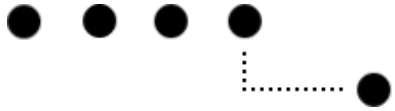

Bern University of Applied Sciences
Engineering and Information Technology

Scala

Stefan Heinemann

08. November 2011

Inhalt

Einleitung

- objektorientiert und funktional
- statisch typisiert
- Bytecode $\rightarrow$ JVM
- Integration mit Java
- Kompilierbar
- Scriptsprache

Einleitung

- objektorientiert und funktional
- statisch typisiert
- Bytecode → JVM
- Integration mit Java
- Kompilierbar
- Scriptsprache

Einleitung

- objektorientiert und funktional
- statisch typisiert
- Bytecode $\rightarrow$ JVM
- Integration mit Java
- Kompilierbar
- Scriptsprache

Einleitung

- objektorientiert und funktional
- statisch typisiert
- Bytecode $\rightarrow$ JVM
- Integration mit Java
- Kompilierbar
- Scriptsprache

Einleitung

- objektorientiert und funktional
- statisch typisiert
- Bytecode $\rightarrow$ JVM
- Integration mit Java
- Kompilierbar
- Scriptsprache

Einleitung

- objektorientiert und funktional
- statisch typisiert
- Bytecode $\rightarrow$ JVM
- Integration mit Java
- Kompilierbar
- Scriptsprache

Hallo Welt

```
1 object HelloWorld {
2   def main(args: Array[String]) {
3     println("Hello World!")
4   }
5 }
```

Klassen

```
1  class foo(var x:Int, y:Int) extends bar(y)  {
2    if (y > 0) {
3      throw new Exception("Invalid Argument");
4    }
5
6    var member:String = "Ich bin drin!"
7
8    override def toString(): String =
9      "foo says hello"
10
11   private def doSomething() {
12     println("something stupid?")
13   }
14 }
```

Traits

- Keine Interfaces
- Keine Mehrfachvererbung

Traits

- Keine Interfaces
- Keine Mehrfachvererbung

Traits

- Ähnlich wie Interfaces
- Können aber Implementationen enthalten
- Können mehrfach an Klassen weitervererbt werden
- Weniger problematisch als Mehrfachvererbung

Traits

- Ähnlich wie Interfaces
- Können aber Implementationen enthalten
- Können mehrfach an Klassen weitervererbt werden
- Weniger problematisch als Mehrfachvererbung

Traits

- Ähnlich wie Interfaces
- Können aber Implementationen enthalten
- Können mehrfach an Klassen weitervererbt werden
- Weniger problematisch als Mehrfachvererbung

Traits

- Ähnlich wie Interfaces
- Können aber Implementationen enthalten
- Können mehrfach an Klassen weitervererbt werden
- Weniger problematisch als Mehrfachvererbung

Funktionsobjekte

- Funktionen sind auch Objekte
- Speichern in Variablen
- Übergeben an Funktionen
- Rückgabewerte
- Callbacks
- Anonym möglich

Funktionsobjekte

- Funktionen sind auch Objekte
- Speichern in Variablen
- Übergeben an Funktionen
- Rückgabewerte
- Callbacks
- Anonym möglich

Funktionsobjekte

- Funktionen sind auch Objekte
- Speichern in Variablen
- Übergeben an Funktionen
- Rückgabewerte
- Callbacks
- Anonym möglich

Funktionsobjekte

- Funktionen sind auch Objekte
- Speichern in Variablen
- Übergeben an Funktionen
- Rückgabewerte
- Callbacks
- Anonym möglich

Funktionsobjekte

- Funktionen sind auch Objekte
- Speichern in Variablen
- Übergeben an Funktionen
- Rückgabewerte
- Callbacks
- Anonym möglich

Funktionsobjekte

- Funktionen sind auch Objekte
- Speichern in Variablen
- Übergeben an Funktionen
- Rückgabewerte
- Callbacks
- Anonym möglich

Pattern Matching

```
1 def describe (x: Any) = x match {
2 case 5         => "five"
3 case true      => "truth"
4 case "hello"   => "hi!"
5 case Nil       => "the empty list"
6 case _         => "something else" // Default
7 }
```

Typinferenz

- Typ kann explizit angegeben werden
- Typ kann manchmal vom Kontext her abgeleitet werden
- Kürzerer Code

Typinferenz

- Typ kann explizit angegeben werden
- Typ kann manchmal vom Kontext her abgeleitet werden
- Kürzerer Code

Typinferenz

- Typ kann explizit angegeben werden
- Typ kann manchmal vom Kontext her abgeleitet werden
- Kürzerer Code

Typinferenz

```
1  var x = "hallo"
```

Fragen

Fragen?