Teste - Desenvolver uma api rest utilizando Java Spring Boot e MongoDB, o cenário é livre.

1 - Qual a diferença entre Spring e Spring Boot ?

Spring Framework

Spring é um dos mais usados Java EE Frameworks para criação de aplicativos. Para a plataforma Java, o framework Spring fornece um modelo elaborado de programação e configuração. O objetivo é simplificar o desenvolvimento do Java EE e ajuda os desenvolvedores a serem mais produtivos no trabalho. Pode ser usado em qualquer tipo de plataforma de implantação. Leva em consideração as crescentes necessidades das empresas atuais e se esforça para atendê-las.

Ao contrário de outros frameworks, o Spring se concentra em várias áreas de um aplicativo e fornece uma ampla gama de recursos.

Um dos principais recursos do framework Spring é a injeção de dependência. Isso ajuda a simplificar as coisas, permitindo-nos desenvolver aplicativos fracamente acoplados.

Spring Boot

Embora o framework Spring se concentre em fornecer flexibilidade para você, o Spring Boot visa encurtar o tamanho do código e fornecer a maneira mais fácil de desenvolver um aplicativo da web. Com a configuração de anotação e os códigos padrão, o Spring Boot reduz o tempo envolvido no desenvolvimento de um aplicativo. Isso ajuda a criar um aplicativo independente com configuração menor ou quase zero.

Como sabemos, todo framework se baseia em alguns princípios. No caso do Boot, são quatro:

1. Prover uma experiência de início de projeto (getting started experience) extremamente rápida e direta;
2. Apresentar uma visão bastante opinativa (opinionated) sobre o modo como devemos configurar nossos projetos Spring mas, ao mesmo tempo, flexível o suficiente para que possa ser facilmente substituída de acordo com os requisitos do projeto;
3. Fornecer uma série de requisitos não funcionais já pré-configurados para o desenvolvedor como, por exemplo, métricas, segurança, acesso a base de dados, servidor de aplicações/servlet embarcado, etc.;
4. Não prover nenhuma geração de código e minimizar a zero a necessidade de arquivos XML.

2 - Quais as vantagens de utilizar o Spring Boot ?

As vantagens adicionais que vêm com o Spring Boot são de grande valor para os desenvolvedores, pois oferecem a conclusão de projetos com esforços mínimos. Para todos os problemas que surgem do framework Spring, o Spring Boot é a solução.

• O Spring Boot não exige a implantação de arquivos WAR.

• Cria aplicativos independentes.

• Ajuda a incorporar o Tomcat, Jetty ou Undertow diretamente.

• Não requer configuração XML.

• Tem como objetivo reduzir o LOC.

• Oferece recursos prontos para produção.

• É mais fácil de lançar.

• Personalização e gerenciamento mais fáceis.

3 - Como escrever testes de integração ?

**Descrição:** É a forma de se testar a combinação das unidades em conjunto.

**Objetivo:** Nesse caso, a ideia é encontrar falhas na junção destas unidades. Pode ser que a classe X funcione bem sozinha, mas ao ser utilizada pela classe Y, ela deixe de funcionar.

**Exemplo:** Colocar todo o software para rodar e começar a usar diversas funcionalidades consideradas centrais no seu programa para confirmar que ele roda e as principais funcionalidades fazem o esperado.

**Benefícios:** além de testar funcionalmente, pode assegurar performance e confiabilidade. Ajudam a garantir que o trabalho de um desenvolvedor não está afetando o trabalho de outro e em equipes grandes isso pode fazer toda a diferença se forem realizados com frequência.

**Teste Integrado** (ou "teste de integração"): você tem as seguintes funções (novamente simplistas):

def soma(a, b):

return a + b

def multiplicacao(a, b):

return a \* b

def minhaFuncaoNadaVer(a, b):

return soma(

multiplicacao(a, b),

multiplicacao(a + b, b)

)

Após você ter feito Testes Unitários para as funções soma e multiplicacão, agora você vai fazer um Teste de Integração da função minhaFuncaoNadaVer, afinal ela integra duas funções do seu sistema (soma e multiplicacao).

Em Python:

assert minhaFuncaoNadaVer(12, 6) == 180

assert minhaFuncaoNadaVer(1, 90) == 8280

4 - Desenvolver uma api rest utilizando Java Spring Boot e MongoDB, o cenário é livre.

5 - As ferrentas deste projeto devem ser as seguintes:

- Java

- Spring Boot

- Undertow como webserver

- Gradle

- MongoDB

Links que podem ajudar no processo de desenvolvimento:

- https://www.mongodb.com/cloud/atlas

- https://www.mongodb.com/products/compass

- https://www.mockable.io

- http://ptsv2.com/

Ferramentas Utilizadas

1 – Ambiente Linux Ubuntu 18.04

2 – IDE Eclipse Version: 2019-06 (4.12.0) Build id: 20190614-1200

3 - Versão Java 1.8

4 - Spring framework - Spring boot version '2.1.7.RELEASE

a - Spring Boot Data MongoDB Starter

b - Spring Boot Data REST Starter

c - Spring Boot Undertow Starter

d - Spring Boot Web Starter

5 - Gradle - Sistema avançado de automatização de builds que une o melhor da flexibilidade do Ant com o gerenciamento de dependencias e as convenções do Maven.

6 – MongoDB - Bancos de dados NoSQL - [orientado a documentos](https://pt.wikipedia.org/wiki/Banco_de_dados_orientado_a_documentos) [livre](https://pt.wikipedia.org/wiki/Software_livre), de [código aberto](https://pt.wikipedia.org/wiki/Código_aberto) e [multiplataforma](https://pt.wikipedia.org/wiki/Multiplataforma), escrito na [linguagem](https://pt.wikipedia.org/wiki/Linguagem_de_programação) C++.

7 - Robo 3T 1.3.1 Shell-centric MongoDB management tool.

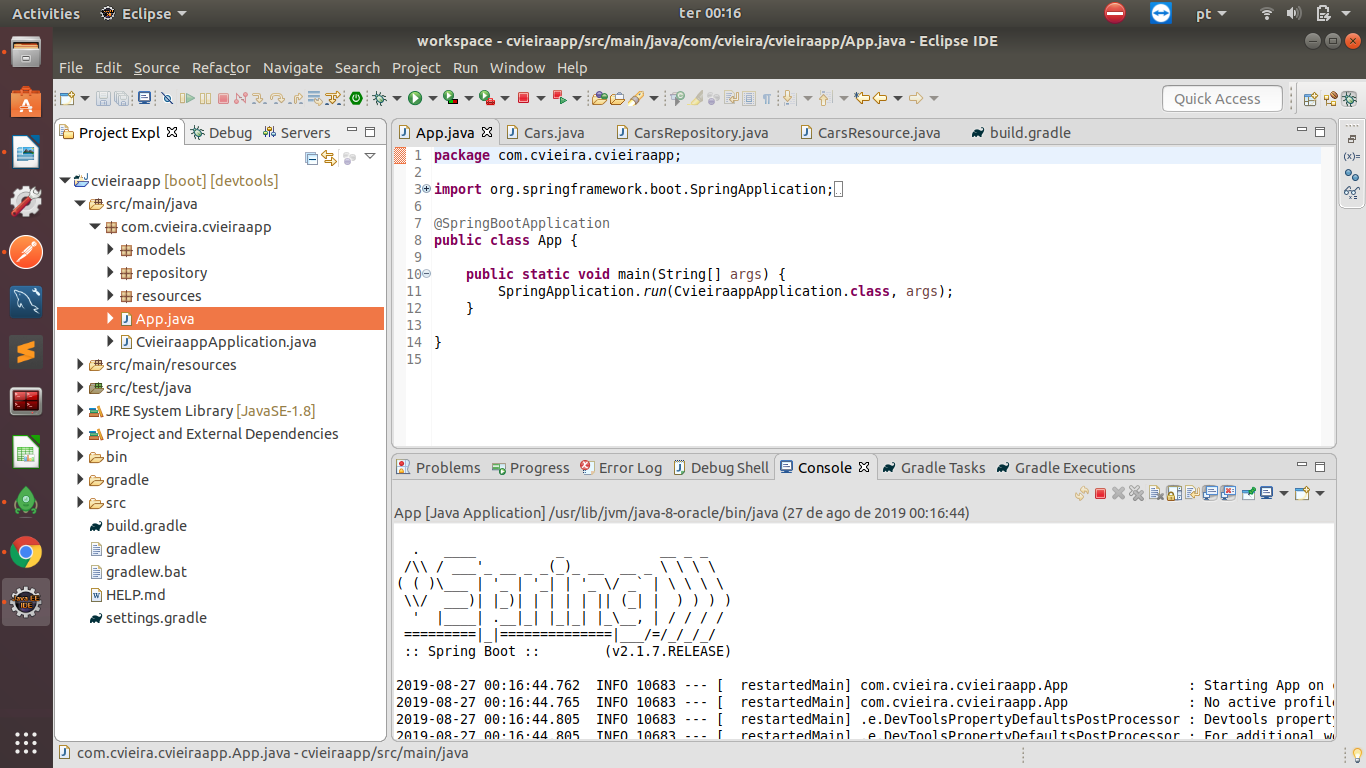
8 - Postman helps you develop APIs faster Postman for Linux Version 6.7.1 linux 4.15.0-58-generic / x64

8 - Git

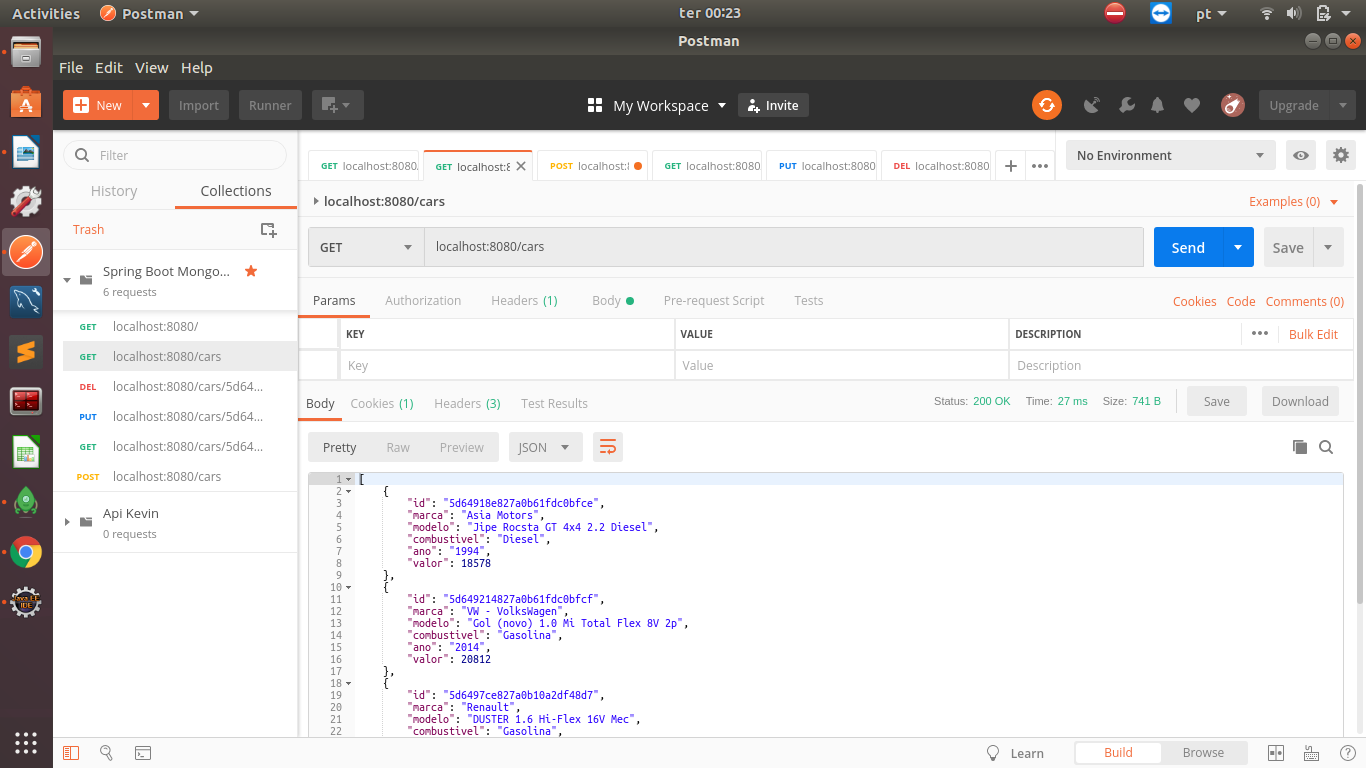
REST api’s – Informações sobre o Veículo

**CRUD** (**Create,** **Read,** **Update and**  **Delete)**

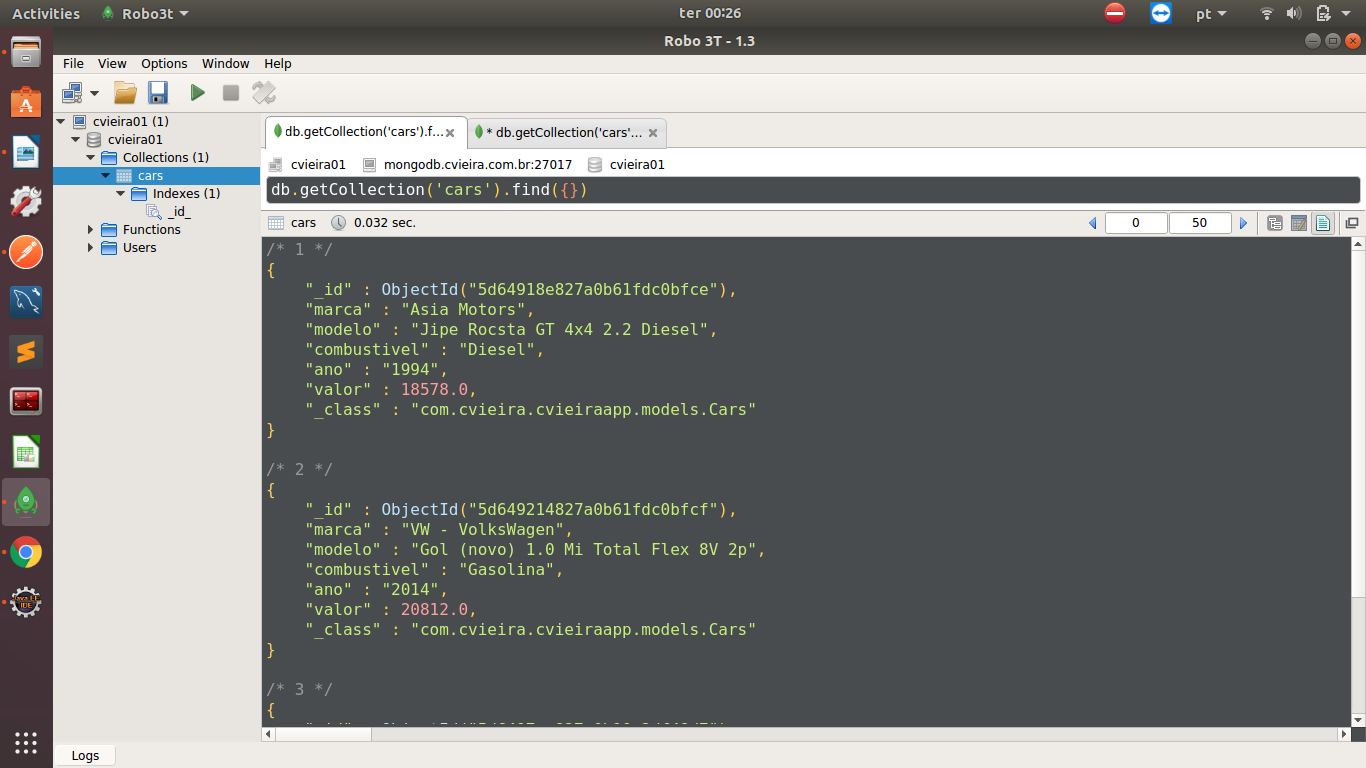
**IDE: Projeto**



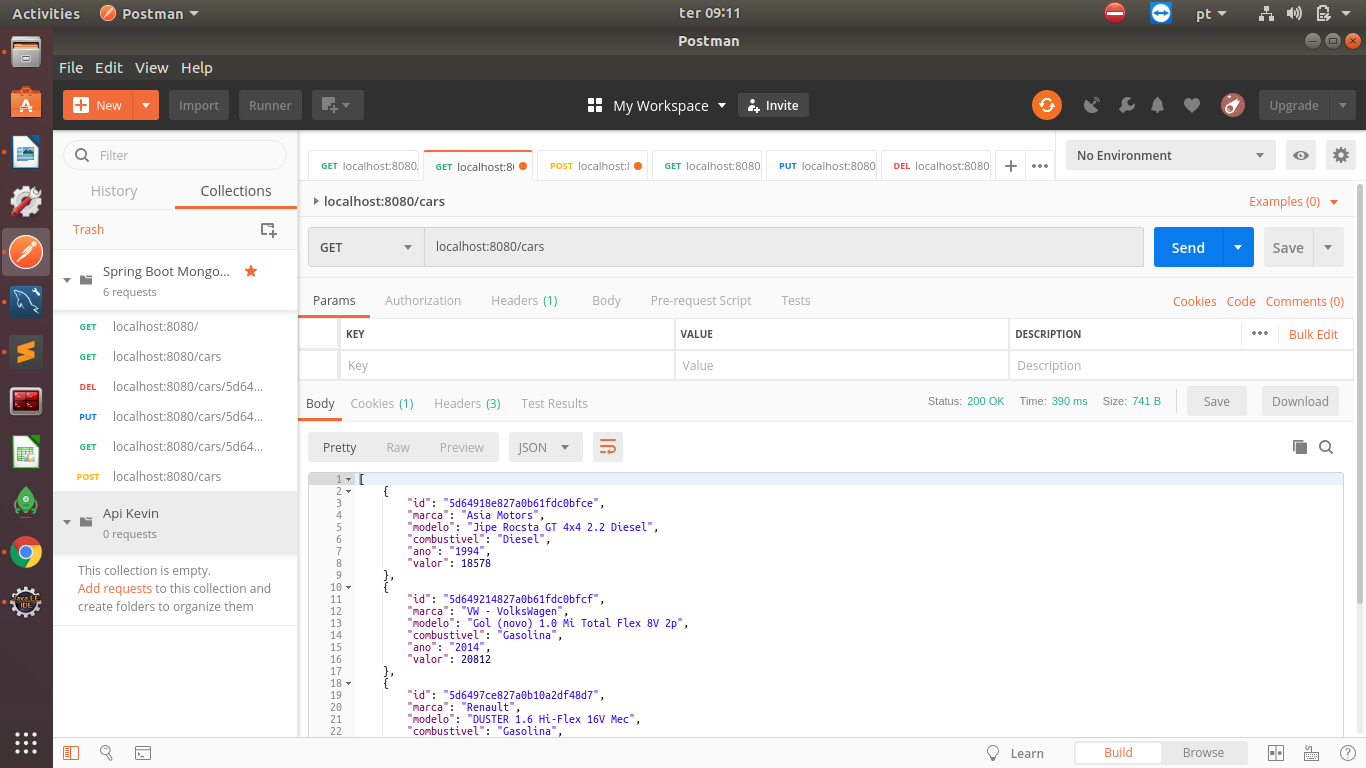
Postman:



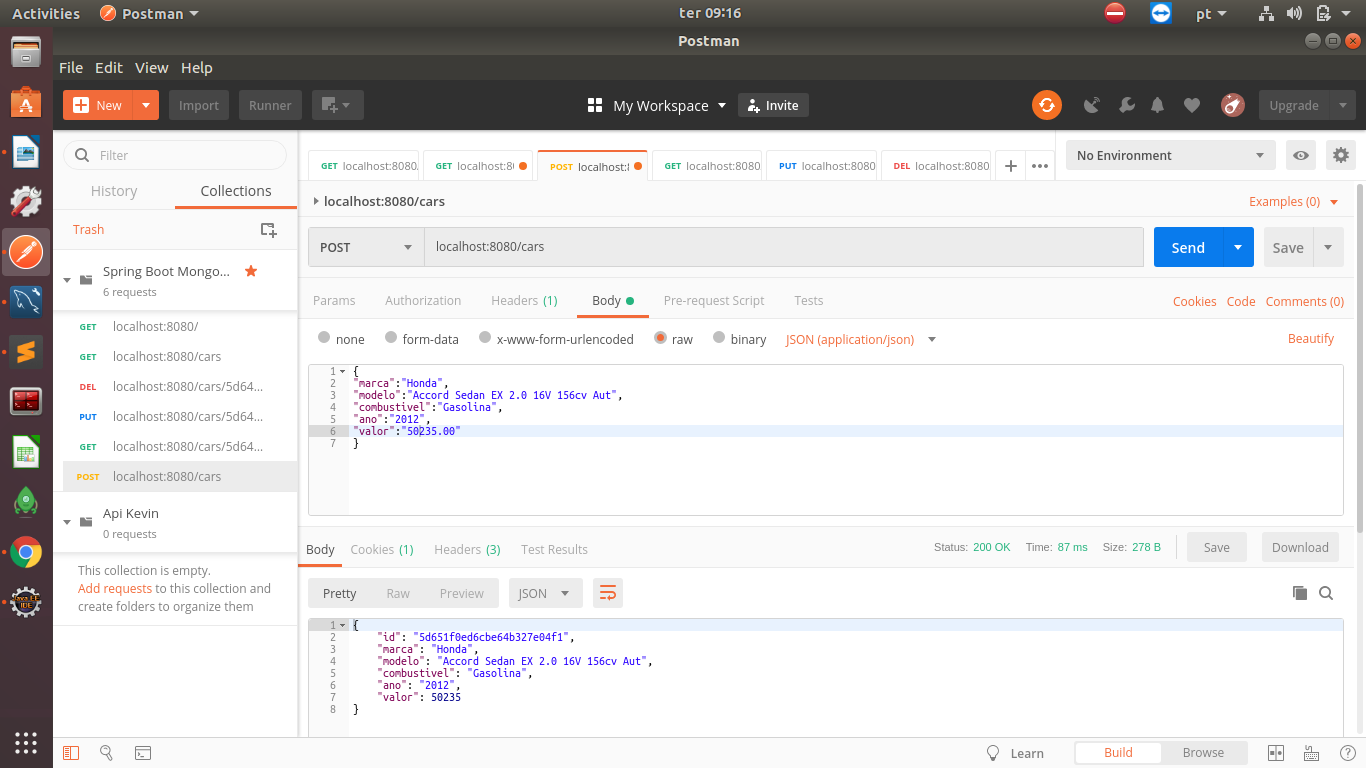
Robo 3T:

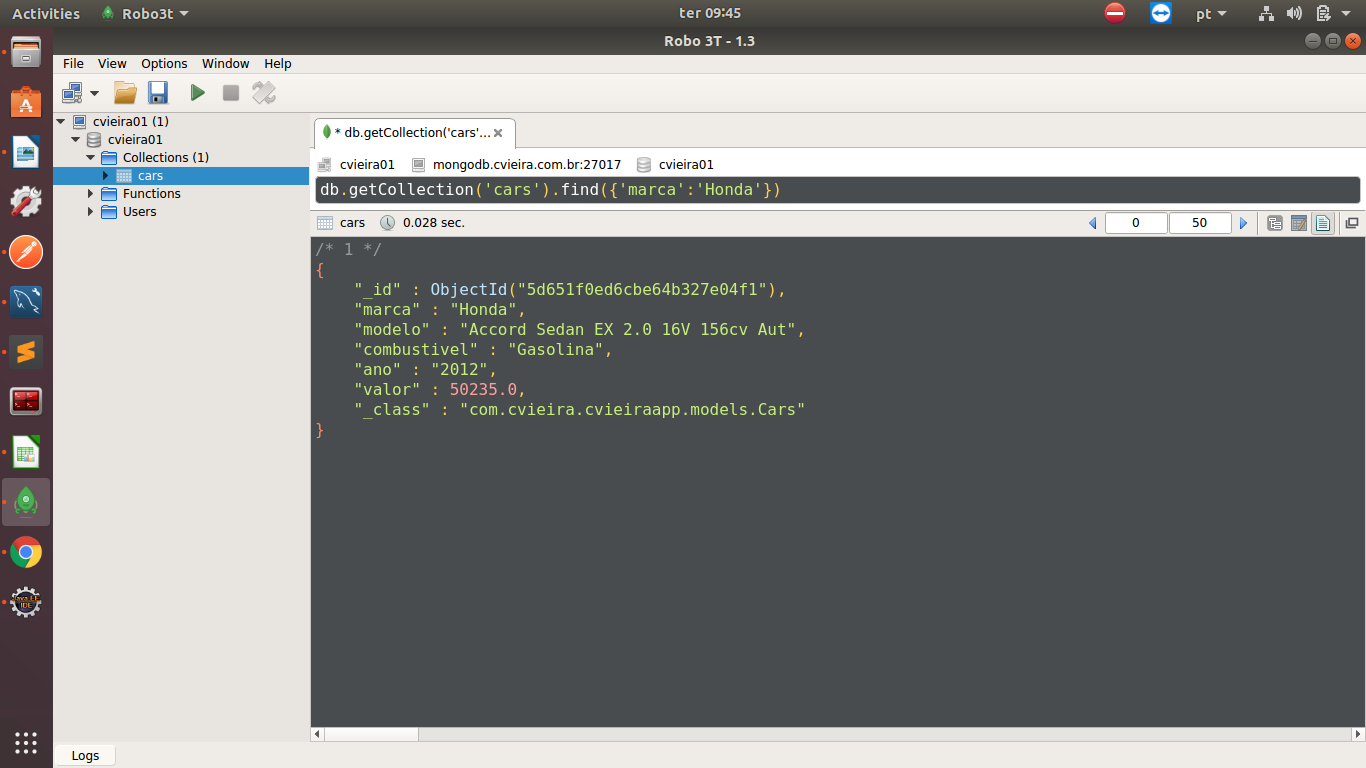


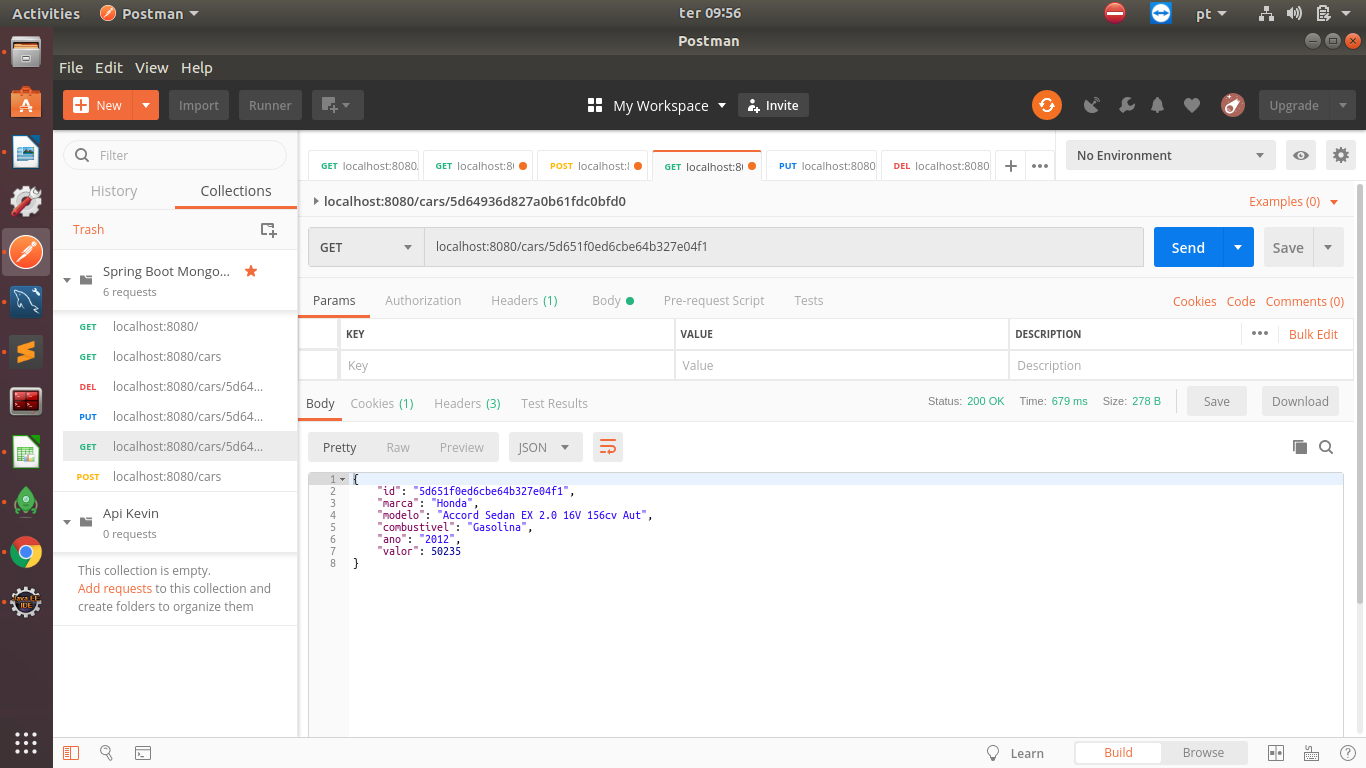
EXECUÇÃO DE TESTE:  **Read**



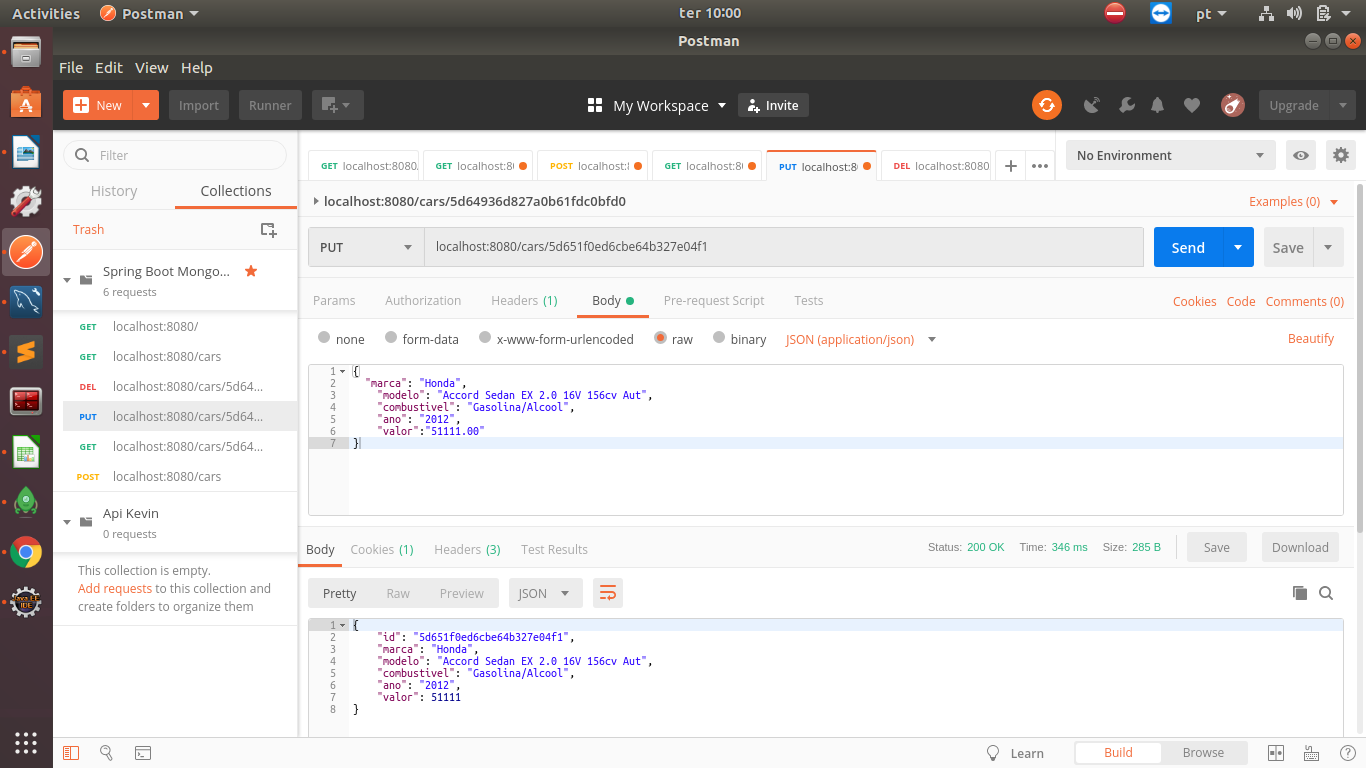
EXECUÇÃO DE TESTE: **Create**







EXECUÇÃO DE TESTE:  **Update**



EXECUÇÃO DE TESTE: **Delete**

