

By default, local variables have the following properties:

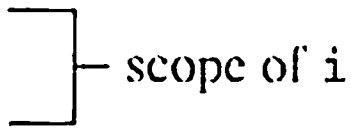
- **Automatic storage duration.** The *storage duration* (or *extent*) of a variable is the portion of program execution during which storage for the variable exists. Storage for a local variable is “automatically” allocated when the enclosing function is called and deallocated when the function returns, so the variable is said to have *automatic storage duration*. A local variable doesn’t retain its value when its enclosing function returns. When the function is called again, there’s no guarantee that the variable will still have its old value.
- **Block scope.** The *scope* of a variable is the portion of the program text in which the variable can be referenced. A local variable has *block scope*: it is visible from its point of declaration to the end of the enclosing function body. Since the scope of a local variable doesn’t extend beyond the function to which it belongs, other functions can use the same name for other purposes.

Section 18.2 covers these and other related concepts in more detail.

C99

Since C99 doesn’t require variable declarations to come at the beginning of a function, it’s possible for a local variable to have a very small scope. In the following example, the scope of `i` doesn’t begin until the line on which it’s declared, which could be near the end of the function body:

```
void f(void)
{
    ...
    int i;
    ...
}
```



Static Local Variables

Putting the word `static` in the declaration of a local variable causes it to have *static storage duration* instead of automatic storage duration. A variable with static storage duration has a permanent storage location, so it retains its value throughout the execution of the program. Consider the following function:

```
void f(void)
{
    static int i;    /* static local variable */
    ...
}
```

Since the local variable `i` has been declared `static`, it occupies the same memory location throughout the execution of the program. When `f` returns, `i` won’t lose its value.

Q&A

A static local variable still has block scope, so it’s not visible to other functions. In a nutshell, a static variable is a place to hide data from other functions but retain it for future calls of the same function.