-= 02.06.2014 =-

staff:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| № | ФИ | Класс | телефон | E-mail |
| 01 | Носков Иван | 10-1 |  | Vanyamagov@yandex.ru |
| 02 | Корнеев Олег | 10-2 |  |  |
| 03 | Шульц Святослав | 10-3 | +79218938047 | Svsch9@yandex.ru |
| 04 | Сокол Алексей | 10-4 | +79119747062 | kanootoko@gmail.com |
| 05 | Дунаева Ольга | 10-6 |  |  |
| 06 | Пантелеева Алиса | 10-6 |  |  |
| 07 | Яковлев Михаил | 10-6 | 89215589606 | yakovmikemikle@gmail.com |
| 08 | Гаевой Никита | 10-1 | +79219521012 | nikgaevoy@mail.ru |
| 09 | Абрамова Мария | 10-4 | +79214026088 | maria.abramova1545@gmail.com |
| 10 | Чепцов Василий | 10-4 | +79522401016 | vcheptsov97@gmail.com |

10:00 - start

SPR?? - account

***Far 3 - запуск***

Shift+F4 -> ***start.bat***

>>> net use q: \\server\shspr

F7 -> INFO

***Z:\INFO\***

Shift+F4 -> ***dirinfo***

>>> Familija Imja

>>> 10-?

>>> dd.mm.yyyy

>>> SPR??

>>> E-mail

11:40 start winapi

12:25 WM\_PAINT:

case WM\_PAINT:

- сообщение перерисовки окна

HDC hDC;

PAINTSTRUCT ps;

case WM\_PAINT:

hDC = BeginPaint(hWnd, &ps);

. . .

SetPixel(hDC, 100, 100, RGB(0, 255, 0));

. . .

EndPaint(hWnd, &ps);

получение размера окна:

RECT rc;

GetClientRect(hWnd, &rc);

w = rc.right;

h = rc.bottom;

смена цвета рисования

SelectObject(hDC, GetStockObject(NULL\_PEN));

SelectObject(hDC, GetStockObject(DC\_BRUSH));

SetDCBrushColor(hDC, RGB(255, 0, 0));

или

SelectObject(hDC, GetStockObject(DC\_PEN));

SetDCPenColor(hDC, RGB(255, 0, 0));

перерисовка - отсылка сигнала

InvalidateRect(hWnd, NULL, FALSE/TRUE очистка);

T01FWIN - добавить "следящие глаза"

про "мышь":

1.от сообщений

WM\_MOUSEMOVE:

WM\_LBUTTONDOWN:

WM\_LBUTTONUP:

R

M

wParam - состояние кнопок:

if (wParam & MK\_LBUTTON) . . .

lParam - координаты мыши в окне

x = (SHORT)LOWORD(lParam);

y = (SHORT)HIWORD(lParam);

2.от таймера

- при создании окна - зарегистрировать "таймер"

- приходит WM\_CREATE (надо вернуть 0, -1 - для отмены создания окна)

SetTimer(hWnd, TimerNo, TimeInMs, NULL);

(в WM\_DESTROY -> KillTimer(hWnd, TimerNo);

будет приходить сообщение WM\_TIMER:

- опросить мышу:

POINT pt;

GetCursorPos(&pt); - глобальные координаты

ScreenToClient(hWnd, &pt); - преобразование координат в СК окна

x = pt.x;

DrawEye(hDC, pt.x, pt.y, ... );

void DrawEye( HDC hDC, INT Xc, INT Yc, INT W, INT H )

{

INT x, y;

E(0, 0, W / 2, H);

E(W / 2, 0, W, H);

len = ... ;

x = W / 4;

y = H / 2;

co = (Xc - x) / len;

si = ... ;

xe = x + co \* len;

ye = y + si \* len;

E(xe - r, ye - r, xe + r, ye + r);

}

!!! InvalidateRect(hWnd, NULL, TRUE);

17:00 про GIT

Git - www.git-scm.com

SmartGit - http://www.syntevo.com/smartgithg/

GIT

- repository -

- stage - пометить на изменение

- commit - отослать помеченное

- www.github.com - зарег.

>>> www.gravatar.org

Д.з. - полностью установить Git SmartGit, GitHub,

Gravatar

-= 03.06.2014 =-

повтор:

**WM\_CREATE:**

CREATESTRUCT \*cs;

cs = (CREATESTRUCT \*)lParam;

cs->hInstance ... - все параметры из CreateWindow

**WM\_SIZE:** - при изменении размера окна

INT w = LOWORD(lParam), h = HIWORD(lParam);

**WM\_CLOSE:** - при попытке зактрыть окно

можно отказать, можно вызвать DestroyWindow(hWnd);

**WM\_DESTROY:** - при уничтожении (сообщение о случившемся)

- для главного окна программы PostQuitMessage(0);

**WM\_PAINT:** - для восстановления содержимого

PAINTSTRUCT ps;

HDC hDC;

hDC = BeginPaint(hWnd, &ps);

. . .

EndPaint(hWnd, &ps);

инициировать полную перерисовку -

InvalidateRect(hWnd, NULL, TRUE); TRUE - все закрасить фоном

**WM\_ERASEBKGND:**

- перерисовка фона

HDC hDC = (HDC)wParam;

return 1; - при закраске

**WM\_TIMER:**

- сигнал от таймера

INT TimerNo = wParam;

создать: SetTimer(hWnd, TimerNo, DelayTimeInMs, NULL);

уничтожить: KillTimer(hWnd, TimerNo);

**WM\_MOUSEMOVE:** - перемещение

**WM\_LBUTTONDOWN:** - нажатие

**WM\_LBUTTONUP:** - отпускание MBUTTON RBUTTON

**WM\_LBUTTONDBLCLK:** - двойное нажатие !!! в классе окна

wc.style = ... | CS\_DBLCLK;

**WM\_MOUSEWHEEL:** - кручение колеса

INT z = (SHORT)HIWORD(wParam); - колесико

для всех мышиных сообщений:

INT

x = (SHORT)LOWORD(lParam);

y = (SHORT)HIWORD(lParam);

в wParam - биты кнопок:

MK\_LBUTTON ...

MK\_SHIFT

MK\_CONTROL

!!! SetCapture(hWnd); - заставляет все сообщения от мыши засылать в указанное окно

SetCapture(NULL); - отмена

**WM\_KEYDOWN:** - нажатие

Key = wParam;

VK\_LEFT VK\_UP ...

'A' 'B' ... 'Z'

'1' '2'

(WORD)lParam - счетчик автоповтора

WORD == unsigned short

**WM\_KEYUP:** - отжатие

**WM\_CHAR:** - нажатие символа

CHAR ch = wParam; <-- TranslateMessage(&msg) в цикле сообщений

- 10:45

Двойная буферизация

совместимый контекст - контекст для рисования в памяти:

HDC hMemDC = CreateCompatibleDC(hDC); hDC - контекст окна

удалить - DeleteDC(hMemDC);

создание изображения в памяти для рисования:

HBITMAP hBm = CreateCompatibleBitmap(hDC, W, H);

удалить DeleteObject(hBm);

выбираем его в контекст для рисования:

SelectObject(hMemDC, hBm);

LineTo(hMemDC, 30, 30);

рисование контекста в дрyгой контекст:

bit-block-transfer

BitBlt(hDC\_куда, Xкуда, Yкуда, Wкуда, Hкуда,

hDC\_откуда, Xоткуда, Yоткуда, SRCCOPY);

StretchBlt(hDC\_куда, Xкуда, Yкуда, Wкуда, Hкуда,

hDC\_откуда, Xоткуда, Yоткуда,

Wоткуда, Hоткуда, SRCCOPY);

`

12:00

- про глаза

T02CLOCK

- часы со стрелками - циферблат - faceclock

!!!

POINT pt[5] = {{1, 2}, {11, 30}, ... };

Polygon(hDC, pt, 5);

12:40 - access to Git (register and create repository)

T02CLOCK

HFONT hFnt;

**hFnt = CreateFont(211, 0, 50, 0, FW\_BOLD,**

**TRUE, FALSE, FALSE, RUSSIAN\_CHARSET,**

**OUT\_DEFAULT\_PRECIS, CLIP\_DEFAULT\_PRECIS,**

**PROOF\_QUALITY, VARIABLE\_PITCH | FF\_ROMAN,**

**"Times");**

**-= 04.06.2014 =-**

**Рисование стрелок (hand draw)**

**1.поворот точки вокруг начала координат**

**x=r·cos(β)**

**y=r·sin(β)**

**x'=r·cos(β+α)=r·cos(β)·cos(α)-r·sin(β)·sin(α)=x·cos(α)-y·sin(α)**

**y'=r·sin(β+α)=r·cos(β)·sin(α)+r·sin(β)·cos(α)=x·sin(α)+y·cos(α)**

**VOID DrawArrow( HDC hDC, INT Xc, INT Yc, INT L, INT W, FLOAT Angle )**

**{**

**INT i;**

**POINT pts[] =**

**{**

**{0, -W}, {-W, 0}, {0, L}, {W, 0}**

**}, pts\_draw[sizeof pts / sizeof pts[0]];**

**FLOAT si = sin(Angle), co = cos(Angle);**

**for (i = 0; i < sizeof pts / sizeof pts[0]; i++)**

**{**

**pts\_draw[i].x = Xc + pts[i].x \* co - pts[i].y \* si;**

**pts\_draw[i].y = Yc + pts[i].x \* si + pts[i].y \* co;**

**}**

**Polygon(hDC, pts\_draw, sizeof pts / sizeof pts[0]);**

**}**