



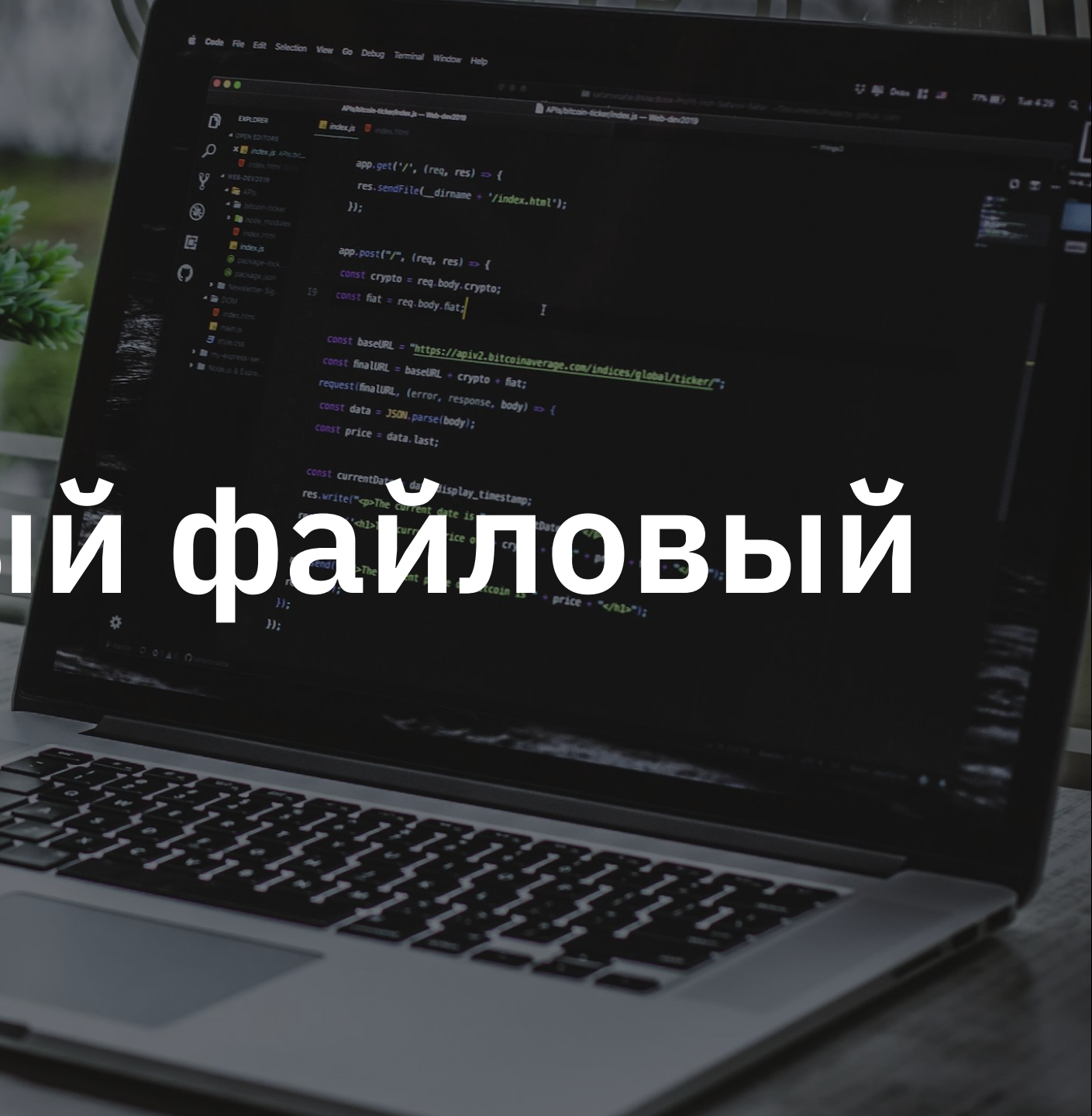
# Итоговый проект

Введение в C#



# Консольный файловый менеджер

Введение в C#



# Что будет на уроке

1. Что нужно сделать
2. Разбор требований к  
итоговому проекту
3. Разбор требований к сдаче
4. Для чего этот проект

Тема итогового проекта - консольный файловый менеджер **начального** уровня.

Что значит начального?

- Минимальный рабочий проект, который удовлетворяет требованиям
- Используем знания предыдущих уроков
- **Работаем над функционалом**, а не над UI
- Учимся работать с ТЗ тестового задания

# Что должно получиться?

C:\CanoScan\CNQL20\CNQSG70

22-01-2011 23:20:09

C:\

+ \$Recycle.Bin

+ CanoScan

+ CNQL20

+ CNQL20

+ CNQSG70

+ CNQL30

+ Documents and Settings

+ Dokumente und Einstellungen

+ Inprise

+ PerfLogs

+ Program Files

+ ProgramData

+ Programme

+ Recovery

+ System Volume Information

+ temp

+ totalcmd

+ LANGUAGE

+ Users

+ Windows

+ addins

+ AppCompat

+ Programs

+ AppPatch

+ assembly

+ BitLockerDiscoveryVolumeContents

1

FILE \*.\*

DISK C:50GB\_System

Available

13,406,224,384

DISK Statistics

Total

Files 150,942

Bytes 18,425,200,027

Matching

Files 150,942

Bytes 18,425,200,027

Tagged

Files 5

Bytes 1,303,540

Current Directory

CNQSG70

Files 24

Bytes 7,190,901

Attributes .....

3

2

◆CANOIT32	.EXE	45,056	.a..	17-06-1998	0:14:00
◆CISDS	.DS	61,440	.a..	29-01-2003	1:45:04
◆CNQU70	.DLL	36,864	.a..	12-09-2002	0:07:00
◆CNZ005	.ICC	1,012	.a..	26-05-1999	23:45:44
◆CSUI	.DLL	1,159,168	.a..	29-01-2003	1:45:04
CSUI_RES	.DLL	581,632	.a..	19-01-2003	16:59:04
DEV	.DLL	94,208	.a..	29-01-2003	1:45:04
IMGENH	.DLL	135,168	.a..	29-01-2003	1:45:04
IOP	.DLL	1,597,440	.a..	29-01-2003	1:45:04

ZTreeWin v2.1.068

Zedtek, Inc.(c) 2010

# Требования



## Функции менеджера

1. Вывод дерева файловой системы с условием “пейджинга” - только два уровня!
2. Копирование каталога
3. Копирование файла
4. Удаление каталога рекурсивно
5. Удаление файла
6. Вывод информации о каталоге
7. Вывод информации о файле



# Примеры комманд

## Не важно, как команды называются

Вывод дерева файловой системы  
с условием “пейджинга”

**ls C:\Source -p 2**

Копирование каталога

**cp C:\Source D:\Target**

Копирование файла

**cp C:\source.txt D:\target.txt**

Удаление каталога рекурсивно

**rm C:\Source**

Удаление файла

**rm C:\source.txt**

Вывод информации

**file C:\source.txt**

# Что такое пейджинг

С этим понятием вы будете сталкиваться часто

Пейджинг - это вывод информации по-странично. Как, к примеру, каталог в любом веб магазине.

## Пример:

1. Допустим у вас есть 200 файлов в каталоге
2. Каждая страница - это 100 файлов, задано программно
3. Первая страница выведет первые 100 файлов
4. Вторая страница выведет с 100 по 200 файл



# Что нужно применить

## Из прошлых уроков

1. Урок 5, пространство System.IO
2. Урок 6, работа с ловлей ошибок
3. Урок 4, рекурсия
4. Все остальные уроки - работа со строками, перечисление, массивы, методы
5. Настройка отображение страницы - должно быть задано в конфигурационном файле
6. При выходе нужно запоминать последнее состояние (активный каталог)
7. Задание на звездочку: при нажатии вверх либо вниз движение по истории команд

# Как нужно сдавать

## Новый репозиторий

1. Делайте отдельный репозиторий на GitHub'е
2. Используйте комментарии в коде
3. Обязательно описание алгоритма работы и readme.md вместе со скриншотом
4. Фокус на функционал - не UI
5. Присылаете ссылку на репозиторий

# Вопросы?