

The Assignment Environment is an automated testing environment

- The tasks are always in the same form
 - You retrieve the task by making a GET-request on the url for the given stage
 - You send your solution via POST-request to the same url which you got the task from
 - If you try to retrieve a task which does not exist you will produce a status **404 RESOURCE_NOT_FOUND**
- After sending your solution via POST-request you will get following result
 - **On correct solution**, status: **202 ACCEPTED**, message: Accepted, <link to the next task>
 - **On wrong solution**, status: **417 EXPECTATION_FAILED**, message: Expectation Failed
 - **On malformed solution**, status: **400 BAD_REQUEST**, message: <proper error message>
 - **On processing error**, status: **500 SERVER_ERROR** message: <proper error message>
- For successfully completing the assignment your software has to solve all test cases automatically without user interaction and errors during the test cases
 - Therefore the testing environment provides the possibility to reset the state
 - The test cases are randomly generated, the seed changes automatically from time to time

Url: <http://localhost:8080/assignment/stage/4/testcase/<testcase>>

Task:

- The left side of the equation still consists of two digits and an operator (+ or -)
- As in stage and two one you are allowed to move **up to** one segment within a **single digit** in the equation
- As an alternative you are allowed to move one segment from one digit to another
- You are now always allowed to flip the operator from + to - and - to +
- You should submit an array containing each possible correct solution
- If there is no correct solution you should submit an empty array

Assignment - Equations - Specification - Stage 4

Communication Sample:

Communication Sample:

Url: <http://localhost:8080/assignment/stage/4/testcase/<testcase>>

GET-request:

```
{  
  "equation": "1-1=2"  
}
```

POST-request:

```
{  
  "correctedEquations"  
    : ["1+1=2"]  
}
```

Result:

```
{  
  "message": "  
    Accepted",  
  "linkToNextTask": "  
    http://  
    localhost:8080/  
    assignment/  
    stage/4/  
    testcase/2"  
}
```

Assignment - Equations - Specification - Stage 4

Communication Sample:

Communication Sample:

Url: <http://localhost:8080/assignment/stage/4/testcase/<testcase>>

GET-request:

```
{  
  "equation": "1-1=8"  
}
```

POST-request:

```
{  
  "correctedEquations"  
    : ["7-1=6"]  
}
```

Result:

```
{  
  "message": "  
    Accepted",  
  "linkToNextTask": "  
    http://  
    localhost:8080/  
    assignment/  
    stage/4/  
    testcase/3"  
}
```