**Summary**

A web service (HTTP REST API) to work with tournaments.

**Glossary**

A *Tournament* is an object consists of max number of participants multiples of 8 (8, 16, 32, etc...) and number of [single-elimination](https://en.wikipedia.org/wiki/Single-elimination_tournament) matches that will be played.

A *Match* is an object consist of two participants, start time, finish time and participant scores.

A *Participant* is an object consist of unique id and unique nickname or even just unique id. The mechanism for creating and storing members is not important. Any options are acceptable from filling in the *java.util.Map* by hand or saving to the database via endpoints.

**Details**

Operations to be provided by the web service:

* Creating a tournament.
* Getting a tournament by its identifier.
* Starting a tournament. Each tournament must have that endpoint, which must generate match grid and appoint opponents for participants (in random way). (including when there are an odd number of participants, e.g. 7/8 or 15/16 participants).
* Adding participants in tournament.
* Removing participants from tournament while it's on hold.
* Summarizing results in match.

**Requirements**

* The API must conform to the REST architecture.
* Do only the server side, you don't need to do visualization.
* It should be a Spring Boot application.
* Maven or Gradle should be used as a build tool.
* Data should only be stored in RDBMS.
* Al least 30% of the code should be covered by tests.
* Submit sources via a public git repository.
* You should’t use @RepositoryRestController.