

# Chapter 1

## Safe Features

---

**Modern C++ has a lot to offer.** Many of the modern language features provide affirmative value. They enable increased productivity and reliability, are easy to understand and use, and are hard to accidentally misuse in consequential ways. This chapter introduces those C++11 and C++14 features that have few, minor, and readily recognizable pitfalls, making misuse easy to avoid. Moreover, these features bring little systemic risk when introduced widely into a predominantly C++03 codebase, allowing organized training to be optional rather than mandatory. An organization’s leadership can feel reasonably comfortable guiding all its engineers toward using features presented in this chapter.

*Safe* features are characterized primarily by being of low risk. Recall from “A *Safe* Feature” in Chapter 0 that the **override** feature (p. 104) was singled out as epitomizing a *safe* feature. Though applicable only in the context of inheritance and virtual functions, this feature is almost impossible to misuse such that it fails to add value. Another example of a feature having low risk but with exceptionally high reward is **static\_assert** (p. 115). This particular feature is so universally useful and immune to inadvertent misuse that we use it liberally throughout the book to illustrate important compile-time properties almost as if it were a C++03 feature. Although not all features presented in this chapter are as eminently useful or widely applicable as these two, all of them are usable with minimal risk and thus are considered *safe*.

In short, widespread adoption of *safe* features is a low-risk proposition. All of these features are easy to understand and use profitably and hard to misuse; hence, formal training is generally not required. An organization need not be concerned about incorporating *safe* features into a predominantly C++03 codebase maintained primarily by those largely unfamiliar with modern features. If you’re new to the features of modern C++, by all means start here.