

# 제4장 계산기 프로그램

---

윈도우 프로그래밍 설계

청주대학교 전자공학과  
한철수

- CString 클래스
- 계산기 프로그램 작성 연습

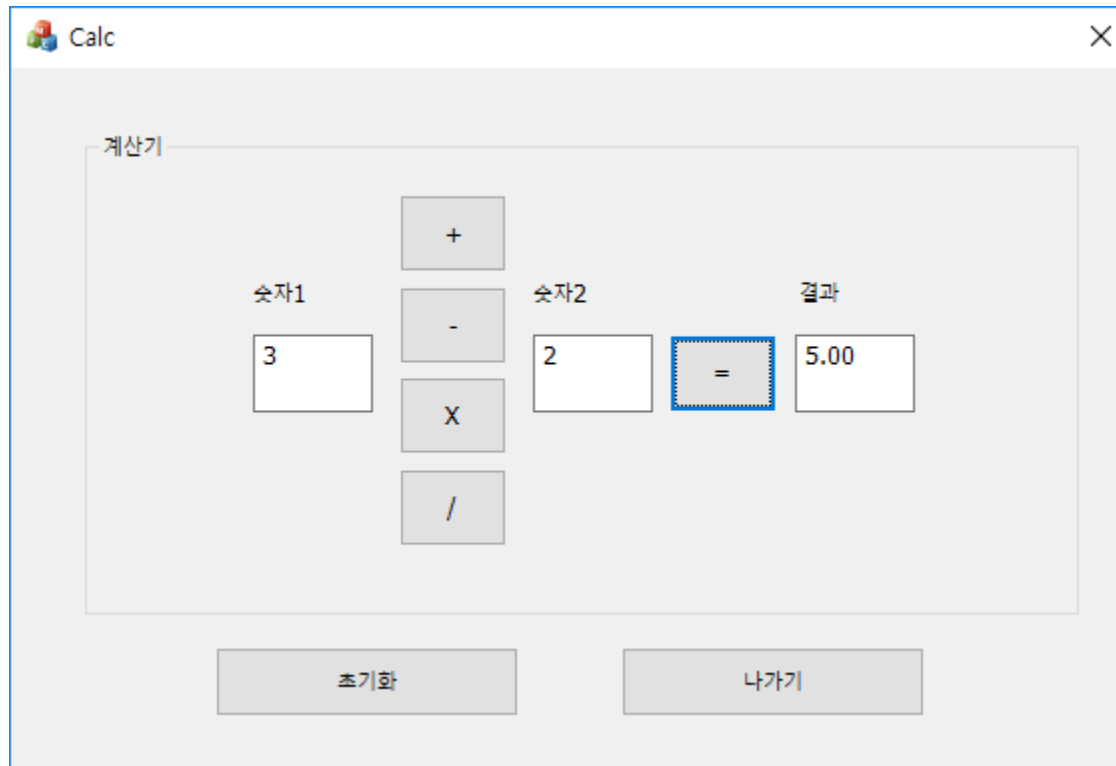
# CString 클래스

---

- CString 클래스는 문자열을 나타내는 클래스임.
- 문자열과 관련된 다양한 멤버 함수를 제공함.
  - 교재 90쪽 CString 클래스의 멤버 함수들을 다양하게 응용해 보길 바랍니다.
- 문자열-숫자 변환 함수
  - atoi() 함수
    - a(ascii) **to** i(int)
    - atoi("30");
      - int형 정수 30을 반환함.
    - \_wtoi()
      - 유니코드 문자열에 대해서 atoi()함수와 동일한 일을 수행함.
  - atol() 함수
    - a(ascii) **to** l(long int)
  - atof() 함수
    - a(ascii) **to** f(floating)
  - itoa() 함수
    - i(int) **to** a(ascii)

# 계산기 프로그램 작성 연습

- 만들고자 하는 프로그램
  1. 사칙 연산을 수행함.
  2. 초기화 버튼을 누르면 화면을 깨끗이 지움.

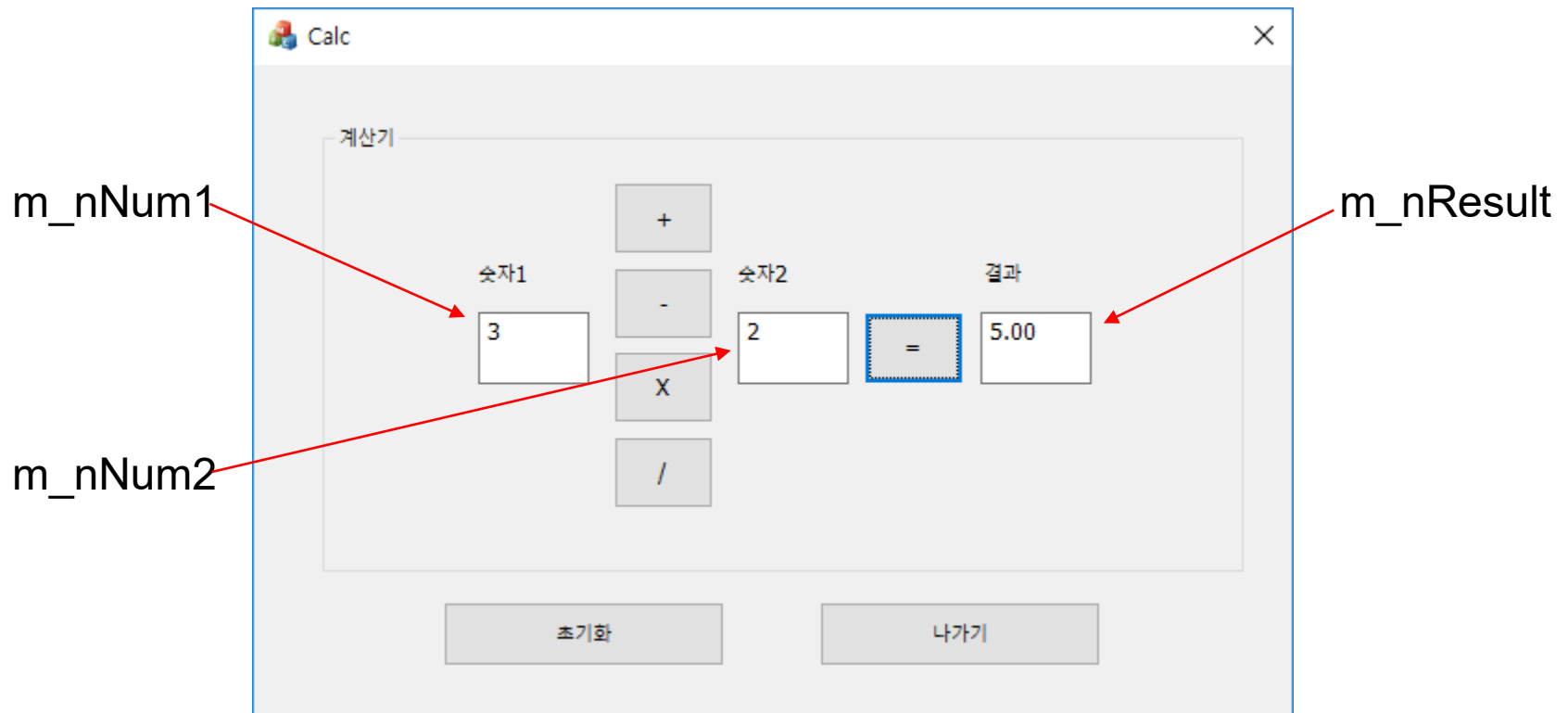


# 멤버 변수 선언

- CCalcDlg 클래스에 멤버 변수를 추가함.

```
int m_nOption;
```

- 에디트 컨트롤마다 CString형의 값 변수를 각각 연결함.



# + , - , X , / 버튼 처리

- + , - , X , / 버튼을 누르면 m\_nOption의 값을 변경함.

```
void CCalcDlg::OnClickedButtonAdd()  
{  
    // TODO: 여기에 컨트롤 알림 처리기 코드를 추가합니다.  
    m_nOption = 1;  
}
```

```
void CCalcDlg::OnClickedButtonSub()  
{  
    // TODO: 여기에 컨트롤 알림 처리기 코드를 추가합니다.  
    m_nOption = 2;  
}
```

```
void CCalcDlg::OnClickedButtonMul()  
{  
    // TODO: 여기에 컨트롤 알림 처리기 코드를 추가합니다.  
    m_nOption = 3;  
}
```

```
void CCalcDlg::OnClickedButtonDiv()  
{  
    // TODO: 여기에 컨트롤 알림 처리기 코드를 추가합니다.  
    m_nOption = 4;  
}
```

# = 버튼 처리

```

void CCalcDlg::OnClickedButtonEqu()
{
    // TODO: 여기에 컨트롤 알림 처리기 코드를 추가합니다.
    char temp[10];
    double tmpResult;

    UpdateData(TRUE);

    switch (m_nOption)
    {
    case 1:
        tmpResult = _wtof(m_nNum1) + _wtof(m_nNum2);
        break;
    case 2:
        tmpResult = _wtof(m_nNum1) - _wtof(m_nNum2);
        break;
    case 3:
        tmpResult = _wtof(m_nNum1) * _wtof(m_nNum2);
        break;
    case 4:
        tmpResult = _wtof(m_nNum1) / _wtof(m_nNum2);
        break;
    default:
        tmpResult = 0.00;
        break;
    }

    sprintf_s(temp, "%2.2f", tmpResult);
    m_nResult = temp;

    UpdateData(FALSE);
}

```

← 문자 배열을 선언함.

← 임시 변수를 하나 선언함.

← 사용자가 화면의 에디트 컨트롤에 입력하여 현재 화면상에 표시된 숫자를 에디트 컨트롤과 연결된 값 변수에 저장함.

← m\_nOption의 값에 따라 사칙 연산을 수행하고, tmpResult에 결과를 저장함. 유니코드 환경에서는 atof() 함수대신 \_wtof() 함수를 사용해야 함.

← tmpResult의 값을 서식을 지정하여 문자열로 바꾸어 배열 temp에 저장함.

← 값 변수에 저장된 값을 에디트 컨트롤로 보내서 현재 화면상에 표시된 숫자를 바꿈.

# 초기화 버튼 처리

```
void CCalcDlg::OnClickedButtonClear()
{
    // TODO: 여기에 컨트롤 알림 처리기 코드를 추가합니다.
    m_nNum1 = _T("");
    m_nNum2 = _T("");
    m_nResult = _T("");
    UpdateData(FALSE);
}
```

← 3개의 변수에 빈 문자열("")을 저장함.

← 변수의 값을 화면의 에디트 컨트롤로 보냄.



# 종료 처리

- 프로그램을 종료 시키기 위한 함수들
  - OnOK();
    - CDialog 클래스의 멤버 함수임.
      - 대화상자를 닫을 때에만 사용 가능.
  - PostQuitMessage(0);
    - 윈도우 API 함수임.
      - 전역함수임.
    - 함수 호출 시 인수에는 종료 코드를 적는다.
      - 정상적인 종료는 0을 사용함.

```
void CCalcDlg::OnClickedButtonExit()  
{  
    // TODO: 여기에 컨트롤 알림 처리기 코드를 추가합니다.  
    PostQuitMessage(0);  
}
```

# 질문

---

**Q & A**