[**TC Elasticsearch feeder - Add Job For populating rounds indexversion 1.0**](https://software.topcoder.com/review/actions/ViewProjectDetails?pid=30076560)

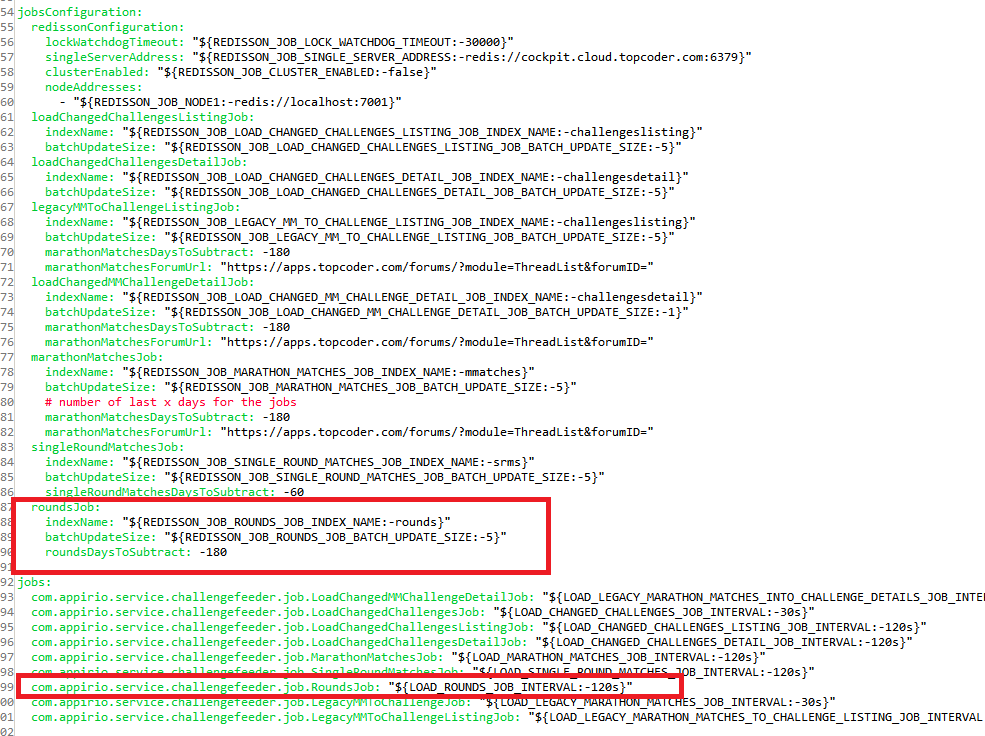
**Verification Guide**

# 

# Setup

The code is based on the commit 279700b3e70c33c558945aea1239ffcc0270870c of the dev branch of https://github.com/topcoder-platform/tc-elasticsearch-feeder-service . You can clone the code base from github and copy the whole source code to it. Or use the whole source code directly.

This challenge introduces new configuration as below:



You can use the default configuration values above

In order to run the rounds job, enter tc-elasticsearch-feeder-service, and 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

For this challenge, you only need to start up the following services by running:

docker-compose up elasticsearch tc-informix tc-cache

You might need to run 'docker-compose down' to tear down the containers if you have run the above command.

Then, add some test data to the informix database, select the informixoltp database and run:

delete from round\_registration;

INSERT INTO round\_registration (round\_id, coder\_id) VALUES (13673, 124766);

INSERT INTO round\_registration (round\_id, coder\_id) VALUES (13673, 124834);

delete from long\_comp\_result;

INSERT INTO long\_comp\_result (round\_id, coder\_id, rated\_ind) VALUES (13673, 124766, 1);

INSERT INTO long\_comp\_result (round\_id, coder\_id, rated\_ind) VALUES (13673, 124834, 0);

insert into round\_registration values(13672, 132456, current,1,null);

insert into round\_registration values(13672, 132458, current,1,null);

insert into round\_registration values(13676, 132456, current,1,null);

delete from room\_result;

insert into room\_result (round\_id, room\_id, coder\_id, rated\_flag) values (13672, 299156, 132456, 1);

insert into room\_result (round\_id, room\_id, coder\_id, rated\_flag) values (13672, 299156, 132458, 1);

insert into room\_result (round\_id, room\_id, coder\_id, rated\_flag) values (13676, 299156, 132458, 1);

Then cd es-mappings directory, and run delete if there is such index(assume that curl is installed in your OS):

curl -XDELETE http://cockpit.cloud.topcoder.com:9200/rounds/

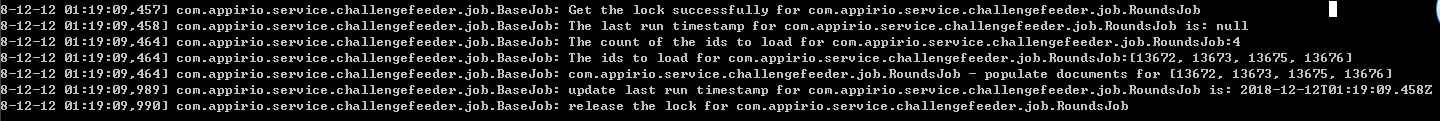
run:

curl -XPUT http://cockpit.cloud.topcoder.com:9200/rounds -d @round.json

There should be no error for the put operation. It's assumed that in the hosts file, cockpit.cloud.topcoder.com is configured for the docker IP that holds the elasticsearch service.

# Verify

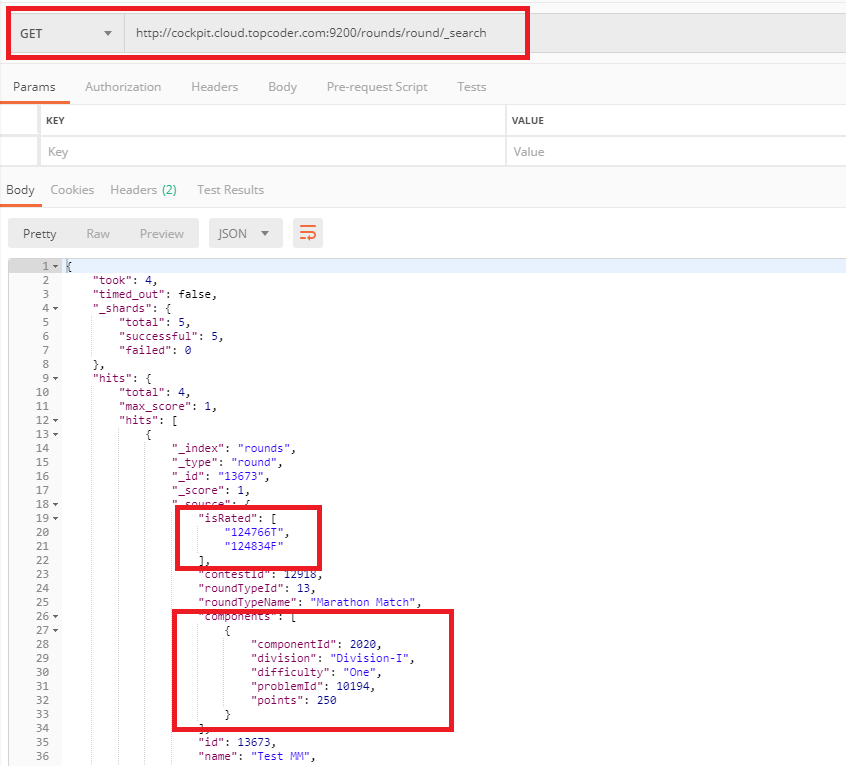
Wait for several minutes for the jobs to finish the loading. In the console, check that the rounds job finished the loading:



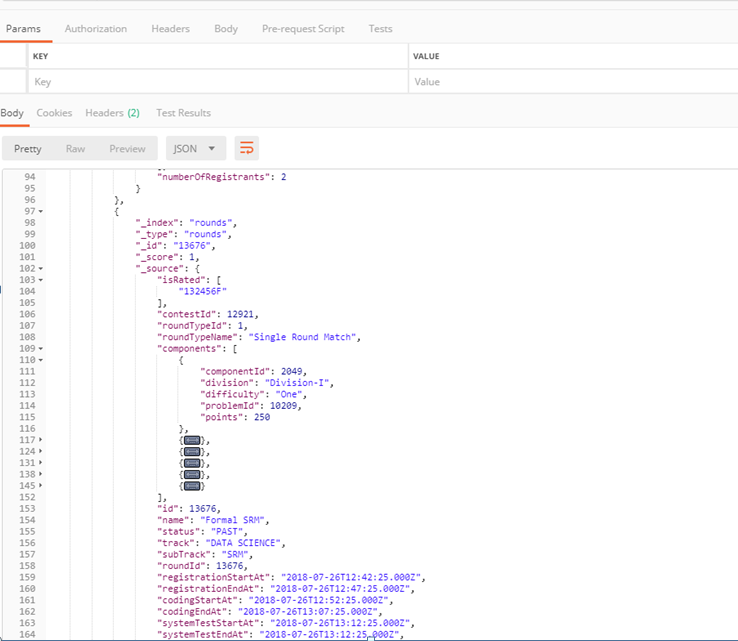
You can use curl to fetch the data from ' http://cockpit.cloud.topcoder.com:9200/rounds/round/\_search' via get method.

curl -XGET http://cockpit.cloud.topcoder.com:9200/rounds/round/\_search

Postman can also be used for the formatted result. Open the postman, and send 'http://cockpit.cloud.topcoder.com:9200/rounds/round/\_search' with http Get method, the result will be:



scroll down, and check SRM:



select the informixoltp database and run:

delete from round\_registration;

delete from long\_comp\_result;

delete from room\_result;

Then wait for several minutes for the rounds job to finish the loading, then check the postman again, all the userIds and isRated should be cleared:

