

# PEP Scala (1)

Email: christian.urban at kcl.ac.uk  
Office: N7.07 (North Wing Bush House)  
Slides & Code: KEATS

# Why Scala?

twitter 

LinkedIn 

Morgan Stanley

CREDIT SUISSE 

...



edf  
ENERGY

Novell.

foursquare™

HSBC 

...

# Why Scala?

- compiles to the JVM  
(also JavaScript, native X86 in the works)
- integrates seamlessly with Java
- combines **functional** and **object-oriented** programming
- it is a bit on the “mathematical” side  
(no pointers, no null)
- often one can write very concise and elegant code

alternatives: Elm, Haskell, Ocaml, ML, Lisp (Racket), ...

# Java vs Scala

## Java

```
public class Point {  
    private final int x, y;  
  
    public Point(int x, int y) {  
        this.x = x;  
        this.y = y;  
    }  
  
    public int x() { return x; }  
  
    public int y() { return y; }  
}
```

---

```
class Point(val x: Int, val y: Int)
```

## Scala

# Scala Tools

- there is a plugin for Eclipse (called Scala IDE)
- there is also a plugin for IntelliJ
- I use the venerable Emacs ;o)

# Types

- Base types

Int, Long, BigInt, Float, Double  
String, Char  
Boolean

- Compound types

List[Int]

lists of Int's

Set[Double]

sets of Double's

(Int, String)

Int-String pair

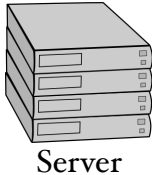
List[(BigInt, String)]

lists of BigInt-String  
pairs

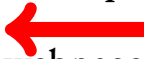
List[List[Int]]

list of lists of Int's

# An Http Request



GET request



webpage



Browser



POST data



```
import java.io.IOException;
import java.net.MalformedURLException;
import java.net.URL;
import java.util.Scanner;

public class URLReader {

    public static String readURL(String sUrl) {
        StringBuilder buf = new StringBuilder();
        Scanner in = null;

        try {
            URL url = new URL(sUrl);
            in = new Scanner(url.openStream());

            while (in.hasNextLine()) {
                buf.append(in.nextLine() + "\n");
            }
            return buf.toString();

        } catch (MalformedURLException e) {
            System.err.println(e);
        } catch (IOException e) {
            System.err.println(e);
        } finally {
            if (in != null) {
                in.close();
            }
        }
        return null;
    }
}
```



```
import java.io.IOException;
import java.net.MalformedURLException;
import java.net.URL;
import java.util.Scanner;

public class URLReader {

    public static String readURL(String sUrl) {
        StringBuilder buf = new StringBuilder();
        Scanner in = null;

        try {
            URL url = new URL(sUrl);
            in = new Scanner(url.openStream());

            while (in.hasNextLine()) {
                buf.append(in.nextLine() + "\n");
            }
            return buf.toString();

        } catch (MalformedURLException e) {
            System.err.println(e);
        } catch (IOException e) {
            System.err.println(e);
        } finally {
            if (in != null) {
                in.close();
            }
        }
        return null;
    }
}
```

# Conclusion

- Scala is still under heavy development (the compiler is terribly slow)
- <http://www.scala-lang.org/>
- it is a rather **deep** language...i.e. gives you a lot of rope to shoot yourself
- hope you have fun with the coursework

**Questions?**