

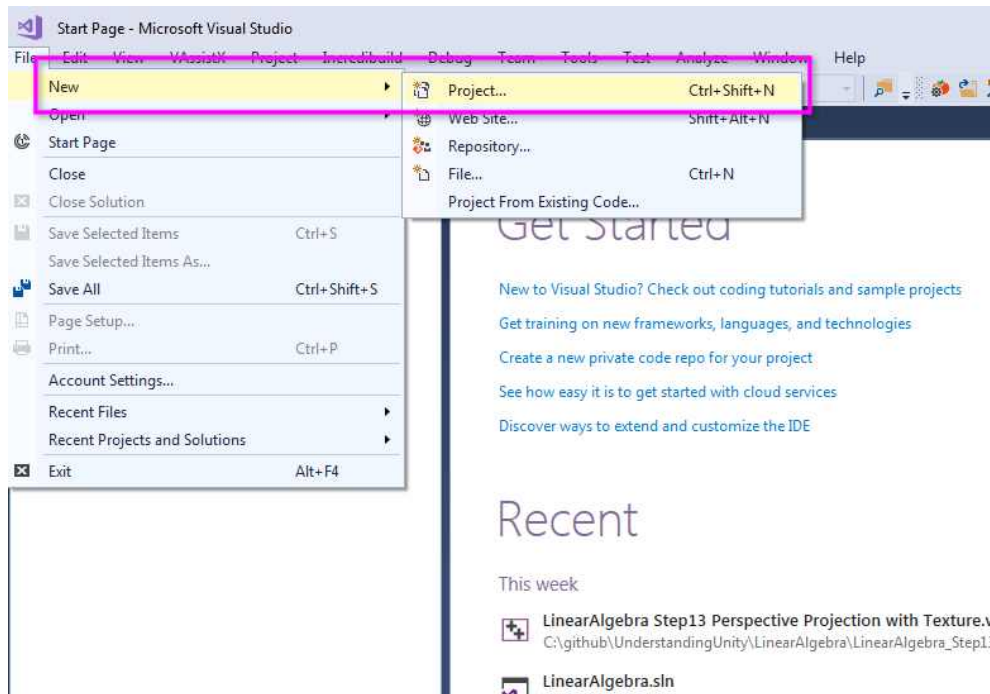
Windows Programming

Windows GDI(Graphics Device Interface)

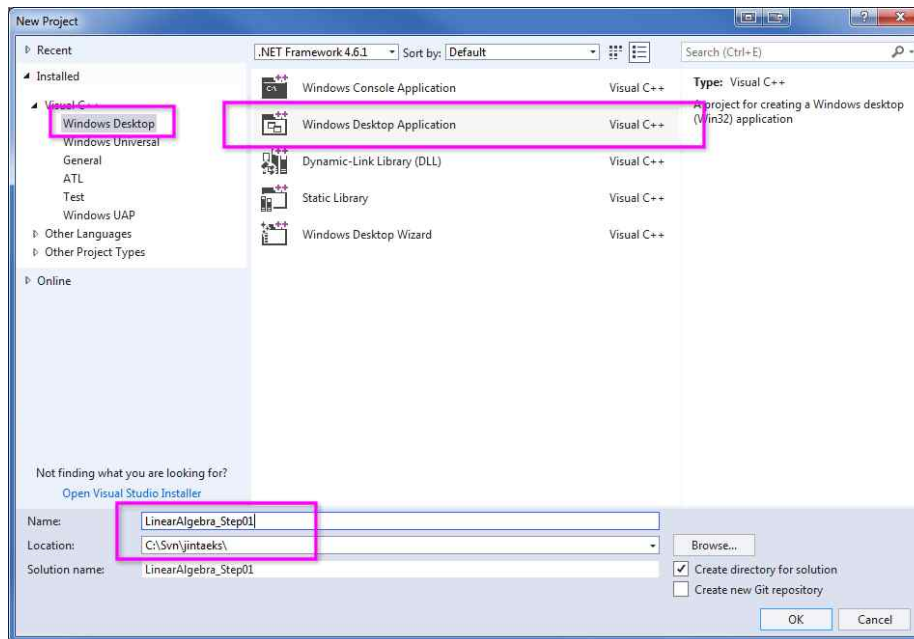
Windows GDI+

1. Generating Win32 Code

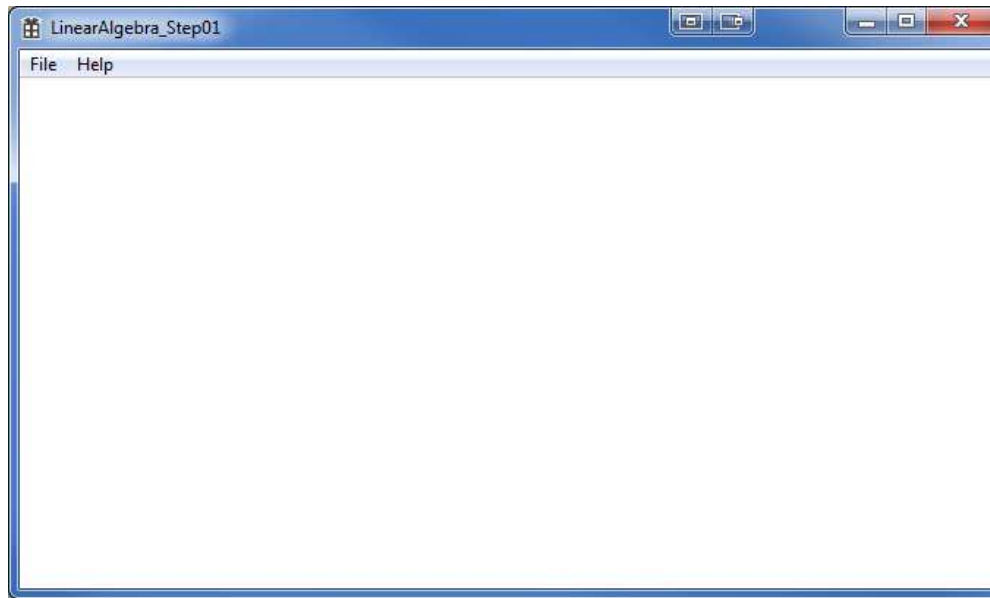
Code Generator in Visual Studio



[Fig] Creating Windows Desktop Application Project: Select [New-->Project...]

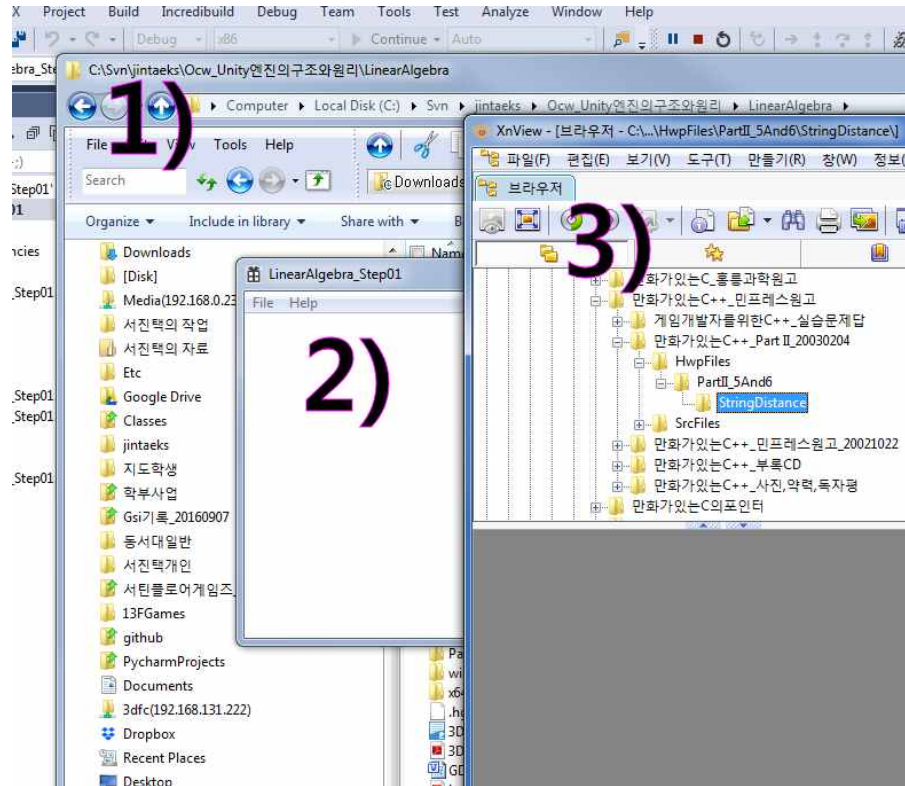


[Fig] Creating Windows Desktop Application: Select Windows Desktop under Visual C++ Tab. Select Windows Desktop Application in Application type list control.



[Fig] Running Application Program: Select [Debug --> Start Debugging]

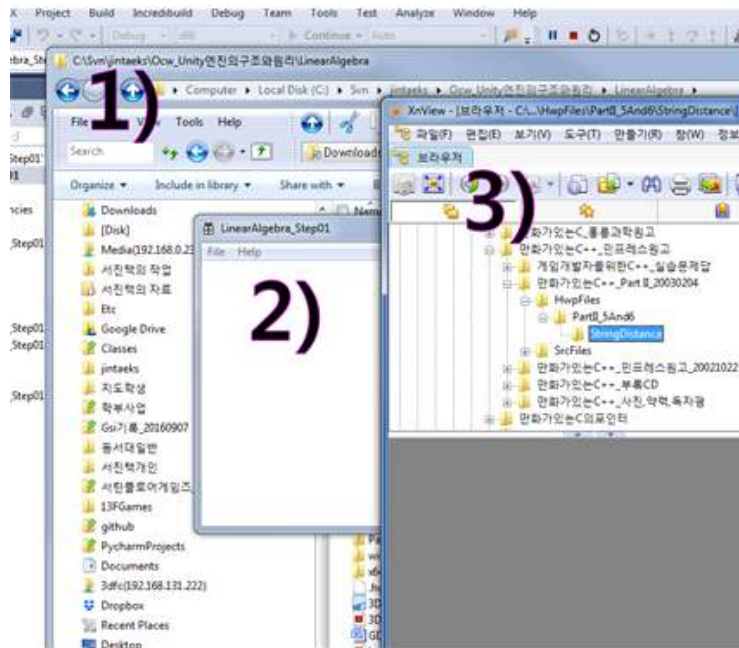
Draw something on Client Area



[Fig] Three Windows: OS manages structures to manage all windows.

Handle

Window Handle



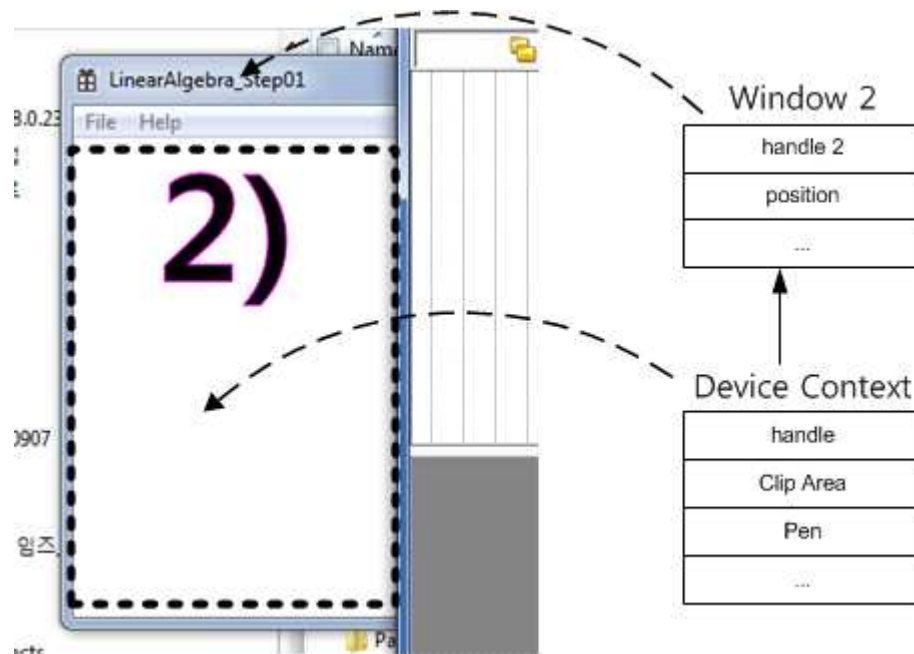
handle 1
position
...

handle 2
position
...

handle 3
position
...

[Fig] window handle

Unique ID which distinguishes windows



[Fig] Device Context Handle

DC, device context

2. Adding Drawing Code

```
void OnPaint(HDC hdc)
{
}
```

Handling WM_PAINT message

```
case WM_PAINT:
{
    PAINTSTRUCT ps;
    HDC hdc = BeginPaint(hWnd, &ps);
    // TODO: Add any drawing code that uses hdc here...
    // _20180519_jintaeks
    OnPaint( hdc );
    EndPaint(hWnd, &ps);
}
break;
```

Invalidate

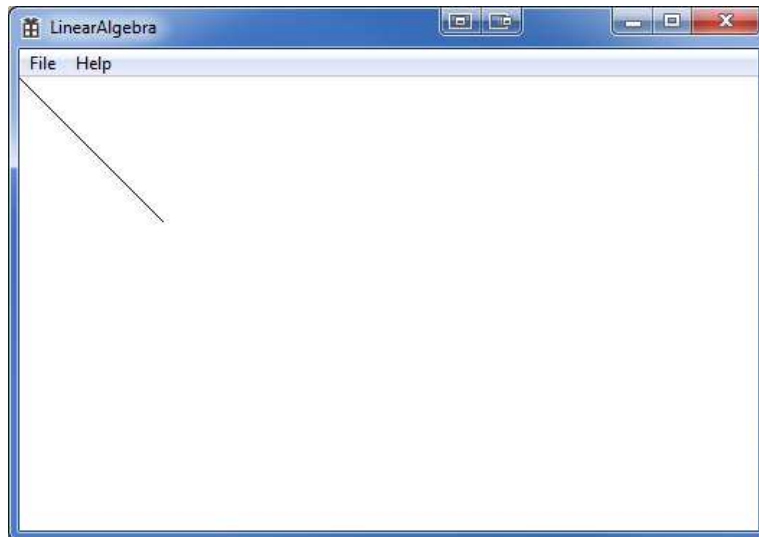
Screen Coordinate



[Fig] Screen Coordinate

CP

```
void OnPaint( HDC hdc )
{
    MoveToEx( hdc, 0, 0, NULL );
    LineTo( hdc, 100, 100 );
}
```



[Fig] Draw a line

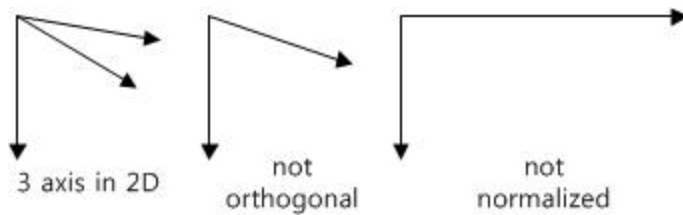
Moving CP

```
MoveToEx( hdc, 0, 0, NULL );
```

Draw Line

```
LineTo( hdc, 100, 100 );
```

Questions about Coordinate System

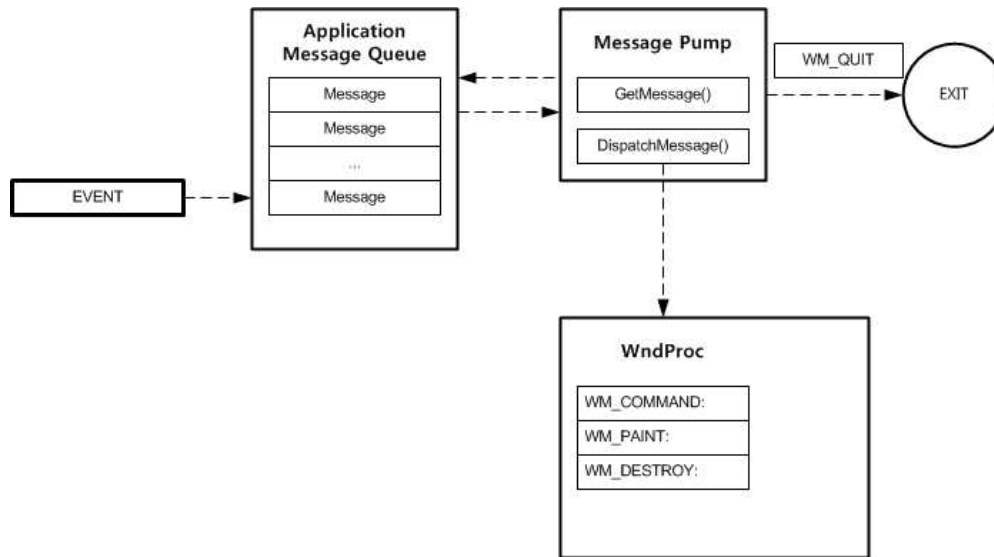


[Fig] Various Coordinates

3. Windows Programming

Windows 3.1

GetDC() vs. BeginPaint()



[Fig] Windows Message Pump

Message Loop

GetMessage() vs. PeekMessage()

WM_QUIT

Windows Procedure

CALLBACK

DispatchMessage()

```
while (GetMessage(&msg, nullptr, 0, 0))  
{  
    DispatchMessage(&msg);  
}
```

Special Message : WM_PAINT

Exercise

Write a code which draws function $f(x) = x^2$

Origins is located at center of a client area.

10 pixel treated as a 1 unit.

@