

2006년 2학기 윈도우 게임 프로그래밍

제2강 SDL의 기초

이대현

한국산업기술대학교



오늘의 학습 내용

- SDL의 기초 사용법
 - SDL의 소개
 - SDL 개발 환경 설정
 - SDL을 통한 이미지의 출력

SDL(Simple DirectMedia Layer)

■ SDL이란?

- 크로스 플랫폼 멀티미디어 라이브러리.
- 비디오, 오디오 및 사용자 입력을 처리하는 API로 구성.
- 기본적으로 2D 그래픽 라이브러리. 3D는 OpenGL을 통해서 지원.

■ SDL이 지원하는 플랫폼

- Windows, Linux, Windows CE, Mac OS, ...



■ 라이센싱

- GNU LGPL(Lesser General Public License)
- 자유롭게 상용 게임을 개발할 수 있음.
- 단, SDL과 게임프로그램은 동적으로 링킹되어야 함(Dynamic Linking).

■ 홈페이지

- www.libsdl.org

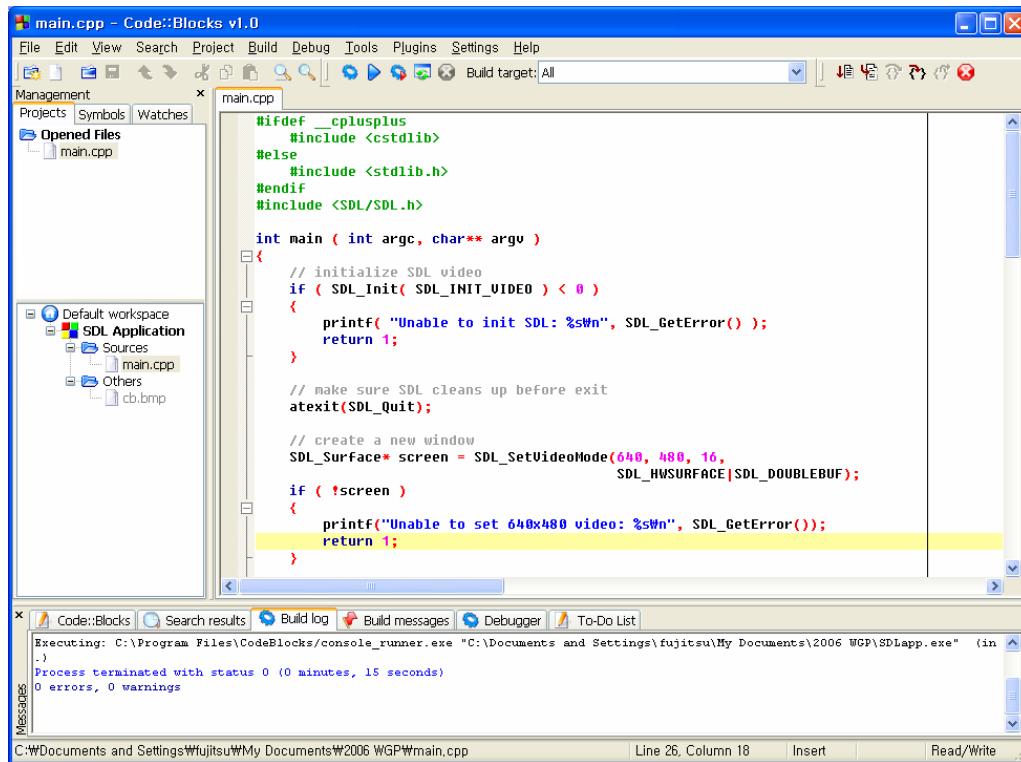
SDL 개발 환경

■ SDL 응용 프로그램 개발을 지원하는 개발 환경

- Visual C++
- Visual Studio.Net
- Eclipse
- Code::Blocks

Code::Blocks 통합 개발 환경(IDE)

- SDL 응용 프로그램을 손쉽게 개발할 수 있음.
- 컴파일러 및 통합 개발 환경 모두 100% 무료.



2006년 2학기 월도우 게임 프로그래밍

Copyright© by 이대현



- Step 1: SDL 라이브러리 초기화
- Step 2: 비디오 모드 설정 및 스크린 표면 생성
- Step 3: 이미지 로딩
- Step 4: 이미지 뿌리기
- Step 5: 화면 갱신
- Step 6: SDL 라이브러리 종료

2006년 2학기 월도우 게임 프로그래밍

Copyright© by 이대현



```
#include <SDL/SDL.h>

int main ( int argc, char** argv )
{
    SDL_Init(SDL_INIT_EVERYTHING);

    SDL_Surface* screen = SDL_SetVideoMode(480, 272, 16, SDL_SWSURFACE);

    SDL_Surface* bmp = SDL_LoadBMP("character.bmp");

    SDL_Rect dstrect;
    dstrect.x = 100;
    dstrect.y = 100;
    SDL_BlitSurface(bmp, NULL, screen, &dstrect);

    SDL_Flip(screen);

    SDL_Delay(5000);

    SDL_FreeSurface(bmp);
    SDL_FreeSurface(screen);
    SDL_Quit();

    return 0;
}
```

2006년 2학기 월도우 게임 프로그래밍

Copyright© by 이대현  신학기술학부
한국산업기술대학교

실행 결과



2006년 2학기 월도우 게임 프로그래밍

Copyright© by 이대현  신학기술학부
한국산업기술대학교

Step 1: SDL 라이브러리의 초기화

```
int SDL_Init(Uint32 flags);
```

- SDL 라이브러리의 각종 서브시스템들을 초기화
- 서브시스템: 타이머, 비디오, 오디오, CDROM, 조이스틱 등등

```
SDL_Init( SDL_INIT_EVERYTHING );
```

- 초기화할 서브시스템들을 지정
- `SDL_INIT_TIMER`, `SDL_INIT_AUDIO`, `SDL_INIT_VIDEO`,
`SDL_INIT_CDROM`, `SDL_INIT_JOYSTICK`, `SDL_INIT_EVERYTHING`
- 플래그를 여러 개 사용할 때는 비트 단위로 OR 연산자 사용
`SDL_INIT_TIMER | SDL_INIT_AUDIO`

Step 2: 비디오 모드 설정

```
SDL_Surface *  
SDL_SetVideoMode(int width, int height, in  
t bitsperpixel, Uint32 flags);
```

- SDL 표면을 생성.
- 표면(Surface): 2D 이미지 데이터가 메모리에 배열된 것.

```
SDL_Surface* screen =  
    SDL_SetVideoMode(640, 480, 16, SDL_SWSURFACE);
```

- 비디오 모드를 설정함.
- 리턴값은 생성된 표면임.
- 표면의 사이즈, 색상깊이(color depth), 및 메모리의 위치 등을 설정함.
- `SDL_SWSURFACE`: 표면을 시스템 메모리에 생성.
- `SDL_FULLSCREEN`: 화면을 풀스크린 모드로 생성.
- `SDL_DOUBLEBUF`: 하드웨어 더블버퍼링을 수행함.

Step 3: 이미지 로딩

```
SDL_Surface *SDL_LoadBMP(const char *file);
```

```
SDL_Surface* bmp = SDL_LoadBMP("character.bmp");
```



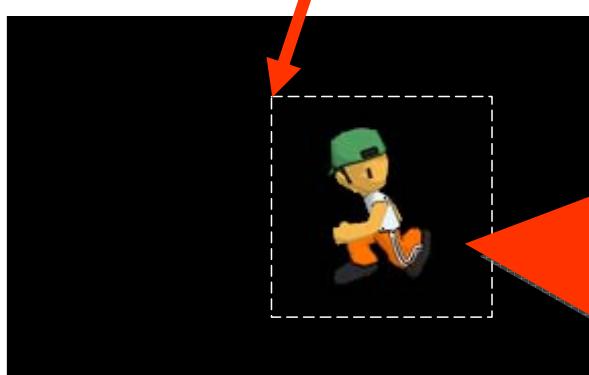
- BMP 파일을 로드하여, 표면으로 만든다.

Step 4: 이미지 뿌리기

```
int  
SDL_BlitSurface(SDL_Surface *src, SDL_Rect *srcrect,  
                 SDL_Surface *dst, SDL_Rect *dstrect);
```

목적 표면

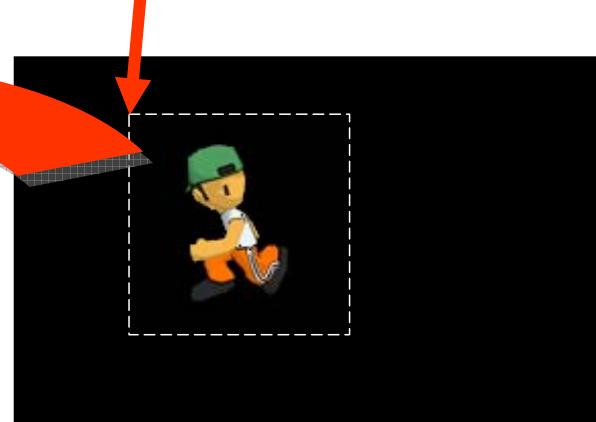
목적 표면위의 (x, y) 위치만을 지정.



.blit

원본 표면위의 (x, y) 위치와 (너비, 높이)를 지정.

원본 표면



- 목적 표면위의 (100,100) 위치에 블리팅.

```
SDL_Rect dstrect;
dstrect.x = 100;
dstrect.y = 100;

SDL_BlitSurface(bmp, NULL, screen, &dstrect);
```

- 'bmp' 표면에서 'screen' 표면으로 블리팅..

- 원본 표면 전체를 뜯함.

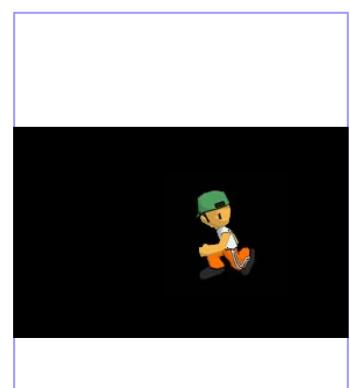
Step 5: 화면 갱신

```
int SDL_Flip(SDL_Surface *screen);
```

비디오 메모리



시스템 메모리



표면의 내용을 비디오 메모리로 옮겨 화면에 나타낸다.

```
SDL_Flip(screen);
```

Step 6: SDL 라이브러리 종료

```
void SDL_FreeSurface(SDL_Surface *surface);  
void SDL_Quit(void);
```

- 생성되었던 bmp와 screen 표면을 해제.

```
SDL_FreeSurface(bmp);  
SDL_FreeSurface(screen);  
SDL_Quit();
```

- SDL 라이브러리의 종료.

Tip: 실제 코드의 예러 처리

```
int main ( int argc, char** argv )  
{  
    if ( SDL_Init( SDL_INIT_VIDEO ) < 0 ) {  
        printf( "Unable to init SDL: %s\n", SDL_GetError() );  
        return 1;  
    }  
  
    SDL_Surface* screen = SDL_SetVideoMode(480, 272, 16, SDL_SWSURFACE);  
    if ( !screen ) {  
        printf("Unable to set video: %s\n", SDL_GetError());  
        return 1;  
    }  
  
    SDL_Surface* bmp = SDL_LoadBMP("character.bmp");  
    if (!bmp) {  
        printf("Unable to load bitmap: %s\n", SDL_GetError());  
        return 1;  
    }  
  
    ... 후략 ...
```

간단한 애니메이션

```
#include <SDL/SDL.h>

int main ( int argc, char** argv )
{
    SDL_Init(SDL_INIT_EVERYTHING);

    SDL_Surface* screen = SDL_SetVideoMode(480, 272, 16, SDL_SWSURFACE);

    SDL_Surface* bmp = SDL_LoadBMP("character.bmp");

    SDL_Rect dstrect;
    dstrect.y = 100;
    for ( int x = 0; x < 480; x++ ) {
        dstrect.x = x;
        SDL_BlitSurface(bmp, NULL, screen, &dstrect);
        SDL_Flip(screen);
        SDL_Delay(10);
    }

    SDL_FreeSurface(bmp);
    SDL_FreeSurface(screen);
    SDL_Quit();

    return 0;
}
```

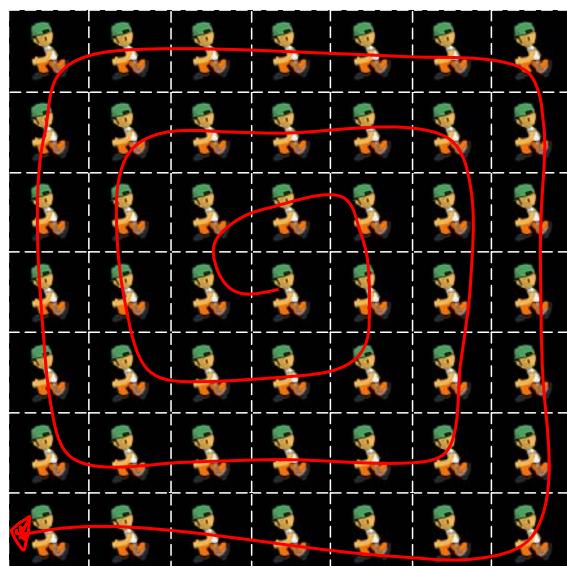
2006년 2학기 월도우 게임 프로그래밍

Copyright© by 이대현  신학기술학부 디자인학과
한국산업기술대학교

실습 과제

■ 캐릭터 소용돌이 애니메이션

- Full screen 모드로 모드 설정.
- 캐릭터 이미지를 화면 정중앙에 위치.
- 정중앙에서 시작하여, 시계방향으로 소용돌이쳐 나가면서 이미지를 움직임.
- 캐릭터의 너비와 높이는 surface->w와 surface->h 를 참조.



2006년 2학기 월도우 게임 프로그래밍

Copyright© by 이대현  신학기술학부 디자인학과
한국산업기술대학교