

#define 매크로 상수

#define 은 상수(숫자, 기호, 문자열 등)나 심지어 함수까지 특정 문자로 치환할 수 있는 기능입니다.
 상수선언 함수형

#define 대체할문자 대체될상수

예를 들어 `printf()` 함수를 `YA` 라는 단어로 바꾸면 앞으로는 `YA` 를 사용해도 똑같은 기능이 발현됩니다.

```
#include <stdio.h>
#define YA printf

void main()
{
    YA("printf가 YA로 치환되었습니다.");
}
```

숫자를 예로 든다면, 자신이 작성하는 코드에서 숫자 `100` 이 자주 사용된다면, 숫자 `100` 을 특정 문자로 지정할 수 있습니다.

```
#include <stdio.h>
#define Hundred 100

int main()
{
    int x = 300;

    printf("%d \n", x + Hundred);
    printf("%d \n", x - Hundred);
    printf("%d \n", x * Hundred);
    printf("%d \n", x / Hundred);
}
```

이럴 경우 단순히 상수 100을 적는 것 보다 가독성이 증가합니다. 가장 중요한 점은 문자를 자유롭게 바꿀 수 있다는 점입니다.

`#define Hundred 100` 과 그냥 변수로 `int Hundered = 100;` 의 지정이 어떤 차이가 있는지 의문이 들 수 있습니다.

이것은 상수와 변수의 차이가 무엇이냐는 질문과 동일합니다.

`#define Hundred 100` 에서 `Hundred` 는 상수이므로 `Hundred = 200;` 과 같은 새로운 변수값을 삽입할 수 없습니다.

하지만 `int Hundred 100;` 에서 `Hundred` 는 변수이므로 `Hundred = 200;` 과 같은 새로운 변수값을 계속 삽입할 수 있습니다.

즉 `#define` 과 같은 상수값의 설정은 해당 값이 바뀌면 안되는 시점에 사용하는 목적이라 볼 수 있습니다.

