
Replace, FindOneOf, Left, Mid 함수는 MFC의 CString 클래스에서 문자열을 처리하는 데 유용한 함수들이다.

1. Replace

- Replace 함수는 문자열 내의 특정 문자열을 다른 문자열로 대체하는 함수
- 대체된 문자열의 수를 반환하고, 대체되지 않은 경우 반환값은 0
- 대소문자를 구분하므로 대소문자를 무시하려면 대신 CString::ReplaceNoCase 함수를 사용

Replace 함수의 기본 구문

```
int Replace(LPCTSTR lpszOld, LPCTSTR lpszNew);
```

lpszOld는 원래 문자열에서 대체할 문자열을 나타내는 TCHAR 형식의 문자열,
lpszNew는 새 문자열을 나타내는 TCHAR 형식의 문자열이다.

간단한 예시 코드

```
CString strTest = _T("The quick brown fox jumps over the lazy dog");  
int nReplaceCount = strTest.Replace(_T("fox"), _T("cat"));  
Replace 함수를 사용하여 문자열 strTest 내의 "fox" 문자열을 "cat" 문자열로 대체
```

이 함수는 대체된 문자열의 수를 반환하므로 nReplaceCount 변수에 값을 할당, 이 코드는 "fox" 문자열이 한 번 발견되었으므로 nReplaceCount 변수에는 1이 저장된다.

ReplaceNoCase 함수의 기본 구문

```
int ReplaceNoCase(LPCTSTR lpszOld, LPCTSTR lpszNew);
```

간단한 예시 코드

```
CString strTest = _T("The quick brown fox jumps over the lazy dog");  
int nReplaceCount = strTest.ReplaceNoCase(_T("fox"), _T("cat"));  
ReplaceNoCase 함수를 사용하여 "fox" 문자열을 "cat" 문자열로 대체
```

2. FindOneOf()

2023/02/18

- 문자열 내에서 지정된 문자 집합 중 하나의 문자를 검색하는 함수
- 검색된 문자의 인덱스를 반환하며, 검색이 실패한 경우 -1을 반환
- 문자열 검색 및 추출에 유용

FindOneOf() 함수의 기본 구문

int FindOneOf(LPCTSTR lpszCharSet) const;

lpszCharSet : 검색할 문자 집합을 나타내는 TCHAR 형식의 문자열

예를 들어, 문자열 "ABC"를 검색하려는 경우 lpszCharSet 매개변수를 _T("ABC")로 설정할 수 있다

간단한 예시 코드

```
CString strTest = _T("Hello, world!");
int nIndex = strTest.FindOneOf(_T(",.!"));
if (nIndex != -1)
{
    CString strResult = strTest.Left(nIndex);
}
```

1. FindOneOf() 함수를 사용하여 "Hello, world!" 문자열에서 ",.!" 문자 집합 중 하나의 문자를 검색한
2. 해당 문자가 발견되면 인덱스 값을 반환하고, 그렇지 않으면 -1을 반환
3. 이 코드에서는 문자열에서 첫 번째 ",", ".", "!" 문자 중 하나를 찾으므로 nIndex 변수에 해당 문자의 인덱스가 할당

위의 코드에서는 인덱스 값 nIndex을 사용하여 "Hello" 문자열을 추출할 수 있고, 추출된 문자열은 Left() 함수를 사용하여 가져올 수 있다.

3. Left()

Left() 및 Mid() 함수는 문자열의 일부분을 추출하는 데 사용되는 함수

Left() 함수는 문자열의 시작부터 지정된 길이까지의 일부분을 추출한다.

2023/02/18

Left() 함수의 기본 구문

CString Left(int nCount) const;

위의 구문에서 nCount는 추출하려는 문자의 개수를 나타내는 int 형식의 정수입니다. Left() 함수는 추출된 일부분을 나타내는 CString 개체를 반환

간단한 예시 코드

```
CString strTest = _T("Hello, world!");  
CString strResult = strTest.Left(5);
```

문자열 "Hello, world!"에서 처음 5개의 문자열을 추출
추출된 문자열인 "Hello"는 strResult 변수에 저장

4. Mid()

: Mid() 함수는 문자열의 지정된 시작 위치부터 지정된 길이까지의 일부분을 추출한다

Mid() 함수의 기본 구문

CString Mid(int nFirst) const;

CString Mid(int nFirst, int nCount) const;

nFirst : 추출을 시작할 위치를 나타내는 int 형식의 정수.

nCount : 추출하려는 문자의 개수를 나타내는 int 형식의 정수

두 번째 형태의 Mid() 함수에서는 nFirst와 nCount 두 개의 인수가 필요하다

간단한 예시 코드

```
CString strTest = _T("Hello, world!");  
CString strResult1 = strTest.Mid(7);  
CString strResult2 = strTest.Mid(0, 5);
```

위의 코드에서는 문자열 "Hello, world!"에서 7번째 문자부터 끝까지 추출한다.

추출된 문자열 "world!"는 strResult1 변수에 저장 또한, 문자열에서 처음 5개의 문자를 추출
추출된 문자열 "Hello"는 strResult2 변수에 저장

- Left()와 Mid()의 차이점

Left() 함수는 문자열의 첫 부분을 추출하는 데 사용되고, Mid() 함수는 문자열의 중간 부분을 추출하는 데 사용된다.

예를 들어, 문자열 "Hello, world!"에서 Left() 함수를 사용하여 첫 다섯 개의 문자를 추출하면 "Hello"가 반환. 반면, Mid() 함수를 사용하여 첫 다섯 개의 문자를 추출하려면 Mid(0, 5)와 같이 사용해야 함

또한, Mid() 함수는 시작 위치와 추출할 문자의 개수를 모두 지정할 수 있기 때문에, Left() 함수보다 더 유연하다. 추출할 문자의 개수를 정확하게 지정할 수 있기 때문에, 문자열 중간부터 시작하는 경우 더욱 적합

따라서, 문자열에서 특정 위치나 길이에 해당하는 일부분을 추출하려면 Left() 함수 또는 Mid() 함수를 사용할 수 있고 선택하는 함수는 추출하려는 문자열의 위치와 길이에 따라 달라진다.