

## C library macro - assert()

### Description

The C library macro **void assert(int expression)** allows diagnostic information to be written to the standard error file. In other words, it can be used to add diagnostics in your C program.

### Declaration

Following is the declaration for assert() Macro.

```
void assert(int expression);
```

### Parameters

- **expression** – This can be a variable or any C expression. If **expression** evaluates to TRUE, assert() does nothing. If **expression** evaluates to FALSE, assert() displays an error message on **stderr** (standard error stream to display error messages and diagnostics) and aborts program execution.

### Return Value

This macro does not return any value.

### Example

The following example shows the usage of assert() macro –

```
#include <assert.h>
#include <stdio.h>
int main () {
    int a;
    char str[50];

    printf("Enter an integer value: ");
    scanf("%d", &a);
    assert(a >= 10);
    printf("Integer entered is %d\n", a);

    printf("Enter string: ");
    scanf("%s", str);
    assert(str != NULL);
    printf("String entered is: %s\n", str);
}
```

```
return(0);  
}
```

Let us compile and run the above program in the interactive mode as shown below –

```
Enter an integer value: 11  
Integer entered is 11  
Enter string: tutorialspoint  
String entered is: tutorialspoint
```