

Resource limits are one of the most common errors Salesforce developers come across. That’s the multi-tenant life, and we take the pros with the cons. I’m sure you’ve seen this one.

Too many query rows : 50001.

Of course, there are dozens of help articles and forum posts that prescribe solutions. Use WHERE and LIMIT conditions to reduce the scope of records to retrieve. Consider chunking the work using batchable jobs.

. . .

But what if you must operate on all the records for a given object? Do you need circular dependency checking, and thus need the entire data set in memory?

Then, you’re in luck! Start by creating a Queueable Apex class, and placing the large query in the Execute method. Next, mark the method as ReadOnly by using the @ReadOnly annotation.

```
global class BigQueryJob implements Queueable {

    @ReadOnly
    global void execute(QueueableContext context) {
        List<Account> allAccounts = [SELECT Id, Name FROM Account];
    }
}
```

Finally, enqueue the job (either via automation or the dev console). When the job runs... Voila! You now have 1 Million Query Rows at your disposal!

15:59:20:035	LIMIT_USAGE_...	(default)
15:59:20:000	LIMIT_USAGE_...	Number of SOQL queries: 1 out of 100
15:59:20:000	LIMIT_USAGE_...	Number of query rows: 4 out of 1000000
15:59:20:000	LIMIT_USAGE_...	Number of SOSL queries: 0 out of 20
15:59:20:000	LIMIT_USAGE_...	Number of DML statements: 0 out of 0
15:59:20:000	LIMIT_USAGE_...	Number of DML rows: 0 out of 0
15:59:20:000	LIMIT_USAGE_...	Maximum CPU time: 0 out of 10000
15:59:20:000	LIMIT_USAGE_...	Maximum heap size: 0 out of 6000000
15:59:20:000	LIMIT_USAGE_...	Number of callouts: 0 out of 100

Of course, don’t be surprised if you run into Heap or CPU Timeout limits when operating on a large data set. If you’re stuck, try the SOQL For Loop or Aggregate SOQL approaches. Maybe you don’t need that map of objects in scope after all.

It’s worth noting that calling the “Execute” method will not grant the limit increase. It must be called from the context of a job to have the full 1 Million rows.

ReadOnly annotation is also available in other contexts of the platform. For example, if you’d like to display a list of over 1,000 records in Visualforce.