

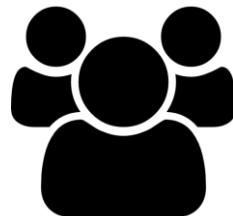
Web Basis: Логика

WEB
COURSE FOR **QA**
ORT DNIPRO

ORTDNIPRO.ORG/QA

Где взять материалы?

Наша группа: QA16



qa16.ortdnipro.org

1. Интерактивное Программное Обеспечение

Example ViewPoint Document Close Save Reset Save&Edit

Brother Dominic 12294 Free Disk Pages Help

XEROX 6085 Workstation

User-Interface Design

To make it easy to compose text and graphics, to do electronic filing, printing, and mailing all at the same workstation, requires a revolutionary user interface design.

Bit-map display - Each of the pixels on the 19" screen is mapped to a bit in memory; thus, arbitrarily complex images can be displayed. The 6085 displays all fonts and graphics as they will be printed. In addition, familiar office objects such as documents, folders, file drawers and in-baskets are portrayed as recognizable images.

The mouse - A unique pointing device that allows the user to quickly select any text, graphic or office object on the display.

See and Point

All functions are visible to the user on the keyboard or on the screen. The user does filing and retrieval by selecting them with the mouse and touching the MOVE, COPY, DELETE or PROPERTIES command keys. Text and graphics are edited with the same keys.

Shorter Production Times

Experience at Xerox with prototype work stations has shown shorter production times and thus lower costs, as a function of the percentage of use of the workstations. The following equation can be used to express this:

$$X(n) = \sum_{i=1}^n \frac{A + FP^i}{denominator + }$$

Figure 1: Data from Table 1 drive

NAME	EXTENSION	SIZE	DATE
COMMAND	COM	22677	15-N-
ANSI	SYS	2556	18-S-
ASSIGN	COM	864	28-N-
ATTRIB	EXE	15091	14-N-
BACKUP	COM	17024	28-A-
CHKDSK	COM	9435	24-O-
CHMOD	COM	6528	27-A-
COMP	COM	3018	10-S-
DEBUG	EXE	15364	15-N-

Text and Graphics

To replace typesetting, the 6085 offers a choice of type fonts and sizes, from 6 point to 36 point:

Here is a sentence of 6-point text.
Here is a sentence of 10-point text.
Here is a sentence of 12-point text.

18-point text.
24-point text.
36-point text.

Table 1: Percentages of use of methods.

Year	Non 6085	6085
1978	85.2	15.8
1980	61.1	39.9
1982	45	55
1984	30	70
1986	10	90
1988	5	95

Activity under the old and the new system

Figure 1: Data from Table 1 drive

1978
1980
1982
1984
1986
1988

0 50 100

DOS & Lotus data Close Redisplay

NAME EXTENSION SIZE DATE

COMMAND COM 22677 15-N-

ANSI SYS 2556 18-S-

ASSIGN COM 864 28-N-

ATTRIB EXE 15091 14-N-

BACKUP COM 17024 28-A-

CHKDSK COM 9435 24-O-

CHMOD COM 6528 27-A-

COMP COM 3018 10-S-

DEBUG EXE 15364 15-N-

2.0 TTY Monthly Profit Blank Folder

Beechnut C ILLUS Blank Illustrator

Blank Canvas

PC Converter Blank Shared Book

Emulator Emulated Rigid Disk Virtual Floppy

Example ViewPo Remote Files

Emulated Rigid Disk Virtual Floppy

Drawers in Japan Swaps DOS & Lotus

Mackey OSBU Xerox Tape Drive

Floppy Drive Wastebasket

Directory

9:27:24 10-29-88

N.H. Local Kevin J. Outbaske

Mail Merge Mail from Ken

Calendar Calc Loader

Blank User Dictionary

Blank Record File

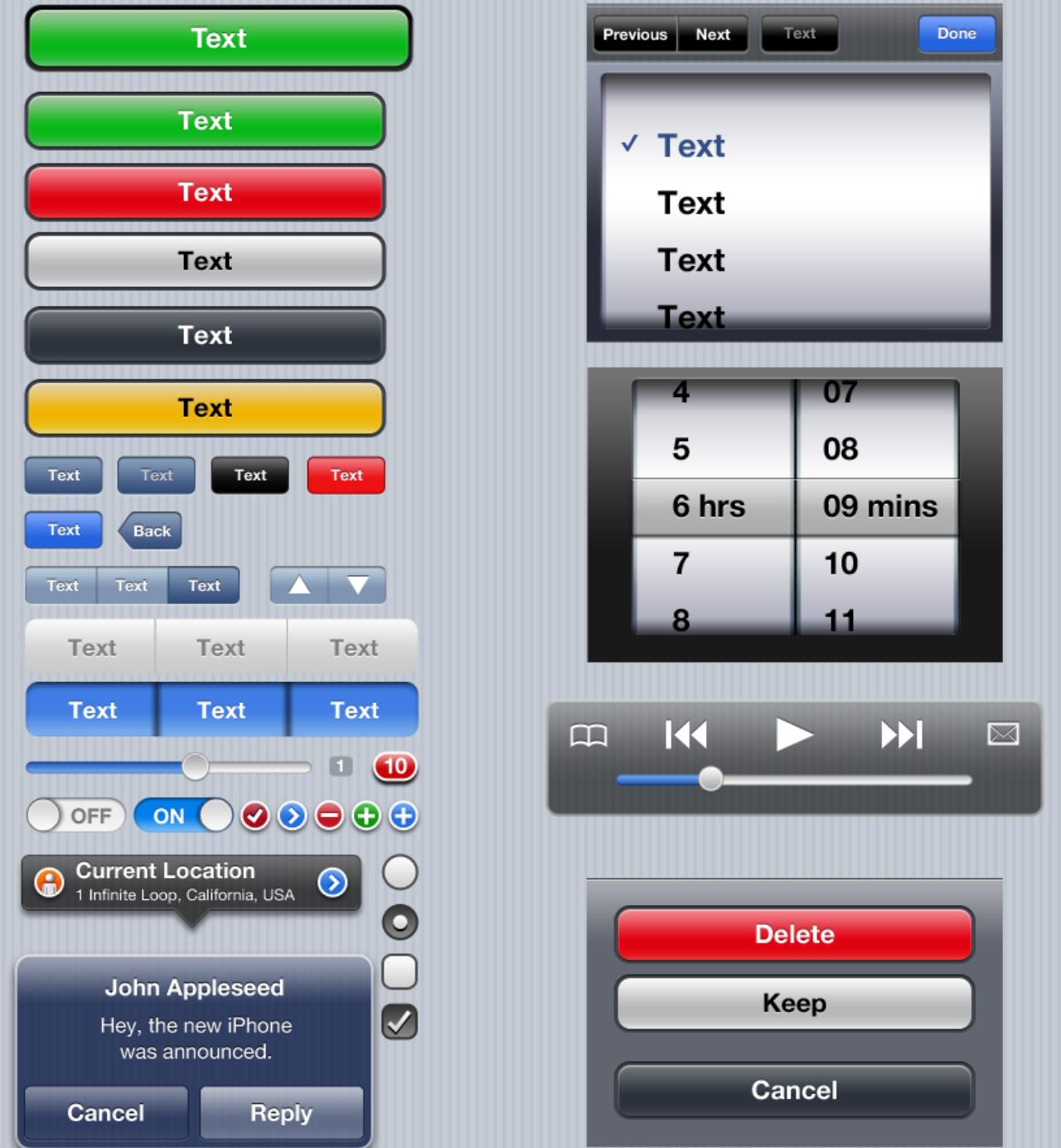
Blank Document

Blank Book

Blank Reference

4427

Скевоморфизм - орнамент или элемент дизайна, который скопирован с формы другого объекта, но изготовлен из других материалов или иным методом. [/Википедия](#).



Элементы интерфейса первых версий iOS.

Скевоморфизм - орнамент или элемент дизайна, который скопирован с формы другого объекта, но изготовлен из других материалов или иным методом.
[/Википедия.](#)

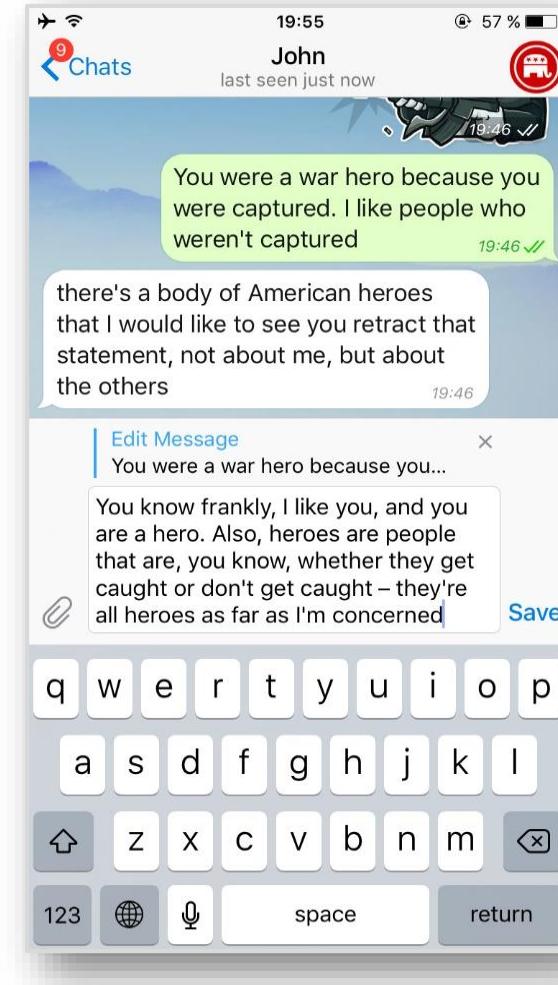
Интерактивное программное обеспечение



Событийная модель управления

https://ru.wikipedia.org/wiki/Событийно-ориентированное_программирование

Интерактивное программное обеспечение

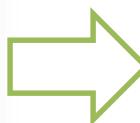
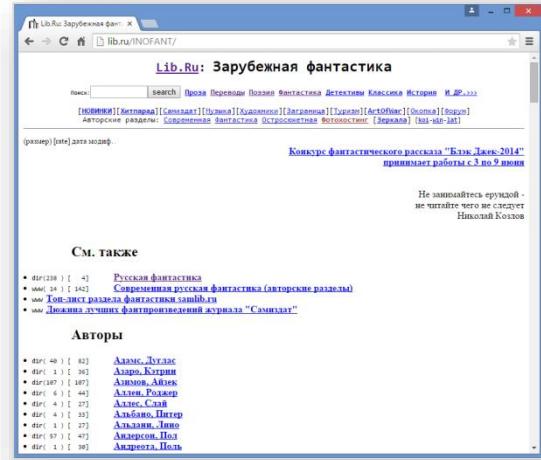
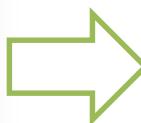


2. Интерактивное программное обеспечение

...в браузере...

HTML статичен

Что неудивительно, ведь **HTML (и CSS)** не является языком программирования.



После того как страница загрузиться в браузер она остаётся неизменной, информация на ней не изменяется. Чтобы получить другую информацию, нужно загрузить другую страницу. Однако пользователи (поработав с **настольным программным обеспечением**) привыкли к какой-никакой но **интерактивности**.

HTML/CSS – декларативные языки

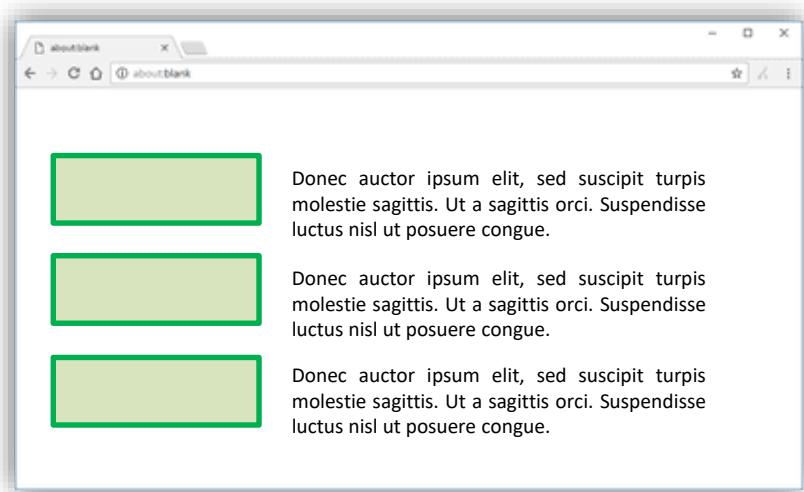
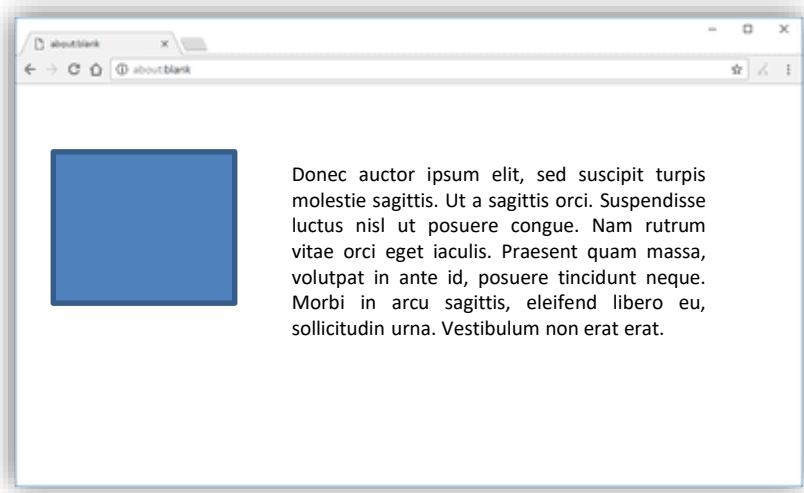
Декларативные языки при помощи директив позволяют указать какой результат мы хотим получить, но не путь его достижения (путь его достижения определяет компьютер). Побочный эффект: всё что не предусмотрено имеющимися директивами – реализовать не получится.

JavaScript – императивный язык

Императивные языки (как правило языки программирования относятся к этой категории) – языки состоящие из инструкций (малых действий, «шагов») выполняемых в определённой последовательности. Тем самым код на императивном языке описывает путь достижения желаемого результата.

Отличительная черта императивных языков: работа (обработка) данных, описание логики (ветвления) работы программы.

Задачи JavaScript?



Изменение
содержимого
страницы (*в первую
очередь **данных***).
А за содержимое
страницы отвечает
разметка...

Задачи JavaScript

1. Манипуляция элементами (тегами) HTML-страницы
(когда страница уже в браузере посетителя);

А если конкретнее, то: изменять разметку документа. Ведь браузер «нарисует» только то что описано в разметке.

2. Делать что-то в ответ на действия пользователя
(реагировать на действия пользователя).

3. На практике

Hello, JS!



Click on the photo to display the next one

Photo #1

«Листалка» фотографий

Задача **JavaScript** – изменять разметку страницы, ссылка на изображение в теге **** тоже относиться к разметке страницы...

Скачайте архив проекта по ссылке:
<https://firebase.xyz/storage/qa/javascript-example.zip>

4. JavaScript – язык программирования

HTML



CSS



JS



Три базовые технологии Front End разработки

5. Алгоритм

Алгоритм

Задача: Скрипт должен рассчитывать сколько гривен **в день** приносит вкладчику депозит размещенный на **полтора года** по ставке в **20% годовых**?

Проблемы:

- ✓ Дан недостаточный объём данных или часть данных задана неявно, нужно уточнять;
- ✓ Часть данных избыточна;
- ✓ Есть сторонние факторы, не известные заранее, но влияющие на результат.

Алгоритмы

Задача: Скрипт должен рассчитывать сколько гривен **в день** приносит вкладчику депозит размещенный на **полтора года** по ставке в **20% годовых**?

Алгоритм:

1. Уточняем сумму депозита.

2. Рассчитываем сколько будет дохода за целый год:

$$\text{Доход} = \text{Сумма} * (20\% / 100);$$

3. Считаем доход за 1 день:

$$\text{Доход_день} = \text{Доход} / \text{Количество_дней_в_году};$$

4. Расчитываем налоги:

$$\text{Сумма_налога} = \text{Доход_день} * ((18\% + 1,5\%) / 100);$$

5. Учитываем налог:

$$\text{Доход_день_после_налога} = \text{Доход_день} - \text{Сумма налога};$$

6. Выводим результаты.

Будет полезным

Пора двигаться дальше...



Предварительные знания – лучший помощник в обучении, поэтому к следующему занятию жду, что **Вы обязательно посмотрите ролик...**

https://youtu.be/SW_UCzFO7X0?t=167