Which style you use is mainly a matter of taste; there's no proof that one style is clearly better than the others. In any event, choosing the right style is less important than applying it consistently.

Q: If `i` is an `int` variable and `f` is a `float` variable, what is the type of the conditional expression `(i > 0 ? i : f)`?

A: When `int` and `float` values are mixed in a conditional expression, as they are here, the expression has type `float`. If `i > 0` is true, the value of the expression will be the value of `i` after conversion to `float` type.

Q: Why doesn't C99 have a better name for its Boolean type? [p. 85]

A: `_Bool` isn't a very elegant name, is it? More common names, such as `bool` or `boolean`, weren't chosen because existing C programs might already define these names, causing older code not to compile.

C99

Q: OK, so why wouldn't the name `_Bool` break older programs as well?

A: The C89 standard specifies that names beginning with an underscore followed by an uppercase letter are reserved for future use and should not be used by programmers.

***Q: The template given for the `switch` statement described it as the "most common form." Are there other forms? [p. 87]**

A: The `switch` statement is a bit more general than described in this chapter, although the description given here is general enough for virtually all programs.

labels ➤6.4

For example, a `switch` statement can contain labels that aren't preceded by the word `case`, which leads to an amusing (?) trap. Suppose that we accidentally misspell the word `default`:

```
switch (...) {
  ...
  defualt: ...
}
```

The compiler may not detect the error, since it assumes that `defualt` is an ordinary label.

Q: I've seen several methods of indenting the `switch` statement. Which way is best?

A: There are at least two common methods. One is to put the statements in each case *after* the case label:

```
switch (coin) {
  case 1:  printf("Cent");
           break;
  case 5:  printf("Nickel");
           break;
  case 10: printf("Dime");
           break;
```