Example 1

- Create a simple application server
  - applications are accessible via HTTP
    - as a basis, use the HTTP server from the previous practicals
  - applications are represented by classes
  - at start, the server receives a list of classes via the parameters of the command-line
  - the server can process URLs formatted as follows:
    \[ http://host/class_name/method_name?param1=value&param2=value \]
  - class_name is the name of a loaded class
  - method_name is a method name
    - any method of the particular class can be used
  - number of parameters of URL has to correspond with the number of parameters of the method (consider the String type only)
  - the method returns String that is returned as a result of the URL call
Assignment 1

• continuing
  – methods can be annotated with the annotation
    @Return(mime="mime/type"), which defines the mime
type of the answer (default is text/plain)
  – example of a class

```java
class App {
    public static String plainText(String param);
    @Return(mime="text/html")
    public static String html(String param1, String param2);
}
```