

# Useful Resources & Links

---

From: 'NodeJS - The Complete Guide (incl. MVC, REST APIs, GraphQL)'

## Section 19: Error Handling

Error Handling in Express.js - Official Docs: <https://expressjs.com/en/guide/error-handling.html>

---

## Section 20: File Upload & Download

Multer Official Docs: <https://github.com/expressjs/multer>

Streaming Files: <https://medium.freecodecamp.org/node-js-streams-everything-you-need-to-know-c9141306be93>

Generating PDFs with PDFKit: [http://pdfkit.org/docs/getting\\_started.html](http://pdfkit.org/docs/getting_started.html)

---

## Section 21: Adding Pagination

### ADDITIONAL INFO

#### Skip & Limit with SQL

When using MongoDB, you can use `skip()` and `limit()` as shown in the last lecture.

*But how would that work in SQL?*

Here's how you would implement pagination in SQL code:

<https://stackoverflow.com/questions/3799193/mysql-data-best-way-to-implement-paging>

To quickly sum it up: The **LIMIT** command allows you to restrict the amount of data points you fetch, it's your `limit()` equivalent. Combined with the **OFFSET** command (which replaces `skip()`), you can control how many items you want to fetch and how many you want to skip.

When using Sequelize, the official docs describe how to add pagination:

<http://docs.sequelizejs.com/manual/tutorial/querying.html#pagination-limiting>

---

## Section 22: Understanding Async Requests

More on the fetch API: <https://developers.google.com/web/updates/2015/03/introduction-to-fetch>

More on AJAX Requests: [https://developer.mozilla.org/en-US/docs/Web/Guide/AJAX/Getting\\_Started](https://developer.mozilla.org/en-US/docs/Web/Guide/AJAX/Getting_Started)

---