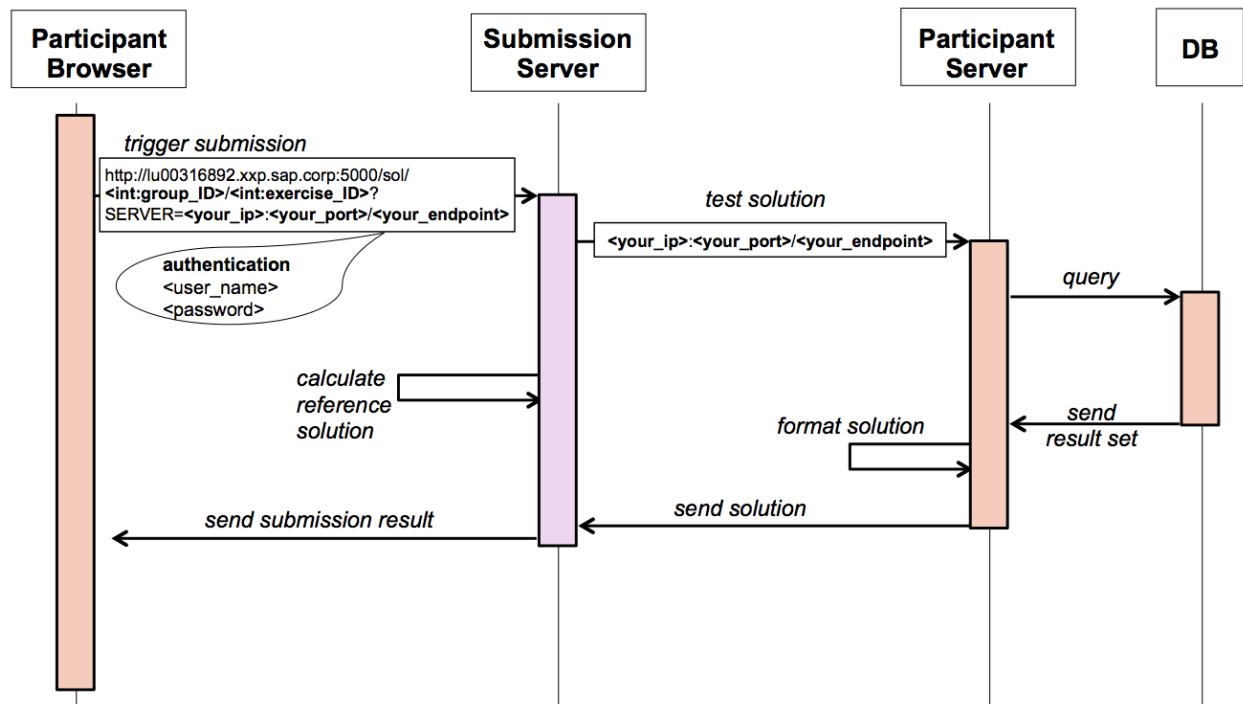


## Challenge 2: Submission Server

In the challenge you have to develop a HTTP participant server that is able to work with HANA. To submit the solutions of the tasks in this challenge your server has to send the results to the submission server.



The submission server has an endpoint:

`http://lu00316892.xxp.sap.corp:5000/sol/<int:group_ID>/<int:exercise_ID>`

Send a request to connect. You need to authenticate yourself with your `<user_name>` and your `<password>` using HTTP basic authentication.

Please append a GET parameter named SERVER with your own endpoint so the submission server can connect your participant server. The template looks like this:

`http://lu00316892.xxp.sap.corp:5000/sol/<int:group_ID>/<int:exercise_ID>?SERVER=< your_ip >:<your_port>/<your_endpoint>`

We recommend that your endpoint's name contains the exercise number so you can debug it better e.g. `projectname/sol/0` for exercise 0.

This will trigger the server to connect to your participant server and request a solution.

Therefore the submission server will send his request to the machine you specify as SERVER parameter.

It expects an endpoint:

`<your_ip>:<your_port>/<your_endpoint>`

to which it sends a html POST request with the input parameters for the exercise. The exercise parameters (if any) will be JSON formatted. Refer to the parameter section (JSON input) in each exercise for the keys.

After you get the parameters your server has to connect to HANA and solve the task with those parameters. This is a query to the database for example.

Afterwards your participant server has to send the result of this exercise (formatted as described in JSON output in each exercise) to the submission server.

**<exercise\_ID>** is 200 + exercise number e.g. for exercise 1 it is 201  
**<group\_ID>** is the number of your team  
**<your\_port>** is 6000 + <group\_ID> e.g. for <group\_ID> 9 it is 6009  
**<user\_name>** group + <group\_ID> e.g. for <group\_ID> 9 it is group9  
**<password>** the HANA database password

## Challenge 2: Tasks 0-4

**0. This exercise is to get your settings right. You should have a running participant server that sends us the solution which is the output JSON below.**

*Input:* -

*Output:* Send the JSON Output Example

JSON Input Example:

-

JSON Output Example :

```
{ "solution": [
    { "company": "SAP" }
]}
```

**1. Return the latest stock price of company, whose symbol contains 'Y'.**

*Input:* String Y

*Output:* Company Symbol, Stock Price (close)

JSON Input Example:

```
{ "letter": "Y" }
```

JSON Output Example :

```
{ "solution": [
    { "symbol": "BBY", "close": "72.95" },
    { "symbol": "BYZ", "close": "74.33" }
]}
```

**2. At which day the most stocks were traded? How many stocks were traded?**

*Input:* -

*Output:* date, amount of stocks

JSON Input Example: -

JSON Output Example :

```
{ "solution": [
    { "date": "14.04.2013", "amount": "65" }
]}
```

### 3. List the company names whose stocks were traded

*Input:* -

*Output:* company symbol, company name

JSON Input Example: -

JSON Output Example :

```
{ "solution": [
  { "symbol": "FFIV", "name": "F5 Networks" },
  { "symbol": "FISV", "name": "Fiserv" }
]}
```

### 4. Calculate the average monthly stock price (closing) of company X?

*Input:* string X of the company symbol

*Output:* for each company and month of trade data: the year, the month, the average stock price

JSON Input Example:

```
{ "symbol": "BBY" }
```

JSON Output Example :

```
{ "solution": [
  { "year": "2010", "month": "03", "average": "72.95" },
  { "year": "2011", "month": "04", "average": "73.95" }
]}
```