PEP Scala (1)

Email:	christian.urban at kcl.ac.uk
Office:	S1.27 (1st floor Strand Building)
Slides & Code:	KEATS

Why Scala?


twitter
LinkedIn
theguardian
Morgan Stanley
CREDIT SUISSE

...


EDF
ENERGY
Novell
foursquare
HSBC

...

Why Scala?

- compiles to the JVM (also JavaScript, X86)
- integrates seamlessly with Java
- combines **functional** and **object-oriented** programming
- allows often to write more elegant code

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; }
}
```

Scala

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

Types

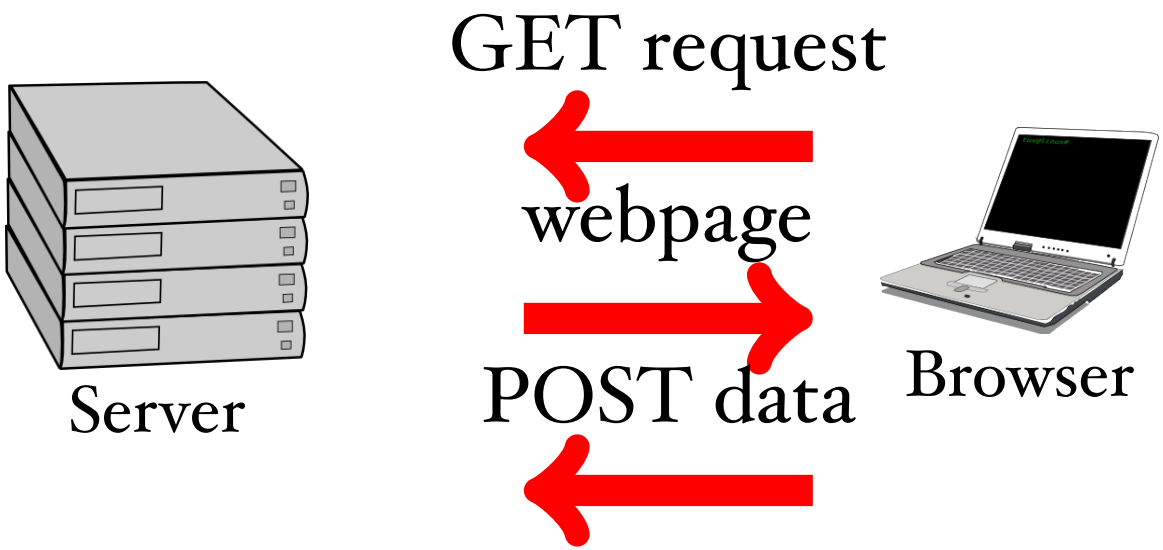
- Base types

  `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



```
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;
    }
}
```