

Лекция 3:

Машина времени и Списки данных

Git, GitHub, Массивы и Циклы



Git: Машина Времени для Кода



До Git: Бардак с Файлами

- kursova.doc
- kursova_final.doc
- kursova_ispravленная.doc
- kursova_tochno_final_v2.doc

Это бардак. Сложно отследить изменения.



С Git и GitHub: Единая История



Git: Программа (Машина времени)

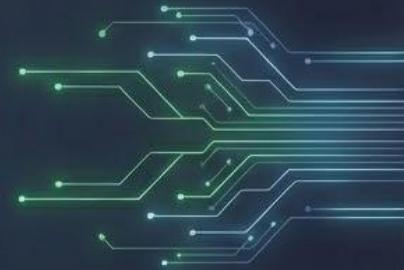
Хранит всё в одном месте, позволяет «отмотать время назад» к любой версии.



GitHub: Сайт (Облачное хранилище)

Соцсеть для кода. Безопасное хранение проекта в интернете.

Четыре Команды Выживания



Команда: `git init` (Инициализация)

Смысл: «Теперь ты Репозиторий. Git, следи за изменениями здесь».

Аналогия: Установка камер наблюдения в комнате.



Команда: `git add .` (Добавить всё)

Смысл: «Я хочу сохранить ВСЕ файлы, которые изменились». Точка — текущая папка.

Аналогия: Мы собрали вещи в чемодан, но еще не закрыли его.



Команда: `git commit -m "Комментарий"` (Зафиксировать)

Смысл: Мы делаем снимок. «Сейв» в игре. Флаг `-m`: обязательное сообщение.

Аналогия: Закрыли чемодан, повесили бирку и сдали в багаж.



Команда: `git push` (Толкнуть)

Смысл: Отправить код на GitHub.

Аналогия: Загрузка фото в Инстаграм. Код останется в интернете.

Работа с кучей файлов: Массивы



Что такое массив?



Специальная переменная-контейнер.
Выглядит как поезд с вагонами. Синтаксис: [].

```
// Плохо:  
const file1 = "cat.jpg";  
const file2 = "dog.png";
```

```
// Хорошо (Массив):  
const files = ["cat.jpg", "dog.png",  
"document.pdf"];
```

Главные правила и методы



Индексы: Компьютеры считают с Нуля.

```
console.log( files[0] ); // "cat.jpg"
```



Свойство Length (Длина): Как узнать количество?

```
console.log( files.length ); // 3
```



Добавление данных (Push):
Прицепить новый вагон.



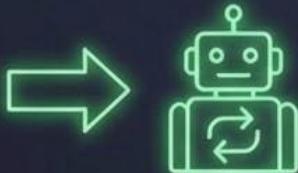
Автоматизация: Введение в Циклы



Проблема и Решение



Если у вас 10 000 файлов, вы не будете писать код для каждого.



Цикл: Мы напишем один алгоритм и заставим компьютер повторить его 10 000 раз.

Аналогия: Повторение действия



Аналогия: «Пока в коробке есть конфеты — ешь».

```
// Пример: Обработка 10 000 файлов
for (let i = 1; i <= 10000; i++) {
    console.log(`Обработка файла №${i}`);
    // Алгоритм обработки
}
console.log("Все файлы обработаны.");
```

Домашнее Задание



Задание 3: Поиск с прерыванием

Создайте массив с пин-кодами: ["1111", "5678", "1234", "9999"].

Создайте переменную `myPin = "1234".`

Напишите цикл, который перебирает массив.

Внутри цикла проверяйте: если пин-код совпадает с `myPin`:

Выведите: "Пин-код найден!"

Остановите цикл командой `break`.

Для всех остальных пишите: "Неверный пин...".

```
const pinCodes = ["1111", "5678", "1234",  
"9999"];  
const myPin = "1234";
```

```
// Ваш цикл и условие с break здесь
```

```
// ...
```