
The `[[noreturn]]` Attribute

The standard attribute `[[noreturn]]` indicates that the function to which it pertains does not return normally.

Description

The presence of the standard `[[noreturn]]` attribute as part of a function declaration informs both the compiler and human readers that such a function never returns control flow to the caller:

```
[[noreturn]] void f()
{
    throw 1;
}
```

The `[[noreturn]]` attribute is not part of a function’s type and is also, therefore, not part of the type of a function pointer. Applying `[[noreturn]]` to a function pointer is not an error, though doing so has no actual effect in standard C++; see *Potential Pitfalls — Misuse of `[[noreturn]]` on function pointers* on page 98. Using it on a pointer might have benefits for external tooling, code expressiveness, and future language evolution:

```
void (*fp [[noreturn]])() = f;
```

Use Cases

Better compiler diagnostics

Consider the task of creating an assertion handler that, when invoked, always aborts execution of the program after printing some useful information about the source of the assertion. Since this specific handler will never return because it unconditionally invokes a `[[noreturn]]std::abort` function, it is a viable candidate for `[[noreturn]]`:

```
[[noreturn]] void abortingAssertionHandler(const char* filename, int line)
{
    LOG_ERROR << "Assertion fired at " << filename << ':' << line;
    std::abort();
}
```