Primeira parte do Curso:

* Como o backend entrega os dados para o frontend;
* O que é uma API REST, que é uma interface que vai fornecer dados e recursos baseados em requisições e respostas HTTP;
* Iniciar um novo projeto Node através do comando npm init;
* Personalizar o arquivo de configuração package.json com scripts e as informações individualizadas do projeto;
* Utilizar um gerenciador de pacotes, como o NPM, para instalar novos módulos que serão utilizados no projeto;
* Como está estruturada a pasta node\_modules e a importância de incluí-la no arquivo .gitignore;
* Construir um servidor local utilizando o módulo http do Node.js;
* Realizar a criação de novas rotas e qual o objetivo das mesmas;
* Instalação e utilização do nodemon para não ter que reiniciar o servidor a cada alteração.

Segunda parte do curso:

* A instalar o Express e os benefícios de se usar um framework para o desenvolvimento de aplicações Web;
* Criar um servidor local usando o Express e configurando-o para escutar requisições em uma porta específica do nosso computador;
* Como estruturar as requisições da API, utilizando corretamente os verbos HTTP e padronizando o endpoint de acordo com o padrão REST;
* Devolver dados como resposta às requisições, usando o padrão JSON;
* Utilizar o Postman para fazer as requisições do tipo GET, POST, PUT e DELETE, cujas últimas três não conseguimos simular pelo navegador.

Terceira parte do curso:

* A diferença entre bancos de dados relacionais e NoSQL;
* Como utilizar o MongoDB Atlas, que é o serviço de banco de dados na nuvem, onde você pode escolher hospedar seu banco em plataformas como Google Cloud Platform, AWS ou Azure;
* Instalar e utilizar a biblioteca Mongoose que é uma solução baseada em esquemas (Schemas) para modelar o banco de dados;
* Criar o schema de livros, para representar a coleção livros no banco de dados, definindo o como será o documento armazenado e os campos requeridos;
* Fazer a conexão da API com o banco de dados, verificando se a mesma ocorreu com sucesso ou não.

Quarta parte do curso:

* A separar os arquivos em pacotes específicos, de acordo com um objetivo em comum, como models, controllers, routes;
* A criar uma classe com métodos estáticos, para usar no arquivo de rotas, deixando o código mais intuitivo e descritivo sobre o que será executado para cada rota;
* A utilizar alguns métodos fornecidos pela biblioteca mongoose para a manipulação dos dados no banco, como *find*, *findById*, *findByIdAndUpdate* e *findByIdAndDelete*.

Quinta parte do curso:

* A incluir um novo schema e facilmente vinculá-lo ao arquivo index.js para que o novo endpoint seja reconhecido e esteja operante rapidamente;
* A associar dados entre duas coleções do banco, fazendo com que ao recuperar os dados do livro, por exemplo, já fossem demonstrados todos os dados cadastrados no endpoint de autores;
* Utilizar o comando populate() para fazer a associação de dados, determinando campos específicos ou selecionando todos os dados do recurso;
* A criar uma busca de dados por um campo que não é o identificador único do documento, como a editora e usar o recurso de passar no próprio endpoint o valor do parâmetro que servirá como critério de busca.