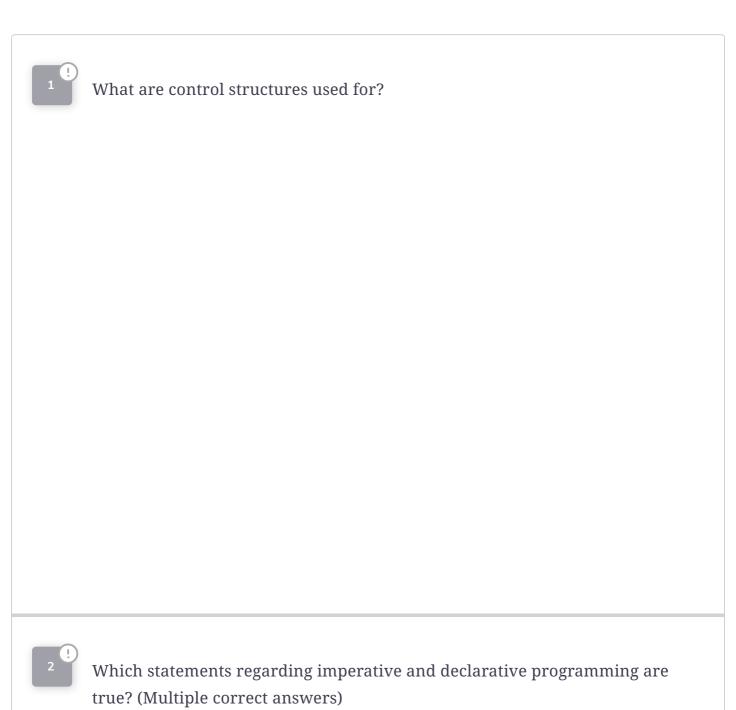
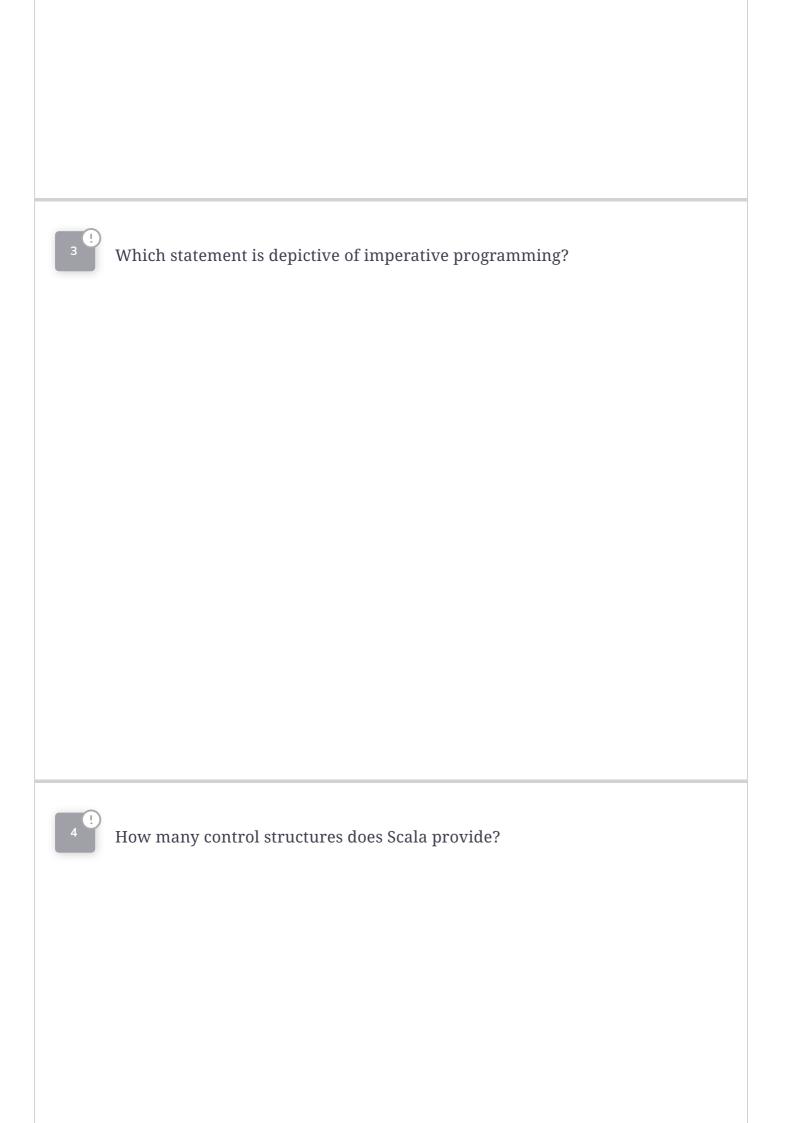
Chapter Quiz

In the following quiz, you will be tested on concepts you learned in this chapter.







How does Scala bring a functional approach to imperative control structures?

6

What will be the output of the following code:

```
val array = Array(1,2,3,4,5)

for (i <- 0 until array.length) {
   if (array(i) % 2 == 0) {
      println(array(i))
   }
}</pre>
```



What will be the output of the following code:

```
val constantPattern: Any = List()

constantPattern match {
   case 5 => print("five")
   case true => print("truth")
   case "hello" => print("hi!")
   case Nil => print("the empty list")
   case _ => print("something else")
}
```

```
val sequencePattern = Array(1,2,0)

sequencePattern match {
   case Array(0,_,_) => println("case1")
   case Array(1,_,_) => println("case2")
   case Array(_,_,0) => println("case3")
   case _ => println("default")
}
```

9 !

while is an expression.

Retake Quiz

In the next chapter, we will jump to functions and start seeing Scala's actual power.