

1.5

ASCII

개요

컴퓨터는 텍스트를 비롯한 다양한 형태의 정보를 저장해야 합니다. 하지만 컴퓨터가 0과 1로 된 데이터만 저장할 수 있기 때문에, 0과 1을 이용해 텍스트 문자로 나타내야 합니다.

ASCII(아스키)는 문자를 컴퓨터가 이해할 수 있는 이진 데이터(0 또는 1)로, 혹은 그 반대로 변환하는 표준 방법입니다.

핵심개념

- * ASCII 코드
- * 유니코드

ASCII 인코딩 표준

65	←→	A
66	←→	B
67	←→	C
⋮		
89	←→	Y
90	←→	Z
<hr/>		
97	←→	a
98	←→	b
⋮		
121	←→	y
122	←→	z

ASCII(아스키)는 컴퓨터가 텍스트 데이터를 저장하기 위해 흔하게 사용하는 표준코드체계입니다. 이 표준에서 숫자 65는 대문자 'A'와 대응됩니다. 따라서 컴퓨터가 대문자 'A'를 저장하고 싶다면 숫자 65를 이진수로 저장할 것입니다(2진수로 나타내면, 1000001). 그 다음의 25개 값들은 다른 대문자 25개와 대응됩니다.

소문자도 ASCII에서 숫자로 나타낼 수 있습니다. 소문자 'a'는 숫자 97로 나타내고 'b'는 98로 나타내며 이 후 다른 소문자들도 마찬가지로 나타냅니다. 따라서 컴퓨터가 소문자 'a'를 저장하려면 숫자 97을 2진수, 1100001로 저장해야 합니다. 여기서 소문자 'a'와 대문자 'A'가 2^5 의 자릿수만 다르다는 것을 눈치 채셨나요? **ASCII에서 소문자는 같은 대문자 글자보다 항상 2^5 만큼 큼니다.** 결과적으로 이진수에서 2^5 의 자릿수만 바뀌면 되기 때문에(소문자는 1, 대문자는 0으로) 소문자와 대문자 간의 변환이 쉬워집니다.

ASCII의 한계

흔히 ASCII 코드는 ASCII 코드 표로 나타냅니다. ASCII 코드 표는 모든 ASCII 코드 문자와 그에 대응하는 숫자를 보여줍니다.

기본 ASCII 코드 표는 **7비트**만 이용해서 모든 문자들을 나타냅니다. 이것은 ASCII 코드로 2^7 개, 즉 128개의 문자를 나타낼 수 있다는 것을 의미합니다. **확장 ASCII**는 **8번째 비트**를 추가하여 총 256개의 문자를 나타낼 수 있도록 합니다.

소문자와 대문자 통틀어 52개 알파벳 밖에 없으므로, 그 외 남는 공간에 구분 기호, 숫자, 몇몇 기본 심볼들(\$나 % 기호 등) 같은 다른 종류의 문자들을 나타낼 수 있습니다.

하지만 8비트 ASCII 코드로도 나타낼 수 없는 문자들이 아직도 많이 있습니다. 우리가 사용할 수 있는 문자들의 개수는 256개보다 많기 때문이다. 예를 들어, 수학 기호들과 영어 외 다른 언어의 글자들은 표준 ASCII 표에 들어가기 힘듭니다. 이 때문에 훨씬 더 많은 문자들을 포함할 수 있는 **유니코드(Unicode)**이 생기게 되었습니다. 예를 들어, 유니코드는 100만개 이상의 문자들을 나타낼 수 있는 문자 인코딩 표준입니다. 유니코드의 첫 128개의 문자는 ASCII의 128개의 문자와 동일하므로 서로 호환이 됩니다.