## Способ 1. **По условию-работает** \*Всегда можно задать условие для возрата большого массива

|  |  |
| --- | --- |
| Запрос | {      "collectionId": "preOrders",          "filter": {          "field": "preOrderId",          "operator":">=",          "value": 0      }  } |
| Метод | module.exports.getDocumentsWithFilter = async function (*request*, *reply*) {    const { collectionId, filter } = *request*.body    const firestore = Firebase.admin.firestore()    try {      const collectionRef = firestore.collection(collectionId)      const query = collectionRef.where(filter.field, filter.operator, filter.value)      const querySnapshot = await query.get()      const documents = querySnapshot.docs.map((*doc*) => *doc*.data())      return { documents }    } catch (err) {      if (err.message === 'Response code 404 (Not Found)' || err.message === 'Not active') {        throw HttpError.NotFound('Not Found or Not active')      }      console.log(err)      throw HttpError.InternalServerError('Server error.')    }  } |
| Результат | {      "documents": [          {              "owner": "nobody",              "preOrderId": 0,              "description": "This is preOrder number 0",              "createdAt": {                  "\_seconds": 1684324547,                  "\_nanoseconds": 422000000              }          },          {              "owner": "nobody",              "preOrderId": 1,              "description": "This is preOrder number 1",              "createdAt": {                  "\_seconds": 1684324548,                  "\_nanoseconds": 344000000              }          },  …          {              "owner": "nobody",              "preOrderId": 1998,              "description": "This is preOrder number 1998",              "createdAt": {                  "\_seconds": 1684325253,                  "\_nanoseconds": 118000000              }          }      ]  } |

## Способ 2. **Через Метод FieldPath – ограничение до 30 записей**

|  |  |
| --- | --- |
| Запрос | {      "collectionId": "preOrders",  } |
| Метод | module.exports.getRandomDataFromDB = async function (*request*, *\_reply*) {    const { collectionId } = *request*.body    const firestore = Firebase.admin.firestore()    try {      const collectionRef = firestore.collection(collectionId)      const documents = await collectionRef.listDocuments()      const documentsIds = documents.map((*document*) => *document*.id)      const preOrdersDocs = await collectionRef.where(Firebase.admin.firestore.FieldPath.documentId(), 'in', documentsIds).get();      console.log(preOrdersDocs)      return { preOrdersDocs }    } catch (err) {      console.log(err)      throw HttpError.InternalServerError('Server error.')    }  } |
| Результат | at process.processTicksAndRejections (node:internal/process/task\_queues:95:5) {code: 3,details: "'IN' supports up to 30 comparison values.",metadata: Metadata { internalRepr: Map(0) {}, options: {} }} |