|  |  |  |
| --- | --- | --- |
|  | Universidade Federal de Alagoas  Ciência da Computação  Sistemas Distribuídos | Danillo Rodrigues Abreu  Luiz Fernando da Silva  17/01/2021 |
| **Projeto 2** – **ToDo List** | | |

**Sumário**

1. **Sobre o projeto**
2. **Instalando as dependências do projeto**
3. **Executando o projeto**
4. **Testando as rotas da API**
5. **Sobre o projeto**

**ToDo List** é uma API Rest desenvolvida com a linguagem de programação Python 3.9.x e o microframework Flask 1.1.x. Com esta API é possível visualizar, criar, atualizar e remover tarefas de uma lista. Sendo assim, para testar as rotas implementadas na aplicação recomenda-se o uso do software [Insomnia](https://insomnia.rest/download/core/?), no qual através de requisições HTTP (GET, POST, PUT e DELETE) descritas na seção 4, é permitido o acesso aos serviços implementados.

1. **Instalando as dependências do projeto**

* Criação do ambiente virtual:

– Windows:

python -m venv .venv

– Linux:

python3 -m venv .venv

* Ativação do ambiente virtual:

– Windows:

.venv\Scripts\activate

– Linux:

source .venv/bin/activate

* Instalação das dependências:

pip install -r requirements.txt

1. **Executando o projeto**

* Windows:

python app.py

* Linux:

python3 app.py

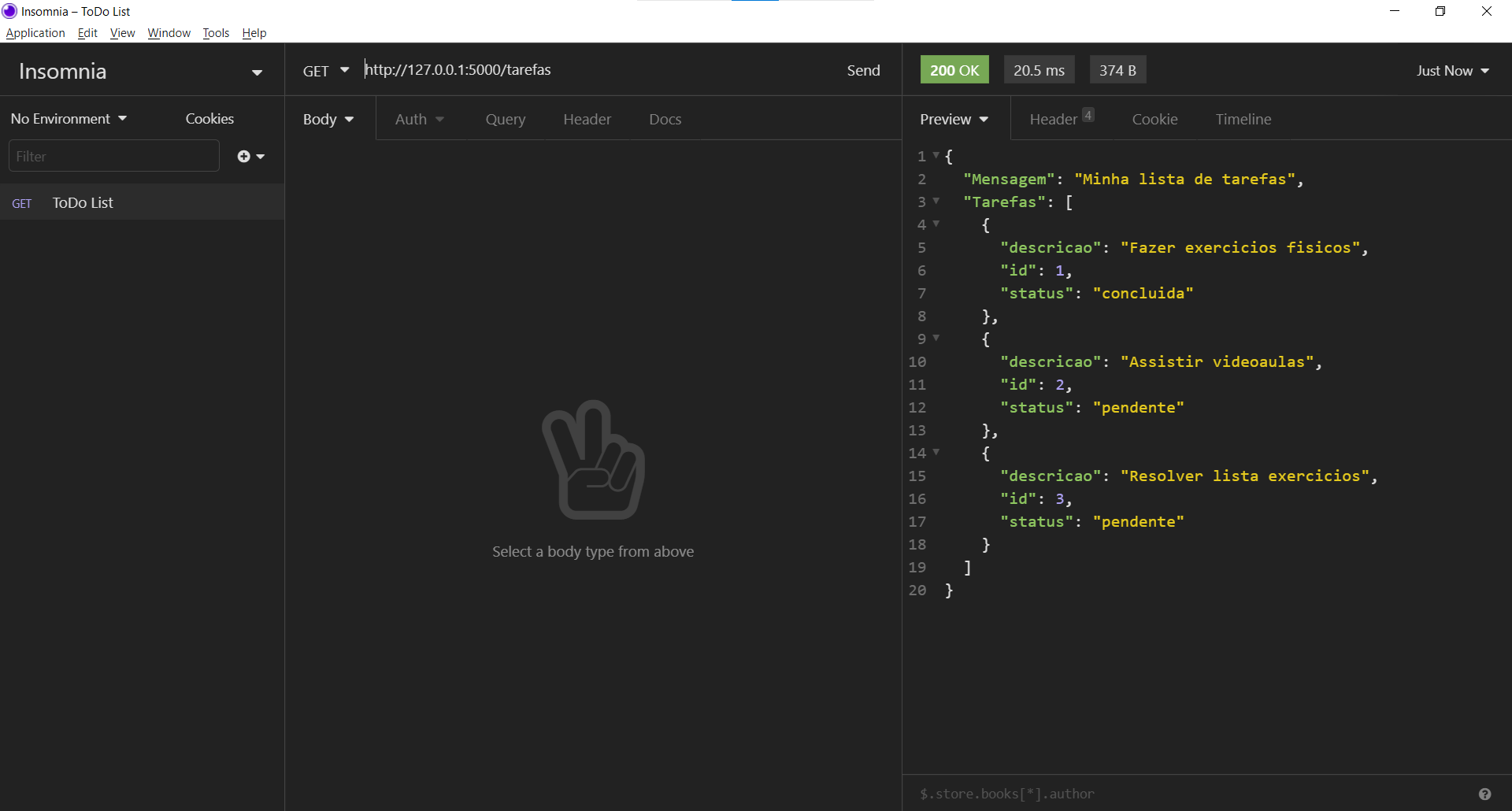
1. **Testando as rotas da API**

* URL de acesso aos serviços da aplicação:

http://127.0.0.1:5000

* GET (retornando todas as tarefas):

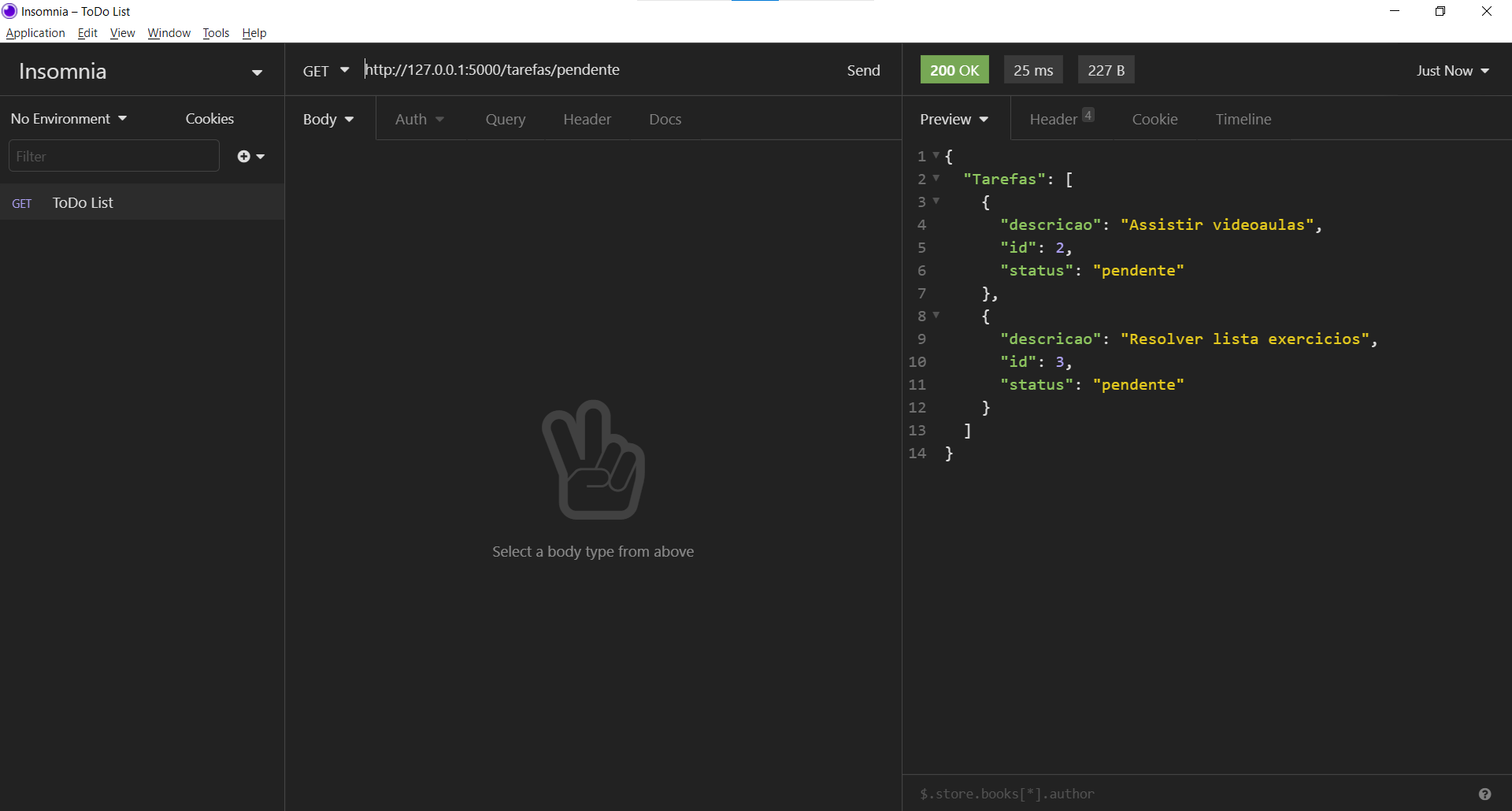
– URL: <http://127.0.0.1:5000/tarefas>.



* GET (retornando todas as tarefas de acordo com seu status):

– URL: <http://127.0.0.1:5000/tarefas/pendente>.

– Para acessar a URL de tarefas com status concluida, basta trocar o endereço no [Insomnia](https://insomnia.rest/download/core/?) para <http://127.0.0.1:5000/tarefas/pendente>.



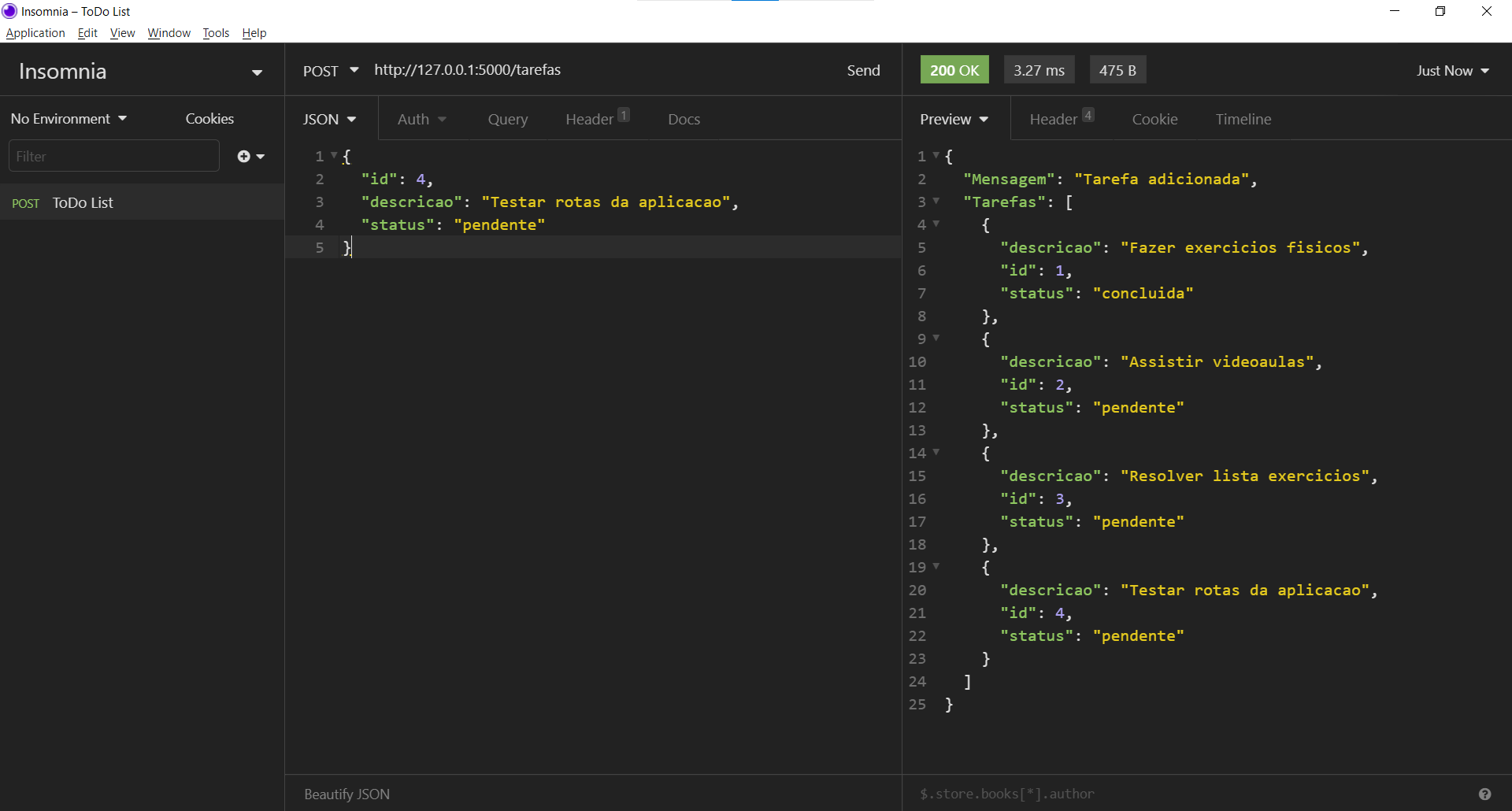
* POST (criando uma nova tarefa com um arquivo JSON):

– URL: <http://127.0.0.1:5000/tarefas>.

– Para realizar a criação de uma nova tarefa, é necessário trocar o tipo de requisição para POST, e em Body escolher a opção JSON e inserir as informações descritas abaixo.

– JSON:

|  |  |
| --- | --- |
| **1** | { |
| **2** | “id”: 4, |
| **3** | “descrição”: “Testar rotas da aplicação”, |
| **4** | “status”: “pendente” |
| **5** | } |



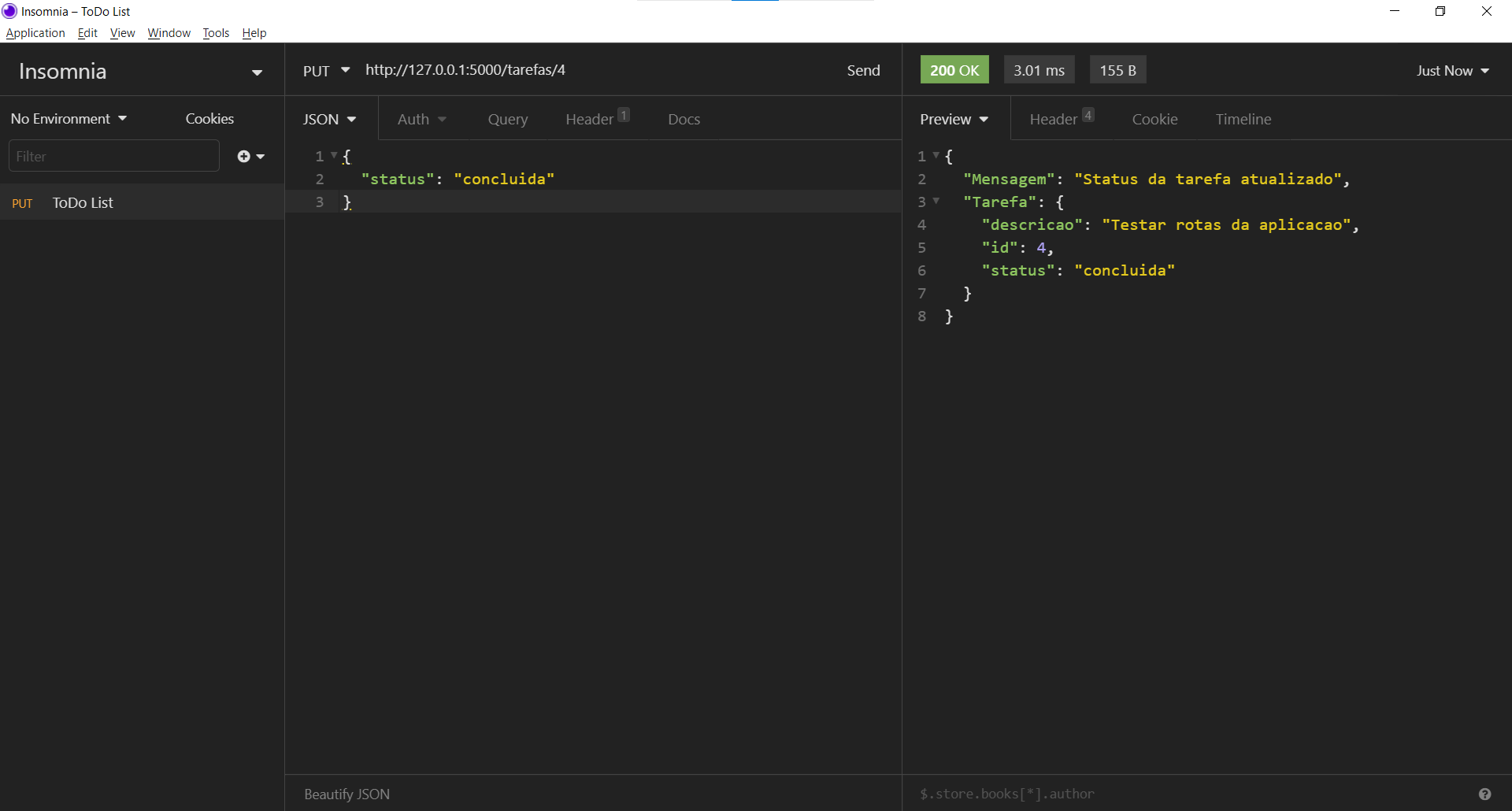
* PUT (atualizando o status de uma tarefa com um arquivo JSON):

– URL: <http://127.0.0.1:5000/tarefas/4>.

– Para este tipo de requisição, selecione o PUT e utilize o JSON a seguir.

– JSON:

|  |  |
| --- | --- |
| **1** | { |
| **2** | “status”: “concluida”, |
| **3** | } |



* DELETE (removendo uma tarefa):

– Para deletar um item da lista, escolha a requisição DELETE e a URL a seguir com o id da respectiva tarefa.

– URL: <http://127.0.0.1:5000/tarefas/4>.