**Topcoder - Populate Marathon Match Related Data Into Challenge Model In Elasticsearch 1.0**

**Verification Guide**

# 

# Setup

For the feeder service, run 'mvn clean package' to compile the code and then cd local folder and run './run.sh' to start the services. Before starting the services, you need to first setup the environment as following sections.

The direct app docker is https://github.com/appirio-tech/tc-common-tutorials/tree/master/docker/direct-app

Please follow the above url to configure and setup the tc direct app docker.

For this challenge, you only need to start up the informix database by running:

docker-compose up tc-informix

If you want to push the data to the amazon elasticsearch service, please refer to Verification\_Topcoder - Elasticsearch Feeder Service For Challenges version 1.0.docx.

For local ElasticSearch, you can use the ElasticSearch docker container in

<https://github.com/topcoder-platform/challenges-logstash-conf/blob/dev/docker-compose.yml>.

Run 'docker-compose up elasticsearch' to start the service.

elasticsearch:

image: elasticsearch:2.3

ports:

- 9200:9200

- 9300:9300

select the informixoltp database and run:

delete from round\_prize;

insert into round\_prize values (13675, 1, 200);

insert into round\_prize values (13675, 2, 100);

delete from round\_registration;

insert into round\_registration values (13675, 132456, current, 0, null);

insert into round\_registration values (13675, 132458, current, 0, null);

delete from round\_event;

insert into round\_event values (13675, 1, 'event name 1', 'reg url 2');

insert into round\_event values (13675, 2, 'event name 1', 'reg url 2');

delete from long\_submission;

delete from long\_component\_state;

insert into long\_component\_state values (1, 13675, 132456, 2041, 99, 130, 2, 2);

insert into long\_submission (long\_component\_state\_id, submission\_number, example, open\_time, submit\_time, submission\_points) values (1, 2, 2, 132000, 1517544546857, 99);

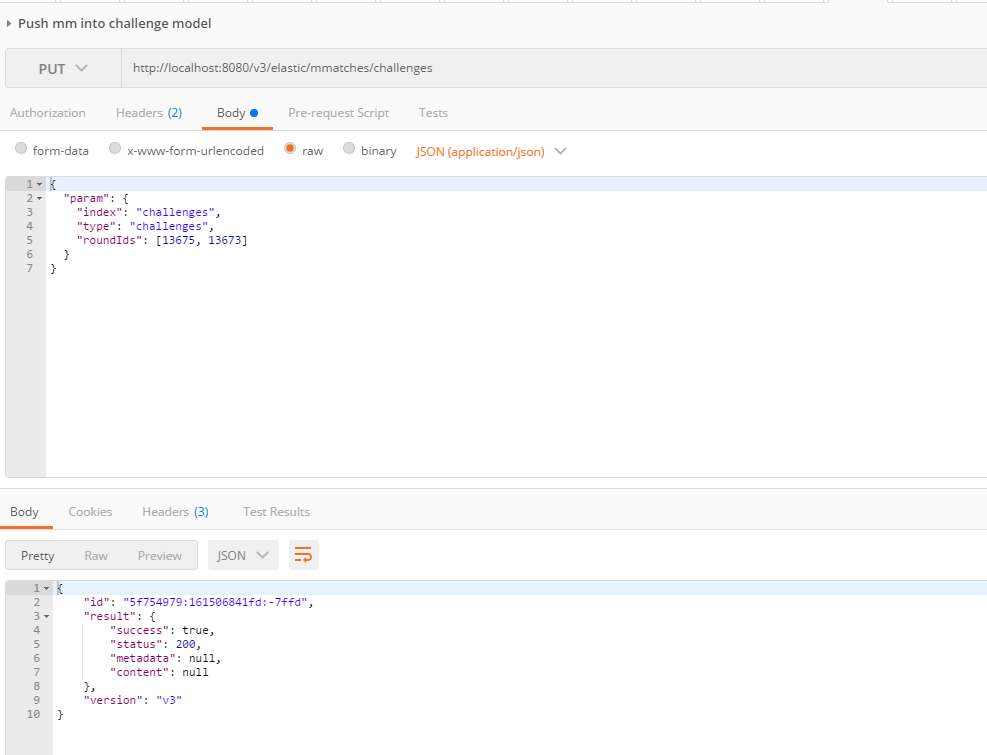
delete from algo\_rating where coder\_id = 132456;

insert into algo\_rating values (132456, 1000, 0, null, 0, 3, current);

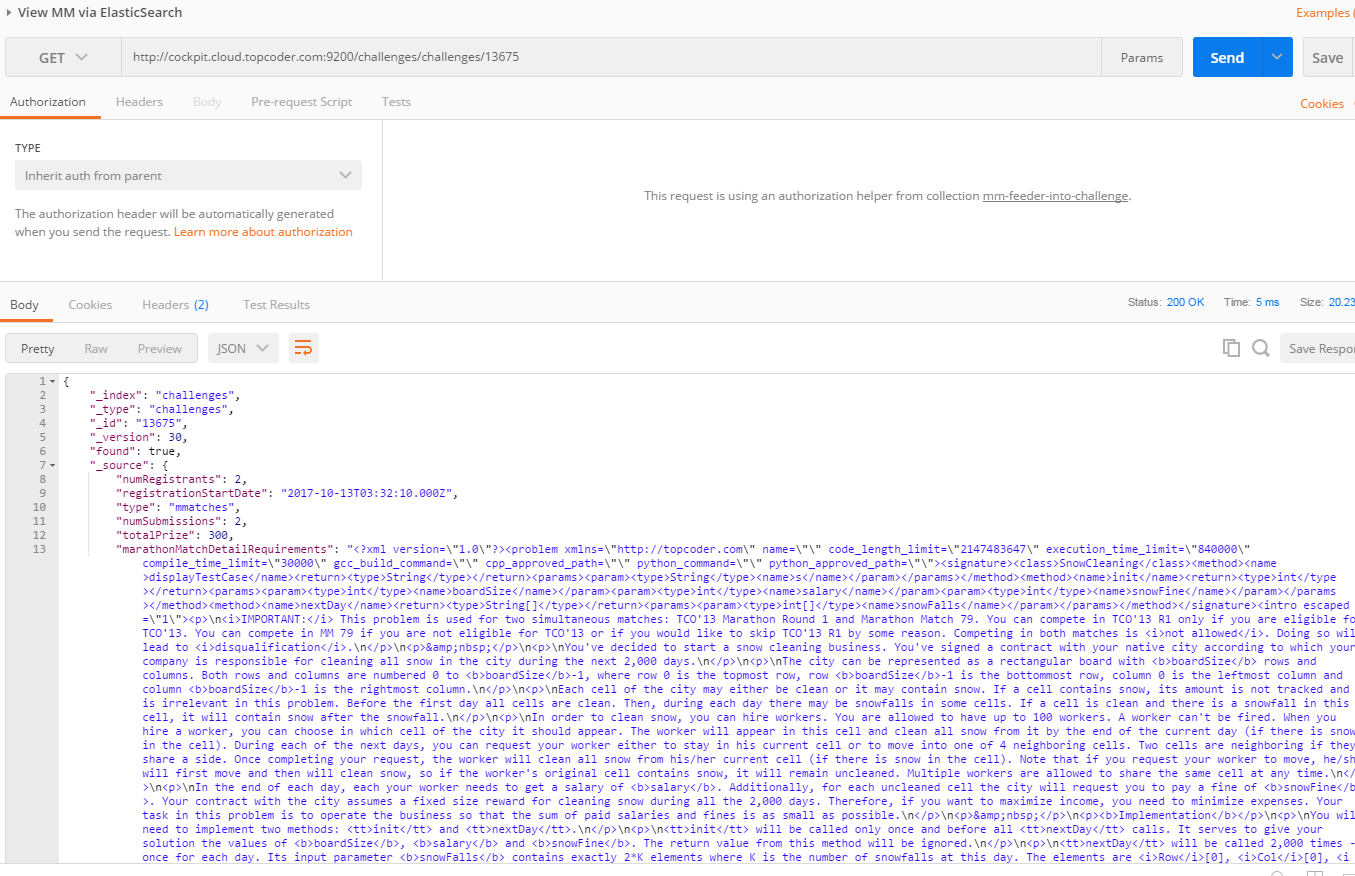
# Verify

Import the docs/ mm-feeder-into-challenge.postman\_collection.json into the postman.

Push mm into challenge model



View MM via ElasticSearch

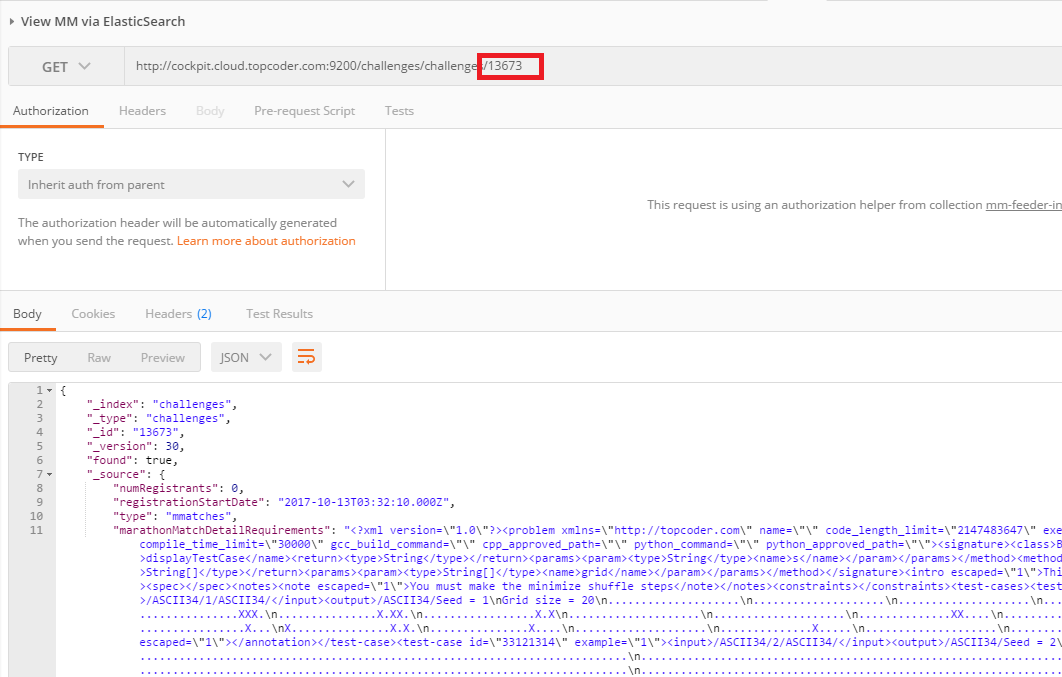


scroll down:





You can change the round id to 13673:



For other invalid conditions, check them one by one similarly.

This challenge also refactors some common used methods from the challenge feeder manager to the util classes. Please refer to Verification\_Topcoder - Elasticsearch Feeder Service For Challenges version 1.0.docx about how to verify the challenge feeders