Wrapping Up

Kotlin's support for object-oriented programming goes beyond what we're used to in Java. Kotlin generates fields and synthesizes getters and setters so you don't have to spend the time writing boilerplate code. Instance members go into classes and class members go into companion objects. Singleton is a first-class citizen in the language. Data classes are useful to represent data more than behavior, and you can enjoy the destructuring capability to extract properties fluently from data classes. The support for generic classes is almost the same as in Java, except for improved type safety that comes from better support for variance and constraints.

We've seen how to create classes in this chapter, but classes rarely live alone. In the next chapter, we'll work with multiple classes to create class hierarchies using inner classes and inheritance.