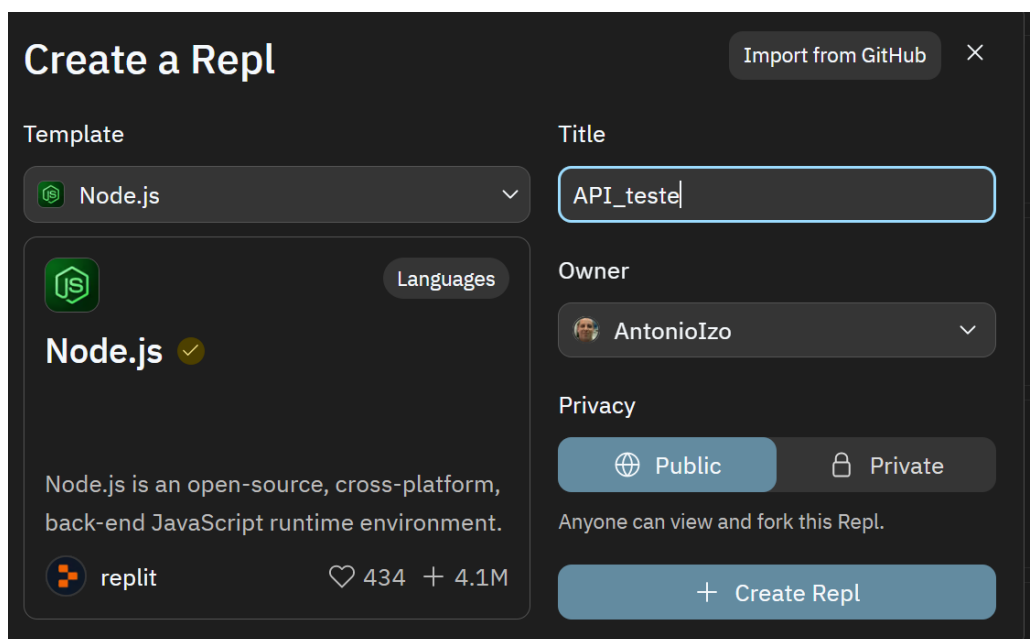


Utilizando o Postman com o Replit

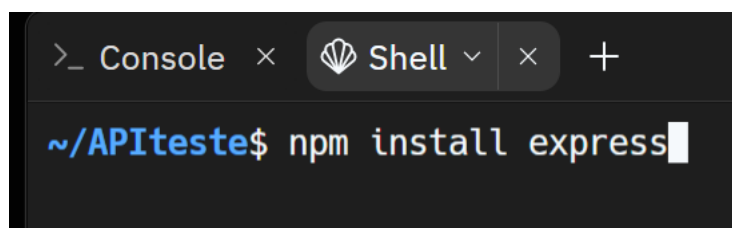
Após a instalação ou acesso do postman *online*, vamos criar nosso teste de API. Irei utilizar a versão da ferramenta web.

O primeiro passo é criar nossa API no Replit. Vamos começar criando um servidor básico e, em seguida, iremos aprimorá-lo conforme avançamos no desenvolvimento da API.

Abra o Replit e crie um novo projeto chamado "API_teste".



Vamos instalar nosso *express*.

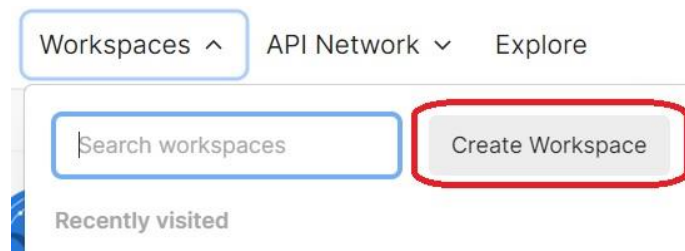


```
const express = require("express");
const app = express();
app.get("/", (req, res) => {
  return res.json({message: "Olá Mundo!"});
});
```

```
})  
app.listen(3000);
```

Repare que nosso projeto possui um servidor com apenas uma rota **app.get**.

Voltando ao Postman, crie um ambiente de trabalho (workspace). A workspace será o seu espaço para organizar o projeto.



Colocarei o nome de 'API_teste'.

Create workspace

Name

Summary

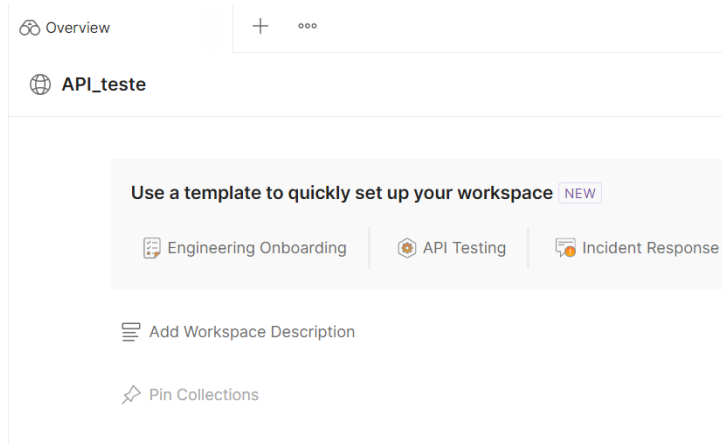
Add a brief summary about this workspace.

Visibility

Determines who can access this workspace.

- ☐ Personal
Only you can access
- ☐ Private
Only invited team members can access
- ☐ Team
All team members can access
- ☐ Partner NEW
Only invited partners and team members can access
- ☒ Public
Everyone can view

Importante que você deixe sua API com a opção 'public' marcada. Clique na opção "create workspace".

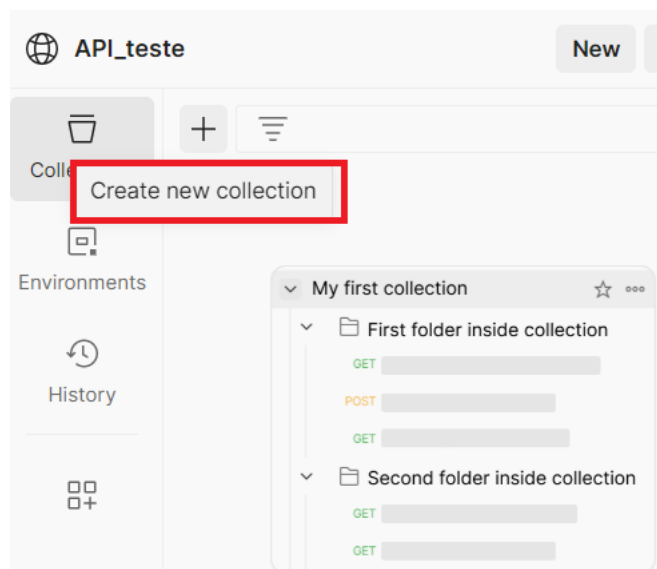


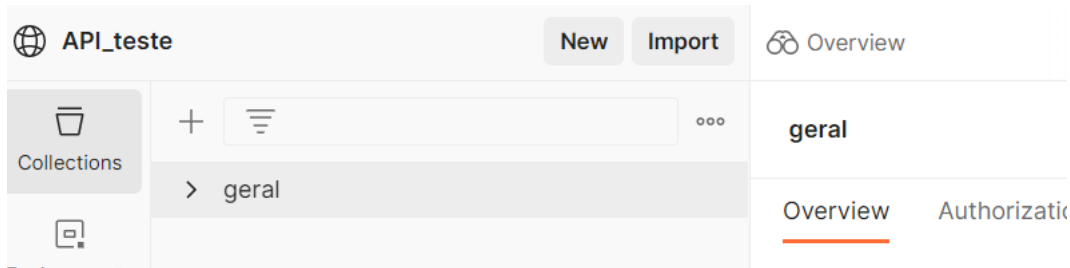
Agora vamos criar nossas coleções.

As coleções guardam todas as requisições para determinado objeto ou classe.

Na metodologia MVC, utilizamos os *controllers* para guardar o CRUD de cada objeto. No Postman faremos o mesmo.

Vamos criar uma coleção chamada 'geral'.

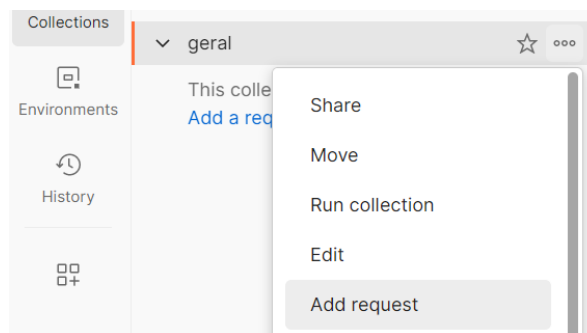




Crie um **request**.

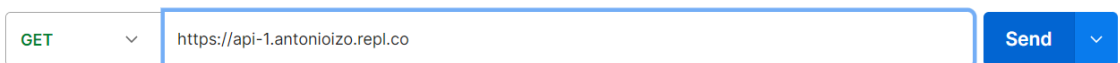
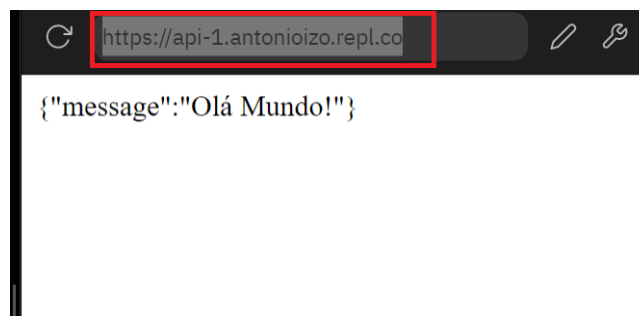
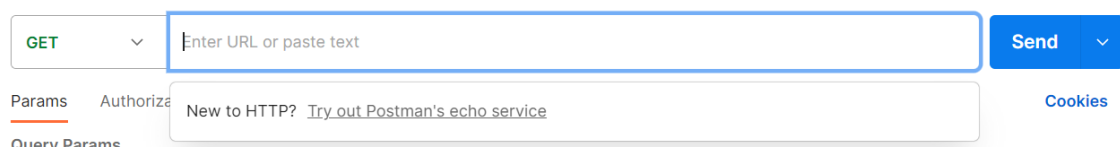
A *request* é a nossa requisição HTTP para o servidor. As requisições mais utilizadas são GET, POST, PUT e DELETE.

Vamos criar uma request do tipo GET.

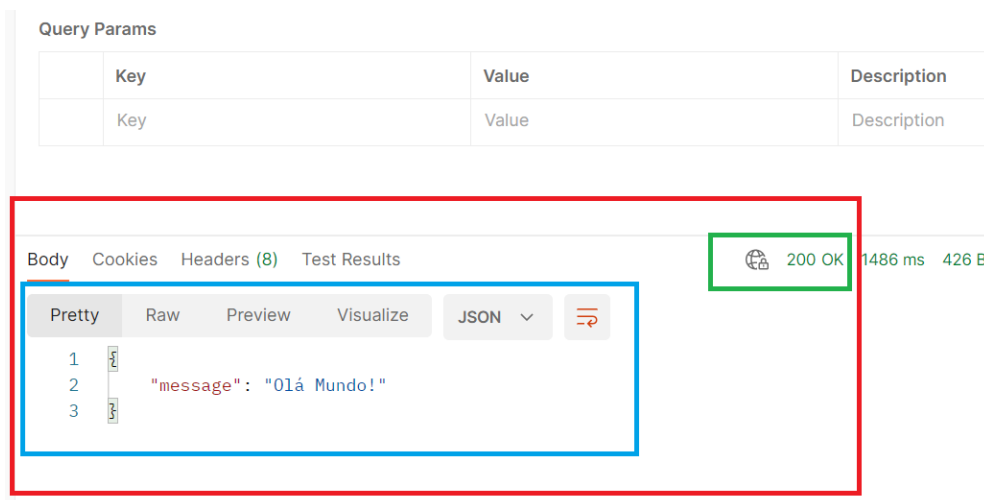


Na tela abaixo, temos que colocar o endereço da nossa API.

Atenção: Antes de executarmos o link da API_teste, precisamos iniciar o servidor para que ele possa estar em execução. O link que utilizamos será o do servidor em execução, e não o do código no Replit.



Se tudo ocorreu corretamente, você verá a imagem abaixo ao clicar em "Send".



O quadrado verde informa **200 ok**, ou seja, código 200 informa que a conexão foi bem sucedida.

O quadrado azul mostra nosso retorno da API. Assim finalizamos nosso teste de conexão com API.

Na próxima aula, você verá a criação do banco de dados e do *model* da API.

Até lá...