

프로젝트 개발 절차

허수진

sujin941220@gmail.com

01090539412

목차

개발 절차(OJT)

개발프로젝트 1. CodeEditor (코드편집기) -객체지향 진행절차 1

(진행중인 프로젝트)

1.UseCase작도

입력 처리 출력		Use Case Level		일수
입력	위치를 이동한다	1레벨	위치를 이동한다	5일
	범위를 지정한다		범위를 지정한다	5일
	붙여쓰다	2레벨	붙여쓰다	3일
	끼워쓰다		화면을 이동하다	3일
	바꾸다		자동줄바꿈	3일
처리	화면을 이동하다		끼워쓰다	3일
	자동줄바꿈		글자 지우기	3일
	새 메모장을 준비한다		블록 지우기	3일
	불러오다	3레벨	저장하다	1일
	폰트지정		불러오다	1일
	담아두다		새 메모을 준비한다	1일
	글자 지우기		자르다	1일
	블록 지우기		붙여넣다	1일
	복사하다		되돌리다	1일
	자르다		찾다	1일
	저장하다		바꾸다	1일
출력	붙여넣는다	4레벨	복사하다	1일
	되돌리다		담아두다	
	인쇄하다		폰트를 지정하다	
	찾다		인쇄하다	

개발프로젝트 1. CodeEditor (코드편집기) - 객체지향 진행절차 2

(진행중인 프로젝트)

1. 객체지향 프로그래밍

1.1 UseCase에서 클래스 다이어그램 작성

(필드, 레코드, 파일) 도출

Use Case ↵

붙여쓰다. ↵

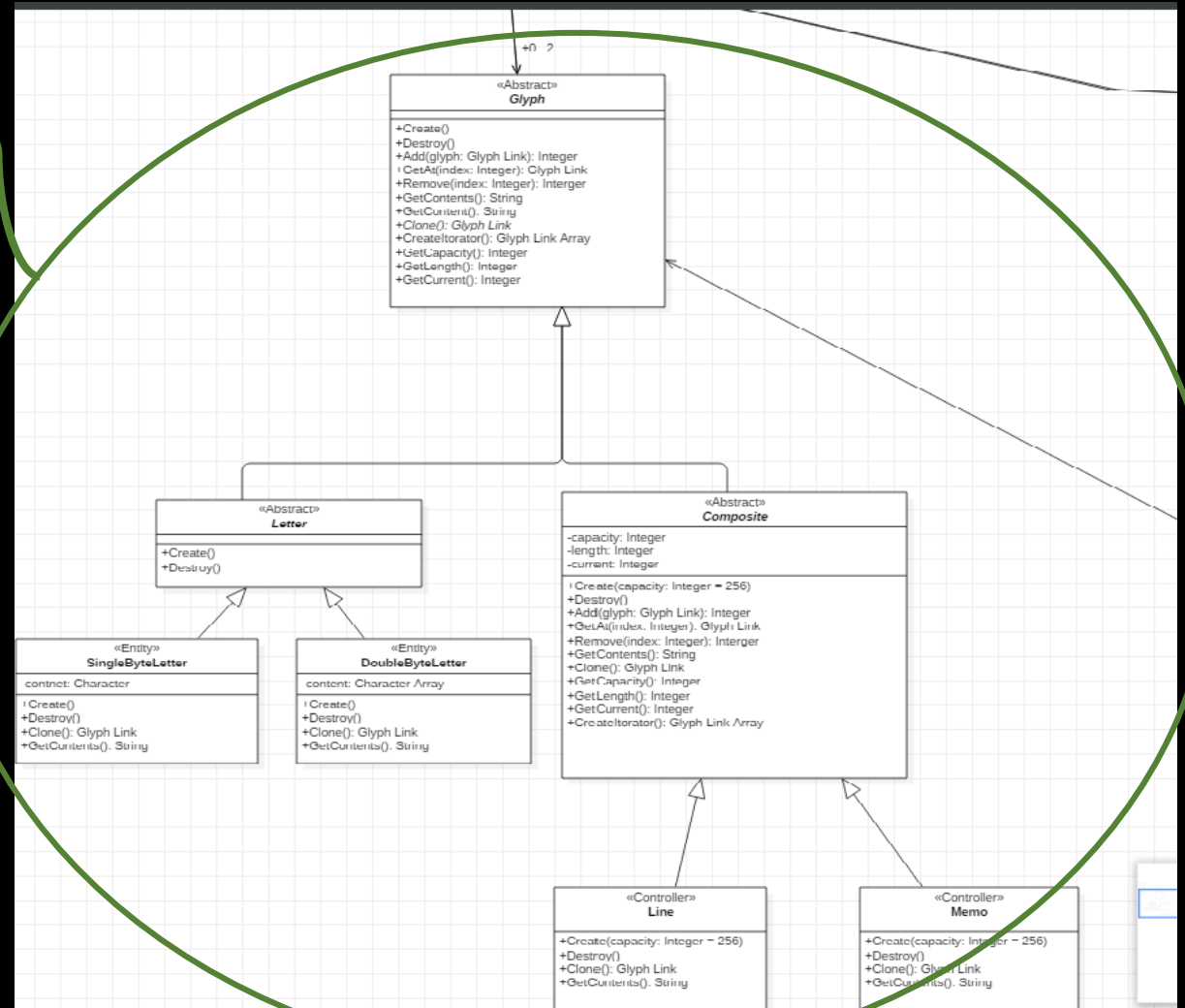
1. 글자의 크기를 정한다. ↵
2. 글자의 위치를 정한다. ↵
3. 글에 글자를 적는다. ↵

대체 ↵

1.1 글자를 적는다. ↵

분기 ↵

3.1 엔터키를 눌렀을 때. 새 줄을 만든다. ↵

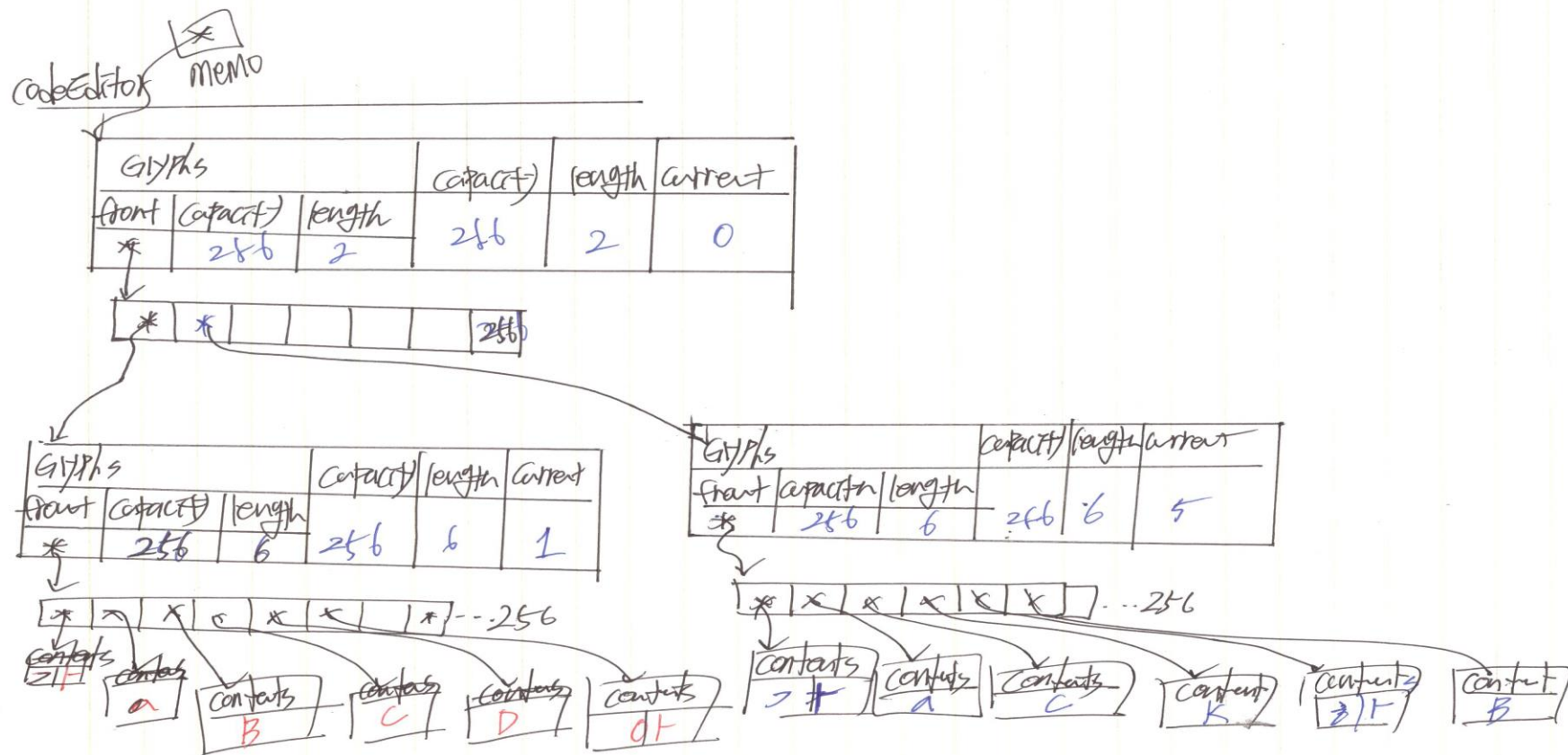


개발프로젝트 1. CodeEditor (코드편집기) -객체지향 진행절차 3

(진행중인 프로젝트)

1. 객체지향 프로그래밍

1.3 메모리맵 작도



CodeEditor에 글자를 입력 메모리 맵 작동

개발프로젝트 1. CodeEditor (코드편집기) -객체지향 진행절차 3

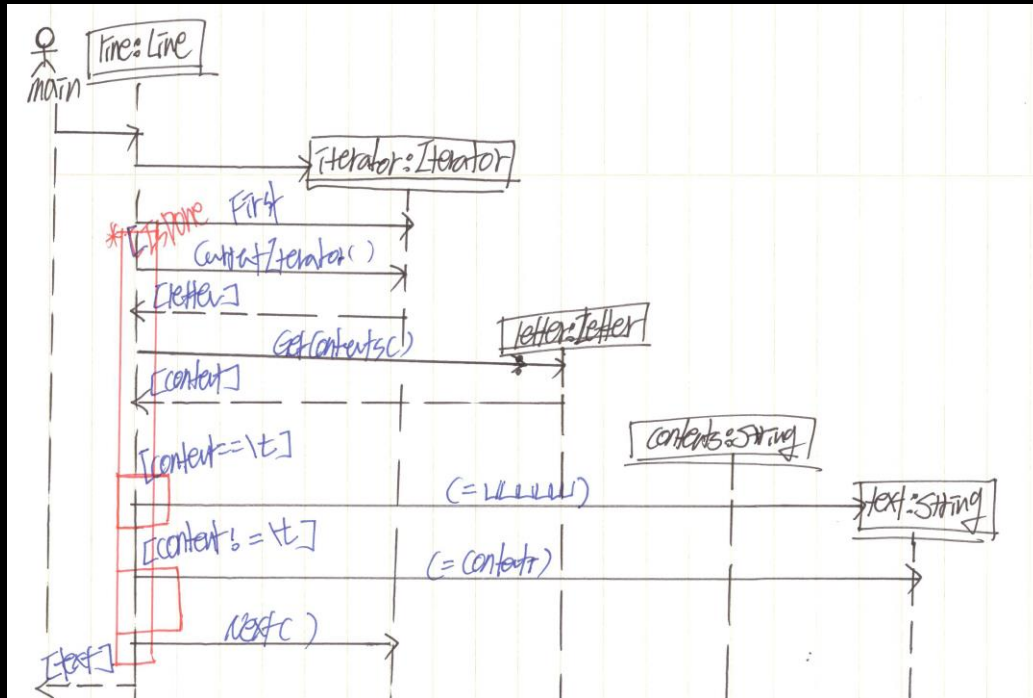
(진행중인 프로젝트)

1.객체지향 프로그래밍

1.4. 시퀀스 다이어그램작도 &(연산 상세설계) 코드작성

시퀀스 다이어그램

코드작성



```
string Line::GetContents() {
    string text = "";

    Iterator<Glyph*>*iterator = this->CreateIterator();

    for (iterator->First(); !iterator->IsDone(); iterator->Next()) {
        text += iterator->CurrentItem()->GetContents();
    }

    return text;
}
```

개발프로젝트 1. CodeEditor (코드편집기) -MFC 프로그래밍
진행절차 4

(진행중인 프로젝트)

2. MFC 프로그래밍

2.1 윈도우 LayOut 설계

메모장				X
파일	편집	찾기	도움말	
열기	되돌리기		버전	
저장	지우기		도움말	
다른이름으로 저장	전체 선택			
불러오기	복사하기			
인쇄하기	잘라내기			
닫기	붙여넣기			
	폰트설정			

개발프로젝트 1. CodeEditor (코드편집기) -MFC 프로그래밍

2. MFC 프로그래밍

진행절차 4

(진행중인 프로젝트)

2.2 사용자 시나리오 작성

- | | | |
|--|--|---|
| 1. 프레임이 생성될 때 <u>OnCreate</u>
1.1 메모를 만든다.
1.2 새 줄을 만든다.
1.3 메모에 새 줄을 추가한다. | 3. 한글 조립 중일 때 <u>OnIMEComposition</u>
3.1 문자를 읽는다.
3.2 <u>조립중</u> 이면 현재줄의 문자를 지운다.
3.3 읽은 문자를 현재 줄에 추가한다.
3.4작업영역을 갱신한다. | 5. 갱신했을 때 <u>OnPaint</u>
5.1메모의 줄의 <u>개수</u> 만큼 반복한다.
5.1.1메모의 칸의 개수만큼 반복한다.
5.1.1.1텍스트를 만든다.
5.1.2텍스트를 출력한다. |
| 2. 문자키를 입력했을 때 <u>OnChar</u>
2.1입력되어진 문자키를 읽는다.
2.2 <u>개행</u> 문자이면,
2.2.1새 줄을 만든다.
2.2.2메모에 새 줄을 추가한다
2.3제어 문자가 아니면,
2.3.1입력받은 문자를 새 줄에 <u>추가</u> 한다.
2.4 작업영역을 갱신한다. | 4. 한글 조립이 완성되었을 때 <u>OnIMEChar</u>
4.1 문자를 읽는다.
4.2 현재줄의 문자를 지운다.
4.3 읽은 문자를 현재 줄에 추가한다.
4.4 작업영역을 갱신한다. | 6.프레임을 닫을 때 <u>OnClose</u>
6.1메모를 지운다.
6.2윈도우 프레임을 <u>닫는다</u> . |

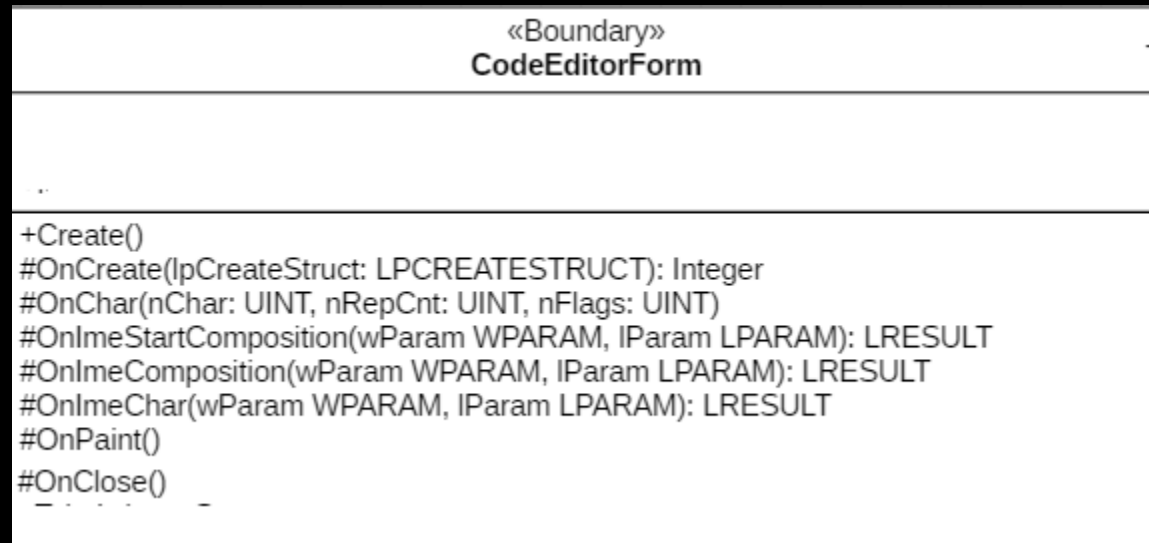
MFC와API의 연산을 사용
현 프로젝트에 맞게 클래스로 묶는다

개발프로젝트 1. CodeEditor (코드편집기) -MFC 프로그래밍 진행절차 4

(진행중인 프로젝트)

2. MFC 프로그래밍

2.3 클래스 다이어그램 작도



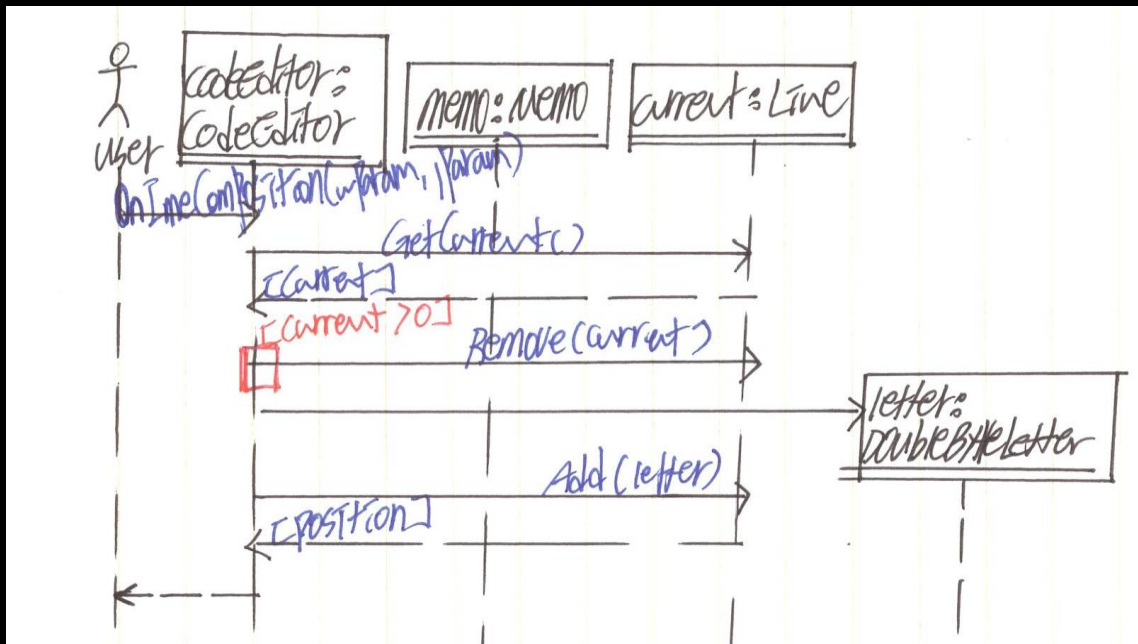
개발프로젝트 1. CodeEditor (코드편집기) -MFC 프로그래밍 진행절차 4

2. MFC 프로그래밍 (진행중인 프로젝트)

2.4 시퀀스 다이어그램 & 코드 작성 (필요한 연산 발생시 → 연산 상세설계)

코드 작성

시퀀스 다이어그램



```
HRESULT CodeEditorForm::OnlineComposition(WPARAM wParam, LPARAM lParam) {
    TCHAR str[2];
    Long current;

    if (lParam & GCS_COMPSTR) {
        str[0] = HIBYTE(LOWORD(wParam));
        str[1] = LOBYTE(LOWORD(wParam));

        current = this->current->GetCurrent();

        if (current >= 0 && this->isComposing == TRUE) {
            this->current->Remove(current);
        }

        Glyph *letter = new DoubleByteLetter(str);
        this->current->Add(letter);
        this->isComposing = TRUE;

        this->Notify();

        Invalidate();
    }
}
```

감사합니다.

허수진

sujin941220@gmail.com

01090539412