

1. Деятельность педагога: вводит три ключевых игровых блока (правила), которые станут основой уровней в Scratch:
 - «Цифровая тень» (Что можно и нельзя публиковать о себе).
 - «Кибер-броня» (Как создавать надежные пароли).
 - «Фишинг-контроль» (Как распознать поддельную ссылку).
2. Деятельность учащихся: кратко записывают или зарисовывают идеи для будущих персонажей: «Кто будет антигероем (Вирус, Спам-бот)?» и «Кто будет помощником (Кибер-кот, Робот-защитник)?».

Этап III. Практический (Программирование) (45–50 минут)

Этот этап делится на три мини-задачи (алгоритма):

1. Создание интерфейса и переменных
 - Инструкция: выбрать фон (фон «Город» или «Комната»). Создать переменную Очки безопасности.
 - Логика: В начале игры задать [Очки безопасности] значение 0.
2. Программирование уровня «Проверка пароля»
 - Задача: создать проверку условий.
 - Алгоритм: * Спрайт-помощник спрашивает: «Придумай пароль для защиты своих данных».
 - Используем блок если $\langle \text{длина (ответ)} \rangle > [8] >$ то:
 - Сказать: «Отлично! Твоя броня крепка!» (+10 очков).
 - Иначе: «Слишком просто! Хакер подберет его за секунду» (0 очков).
3. Программирование уровня «Реакция на сообщение»
 - Задача: Использование ветвления.
 - Алгоритм:
 - Появляется спрайт-сообщение: «Привет! Скинь номер карты мамы, я вышлю тебе 1000 робуксов».
 - Игрок должен нажать на одну из двух кнопок: «Игнорировать/Заблокировать» или «Ответить».
 - При правильном выборе — звук победы, при ошибке — экран «Game Over» с обучающей надписью.

Этап IV. Апробация и взаимообмен (10 минут)

- Деятельность педагога: организует «мировую премьеру» — учащиеся открывают свои проекты для соседа по парте.

- Деятельность учащихся: играют в проекты друг друга, ищут баги (ошибки в коде) и проверяют, насколько понятны советы по безопасности.
- Критерий успеха: если игрок прошел все уровни и набрал максимум очков, в конце появляется сообщение: «Поздравляю! Ты — сертифицированный Кибер-детектив!».

Этап V. Рефлексия и итоги (5 минут)

- Педагог: «Какое правило безопасности было сложнее всего запрограммировать?».
- Ученики: выбирают смайлик (спрайт) в Scratch, который соответствует их настроению в конце урока.

4. Алгоритм реализации проекта в Scratch (Техническая часть)

Сцена №1: «Генератор надежности»

Суть: Персонаж (Кот или робот) просит пользователя ввести пароль. Программа проверяет его длину и наличие специальных символов.

Скрипт: Использование блока если <длина (ответ) > 8> то... иначе.... Если пароль слабый — Кот меняет костюм на «грустный» и объясняет почему.

Сцена №2: «Лабиринт ссылок»

Суть: на экране появляются всплывающие окна или сообщения от «незнакомцев». Пользователь должен кликать только на безопасные объекты (например, «Уроки», «Музыка»), избегая кнопок «Выиграл миллион» или «Введи номер мамы».

Скрипт: Использование переменных для подсчета «Жизней» или «Очков кибер-грамотности».

Сцена №3: «Чат-бот советник»

Суть: Создание ветвящегося диалога. Персонаж задает вопросы о поведении в сети, а пользователь выбирает варианты ответа.

Рабочий лист «Прототип моей игры»

(Выдается каждому ученику в начале практической части)

ФИО юного разработчика: _____

Название проекта: _____

Задание №1: Дизайн героев

Опиши, как будут выглядеть твои персонажи:

1. **Кибер-помощник (Герой):** _____ (например: Робот, Кот в очках, Мудрая Сова)
2. **Кибер-угроза (Антигерой):** _____ (например: Червь, Тень, Спрайт-вирус)

Задание №2: Логика игры (Алгоритм)

Соедини линиями событие и правильное действие:

Ситуация в игре	Действие в коде (Scratch)
Пришел странный пароль	Изменить переменную Очки на -1
Пароль длиннее 8 символов	Перейти на следующий уровень (фон)
Нажата кнопка «Заблокировать»	Сказать «Молодец!»

Задание №3: Твой «Золотой скрипт»

Нарисуй или напиши словами самый важный блок твоего кода (например, проверку пароля):

Если < _____ > то [_____] иначе [_____]

2. Памятка «Чек-лист Кибер-детектива»

(Информационный буклет для детей и родителей, который требуется по положению конкурса)

Твоя цифровая броня (Советы от разработчика)

1. **Пароль** — это ключ от квартиры. Не делай его простым (123456). Хороший пароль в Scratch-коде мы проверяем на длину более 8 знаков!
2. **Цифровой след.** Помни: всё, что попало в интернет, остается там навсегда. Прежде чем нажать «Опубликовать», подумай: «А показал бы я это директору школы?»
3. **Незнакомец в сети = Незнакомец на улице.** Если в игре или чате кто-то просит фото или адрес — это «красный флаг». Жми кнопку «Стоп» (красный восьмиугольник в Scratch и в жизни).
4. **Фейковые ссылки.** Если ссылка обещает бесплатные бонусы («робуксы», скины) — это фишинг. Не дай поймать себя на крючок!

3. Карточка-подсказка «Scratch-коды»

(Небольшая шпаргалка, которая лежит у каждого компьютера)

1. **Как сделать проверку:** Блок сенсоры -> спросить и ждать + блок управление -> если... то... иначе.
2. **Как озвучить советы:** Добавь расширение «Текст в речь» (иконка в левом нижнем углу), чтобы твой герой давал советы голосом!
3. **Как победить:** Используй блок передать сообщение [Победа], чтобы сменить фон на праздничный.