Scala vs. Java from A. Sundararajan's Weblog (https://blogs.oracle.com/sundararajan/entry/scala_for_java_programmers)

Feature	**Java**	**Scala**
Static typing	Yes	Yes
Object oriented programming	Yes	Yes
Functional programming	No	Yes
Variable declaration	// type var_name = init_value; int i = 0;	// var var_name: type = init_value; var i:int = 0;
Constant declaration	// final type var_name = init_value; final int i = 0;	// val var_name: type = init_value; val i:int = 0;

Class declaration	class Person {
	  // members here
	}
		class Person {
		  // members here
		}

Methods	// RetType name(PType1 pName1, PType2 pName2)
	class Person {
	  public String getName () {
	  // code here...
	  }
	  public void setAge(int age) {
	  // code here...
	  }
	}
		// def name(pName1: pType1, pName2: pType2)
		class Person {
		  def getName(): String {
		  // code here...
		  }
		  def setAge(age: Int): Unit {
		  // code here...
		  }
		}

Operator overloading	No	class Complex {
		  def + (other: Complex): Complex {
		  // code here
		  }
		}

Static fields and methods	class Person {
	  private static Person president = …
	  private static Person getPresident () {
	  return president;
	  }
	}
		No static members.

Singletons	No language support. Can be simulated with a private constructor.	object President {
		  code here...
		}

Interfaces	interface Runnable {
	  void run();
	}
		trait Runnable {
		  def run: Unit;
		}