



dimon.work/kurs.html

1. Oprogramowanie interaktywne

XEROX 6085 Workstation

User-Interface Design

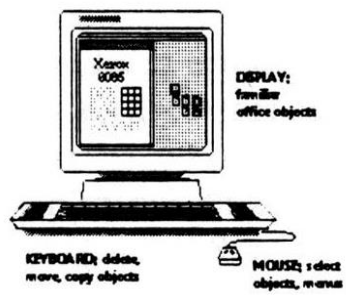
To make it easy to compose text and graphics, to do electronic filing, printing, and mailing all at the same workstation, requires a revolutionary user interface design.

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:

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

Table 1: Percentages of use of methods.

Activity under the old and the new

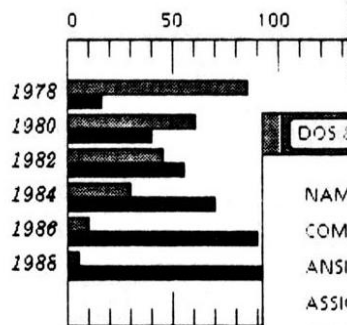


Figure 1: Data from Table 1 drive

$$X(n) = \sum_{i=1}^n \int_a^b \frac{A + PF^i}{denominator + ...}$$

Workstation usage percentages Table 1 and illustrated in Figure 6085 users are likely to do more composition and layout, control process including printing and di

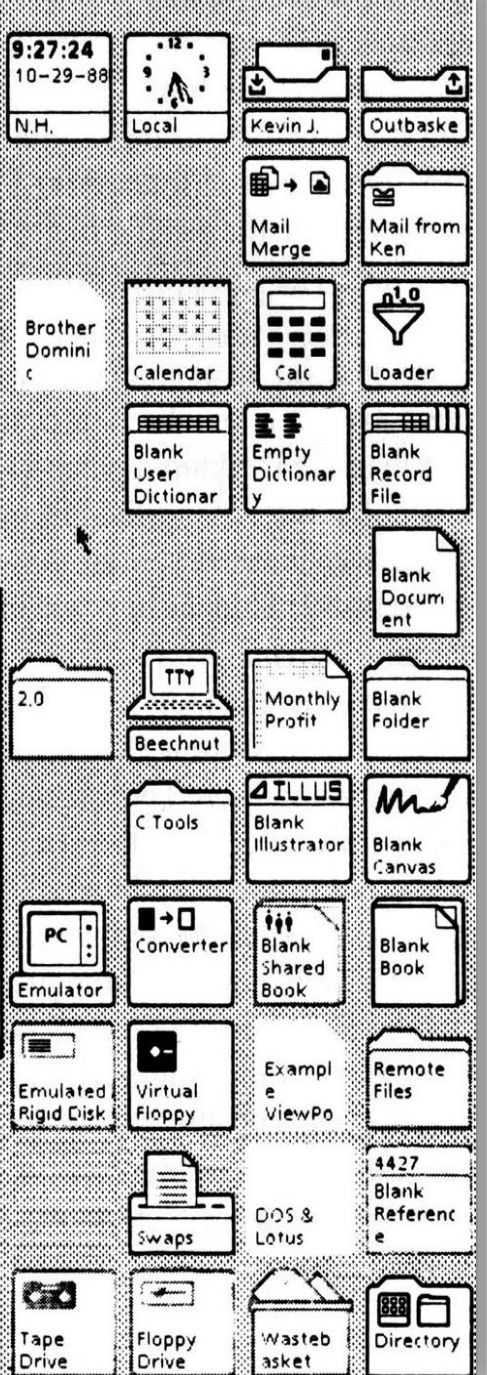
Text and Graphics

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.



NAME	EXTENSION	SIZE	DATE
COMMAND	COM	22677	15-N
ANSI	SYS	2556	18-S
ASSIGN	COM	364	28-N
ATTRIB	EXE	15091	14-N
BACKUP	COM	17024	28-A
CHKDSK	COM	9435	24-C
CHMOD	COM	6528	27-A
COMP	COM	3018	10-S
DEBUG	EXE	15364	15-N



Skeumorfizm - ornament lub element designu, który jest kopiowany z formy innego obiektu, ale wykonany z innych materiałów lub inną metodą. /Википедия.



Elementy interfejsu pierwszych wersji iOS.

Skeumorfizm - ornament lub element designu, który jest kopiowany z formy innego obiektu, ale wykonany z innych materiałów lub inną metodą [/Википедия](#).

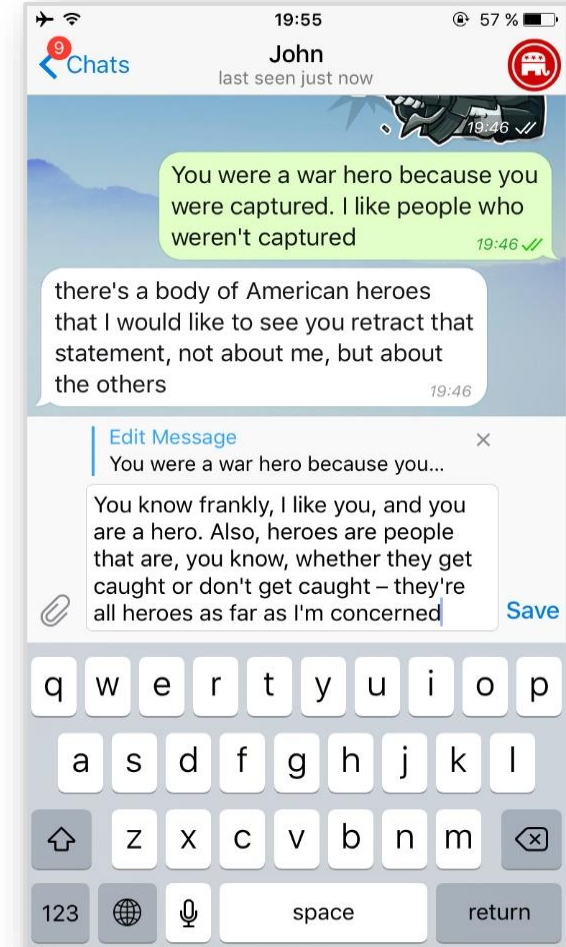
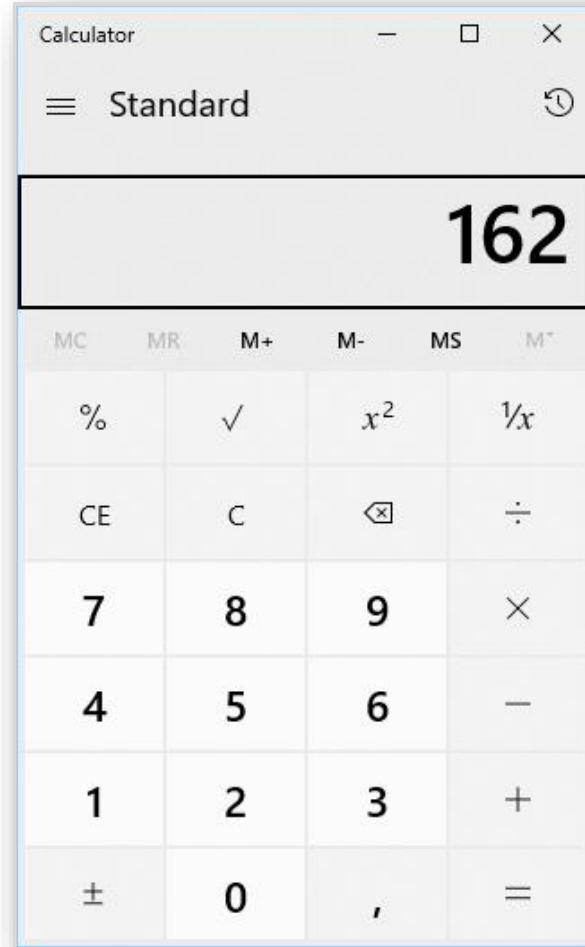
Opragromawanie interaktywne



Zdarzeniowy model sterowania

https://en.wikipedia.org/wiki/Event-driven_programming

Oprogramowanie interaktywne

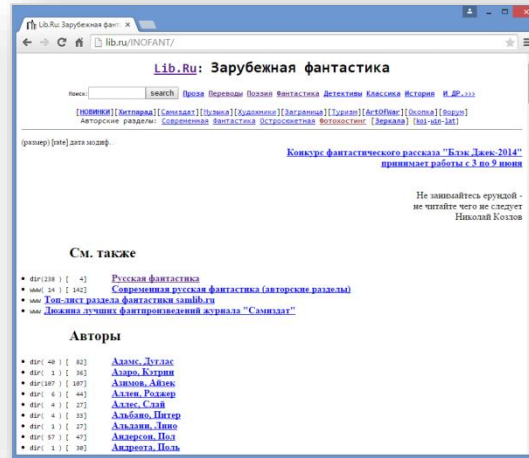


<https://en.wikipedia.org/wiki/Interactivity>

2. Oprogramowanie interaktywne ...w przedtądarce...

HTML jest statyczny.

Nie jest to zaskakujące, ponieważ **HTML** (i **CSS**) nie jest językiem programowania.



Po załadowaniu strony do przeglądarki pozostaje ona niezmienną. Aby uzyskać inną informację, należy załadować nową stronę. Jednak użytkownicy (przyzwyczajeni do pracy z oprogramowaniem desktopowym) oczekują pewnej **interaktywności**.

HTML/CSS – języki deklaratywne

Deklaratywne języki pozwalają za pomocą dyrektyw wskazać, jaki efekt chcemy osiągnąć, ale nie sposób, w jaki do tego dojść (komputer ustala sposób realizacji).
Efekt uboczny: to, czego nie przewidują dostępne dyrektywy, nie może zostać zrealizowane.

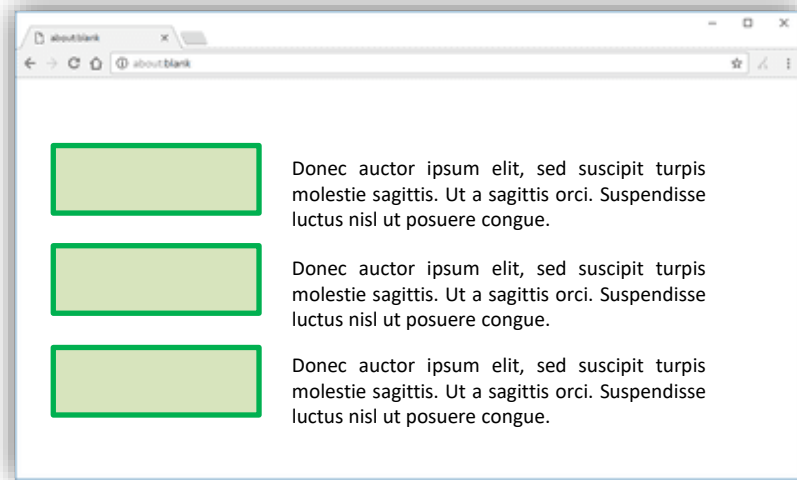
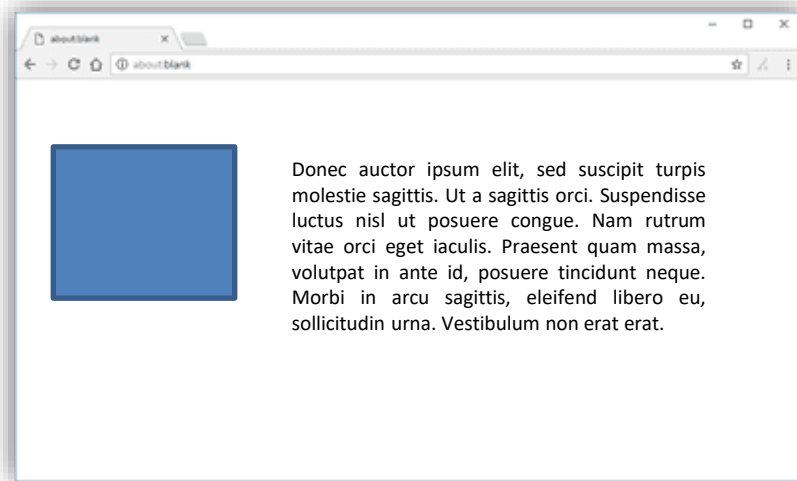
JavaScript – język imperatywny

Imperatywne języki (zazwyczaj należą do nich języki programowania) składają się z instrukcji (małych działań, „kroków”) wykonywanych w określonej kolejności.

Kod w języku imperatywnym opisuje sposób osiągnięcia oczekiwanego wyniku.

Charakterystyczna cecha języków imperatywnych: praca z danymi, opis logiki działania programu.

Zadania JavaScript?



Zmiana zawartości
strony (głównie
danych).

A znaczniki są
odpowiedzialne za
zawartość strony...

Zadania JavaScript

1. Manipulowanie elementami strony HTML (znacznikami)
(gdy strona znajduje się już w przeglądarce użytkownika);

A dokładniej: modyfikować znaczniki dokumentu. Ponieważ przeglądarka „narysuje” tylko to, co jest opisane w znacznikach. .

2. Udzielać odpowiedzi na działania użytkownika
(reagować na działania użytkownika).

3. W praktyce

Hello, JS!



Click on the photo to display the next one

Photo #1

zdjęćGaleria

Zadanie **JavaScript** – zmieniać strukturę stronę, link do obrazu w znaczniku `` również należy do struktury strony ...

Skorzystaj z szablonu w repozytorium lekcji [dimon.work/kurs/js-demo-example.zip](https://github.com/dimon-work/kurs-js-demo-example)

4. JavaScript – język programowania

HTML



CSS



JS



Trzy podstawowe technologie Front-Endu:

JavaScript – język programowania

Jego «elementy składowe»

Zmienne / Typy / Operacje

Wstawienie (Instrukcje warunkowe)

Pętle / Tablice (struktury danych)

Funkcje

Obiekty

5. Algorytm

Algorytm

Zadanie : Skrypt powinien obliczać, ile PLN dziennie przynosi użytkownikowi depozyt ulokowany na półtora roku przy rocznej stopie procentowej 4.5%.

Problemy :

- ✓ *Dostępne dane są niepełne lub część danych jest ukryta, trzeba je doprecyzować.*
- ✓ *Część danych jest nadmiarowa.*
- ✓ *Istnieją czynniki zewnętrzne, które mogą wpłynąć na wynik, ale nie są znane z góry.*

Algorytmy

Zadanie : Skrypt powinien obliczać, ile PLN **dziennie** przynosi użytkownikowi depozyt ulokowany na **półtora roku** przy rocznej stopie procentowej **4.5%**.

Algorytm:

1. Doprecyzowanie kwoty depozytu.

2. Obliczenie rocznego dochodu:

$$\text{Dochód} = \text{Kwota} * (4.5\%/100);$$

3. Obliczenie dziennego dochodu:

$$\text{Dochód_dzień} = \text{Dochód} / \text{Liczba_dni_w_roku};$$

4. Obliczenie podatku:

$$\text{Kwota_podatku} = \text{Dochód_dzień} * ((19\% + 1,5\%)/100);$$

5. Uwzględnienie podatku:

$$\text{Dochód_dzień_po_podatku} = \text{Dochód_dzień} - \text{Kwota_podatku};$$

6. Wyświetlenie wyników.