## [Xray Cloud] How to see more than 100 Tests in a Test Execution via GraphQL

When getting information via GraphQL, we always have a limit of 100 issues to be shown per time.

Problem:

And if we have a Test Execution with more that 100 Tets, can we see them all?

Answer:

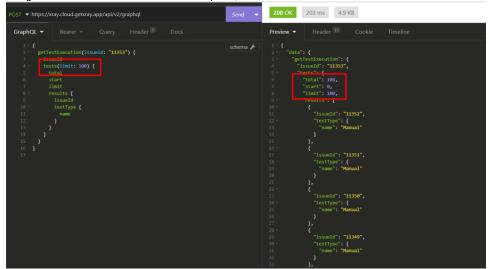
Yes, you can! The results only appear paginated. So in order to see them all, we have to make different queries to show the results.

Let's see a practical example using Insomnia:

• Test Execution with 108 Test Cases:

## Test Exec 1 Comment Comment Selected for Development Workflow ✓ ■ Test Execution Status: BACKLOG (View workflow) Priority: ↑ Medium Resolution: Unresolved Labels: None Description Click to add description Tests Create Test + Add ~ **Overall Execution Status** TOTAL TESTS: 108

• Using the function GetTestExecution from GraphQL, we will be able to see the first 100 Tets:



To see from 100, it's necessary to use the Start parameter as shown below. This will show you the tests from the 100° one.

```
GraphOL ▼ Bearer ▼ Query Header □ Docs

Preview ▼ Header □ Cookie Timeline

1 * {
2 * getTestExecution(issueId: "11353") {
3 * issueId
4 * Tests(start: 100, limit: 100) {
5 * tettal
6 * start
7 * limit
8 * results {
9 * issueId
10 * testType {
11 * name
12 * }
13 * }
14 * }
15 * }
16 * {
17 * "issueId": "11252",
18 * "results": [
10 * "issueId": "11252",
19 * "issueId": "11251",
10 * "issueId": "11251",
11 * "issueId": "11251",
12 * "issueId": "11251",
13 * "issueId": "11251",
14 * "issueId": "11251",
15 * "issueId": "11250",
20 * "issueId": "11250",
21 * "issueId": "11250",
22 * "issueId": "11250",
23 * "issueId": "11250",
24 * "testType': {
25 * "name": "Manual"
26 * }
27 * ]
28 * [
29 * "issueId": "11240",
```

You can use any number in the start parameter. Imagine, if you have 300 Tests, then you'll have to make 3 queries: start: 0; start:100, start:200, and so on.

• It is possible to retrieve all these Tests in a single request since each GraphQL request supports 25 resolvers. This means that in a single request we can get information from 25x100= 2.5k issues. For that, just build the code as shown below:

```
POST ▼ https://xray.cloud.getxray.app/api/v2/graphql
                                                                 Send
GraphQL ▼
                                                          Docs
                 Bearer 🔻
                                Query
                                                            schema 🔑
        step1: getTestExecution(issueId: "11353") {
        step2: getTestExecution(issueId: "11353") {
```

## Related articles

GraphQL API

https://xray.cloud.getxray.app/doc/graphql/

https://xray.cloud.getxray.app/doc/graphql/gettestexecution.doc.html