Steps to execute the benchmark

1. Build `client-benchmark-noop-api-plugin` with `./gradlew :client:client-benchmark-noop-api-plugin:assemble`
2. Install it on the target host with `bin/elasticsearch-plugin install file:///full/path/to/client-benchmark-noop-api-plugin.zip`.
3. Start Elasticsearch on the target host (ideally *not* on the machine that runs the benchmarks)
4. Run the benchmark with

```
./gradlew -p client/benchmark run --args ' params go here'
```

Everything in the `'` gets sent on the command line to JMH. The leading inside the `'`s is important. Without it parameters are sometimes sent to gradle.

See below for some example invocations.

Example benchmark

In general, you should define a few GC-related settings `-Xms8192M -Xmx8192M -XX:+UseConcMarkSweepGC -verbose:gc -XX:+PrintGCDetails` and keep an eye on GC activity. You can also define `-XX:+PrintCompilation` to see JIT activity.

Bulk indexing

Download benchmark data from http://benchmarks.elasticsearch.org.s3.amazonaws.com/corpora/geonames and decompress them.

Example invocation:

```
wget http://benchmarks.elasticsearch.org.s3.amazonaws.com/corpora/geonames/documents-2.json.bz2
bzip2 -d documents-2.json.bz2
mv documents-2.json client/benchmark/build
gradlew -p client/benchmark run --args ' rest bulk localhost build/documents-2.json geonames type 8647880 5000'
```

The parameters are all in the `'`s and are in order:

- Client type: Use either "rest" or "transport"
- Benchmark type: Use either "bulk" or "search"
- Benchmark target host IP (the host where Elasticsearch is running)
- full path to the file that should be bulk indexed
- name of the index
- name of the (sole) type in the index
- number of documents in the file
- bulk size

Search

Example invocation:

```
./gradlew -p client/benchmark run --args ' rest search localhost geonames {"query":{"match_phrase":{"name":"Sankt Georgen"}}} 500,1000,1100,1200'
```

The parameters are in order:

- Client type: Always "rest"
- Benchmark type: Use either "bulk" or "search"
- Benchmark target host IP (the host where Elasticsearch is running)
- name of the index
- a search request body (remember to escape double quotes).
- A comma-separated list of target throughput rates