

Web API- ASP.NET Core

```
object to mirror_mod.mirror_object
operation == "MIRROR_X":
mirror_mod.use_x = True
mirror_mod.use_y = False
mirror_mod.use_z = False
operation == "MIRROR_Y":
mirror_mod.use_x = False
mirror_mod.use_y = True
mirror_mod.use_z = False
operation == "MIRROR_Z":
mirror_mod.use_x = False
mirror_mod.use_y = False
mirror_mod.use_z = True
```

```
selection at the end -add
mirror_ob.select= 1
modifier_ob.select=1
context.scene.objects.active
("Selected" + str(modifier_ob))
mirror_ob.select = 0
= bpy.context.selected_object
data.objects[one.name].select
print("please select exactly
```

--- OPERATOR CLASSES ---

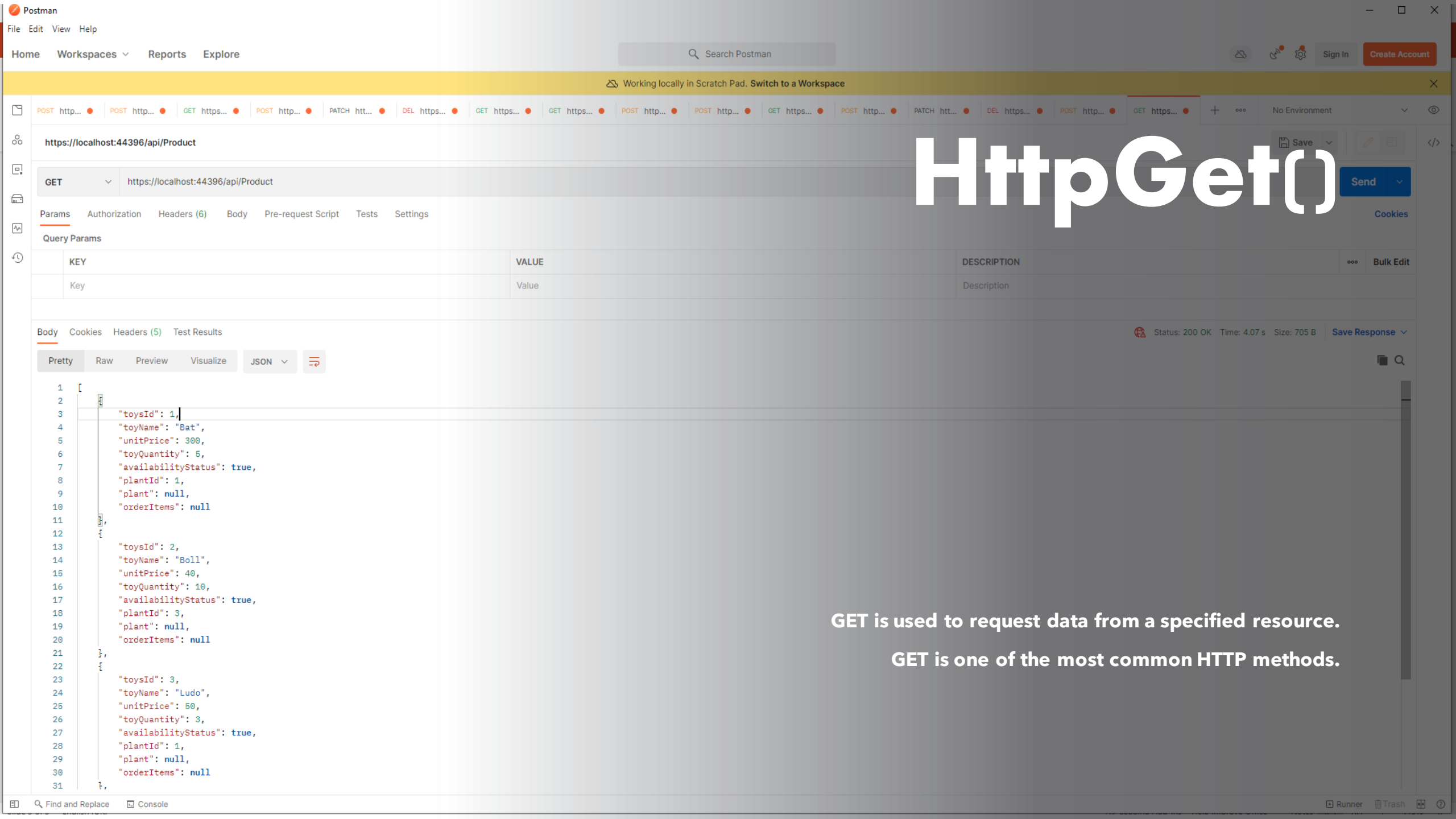
```
types.Operator):
X mirror to the selected
object.mirror_mirror_x"
mirror X"
```

CREATED BY :- Deep Parmar

Index

- HTTP Methods
- Repository Pattern
- Passing Headers

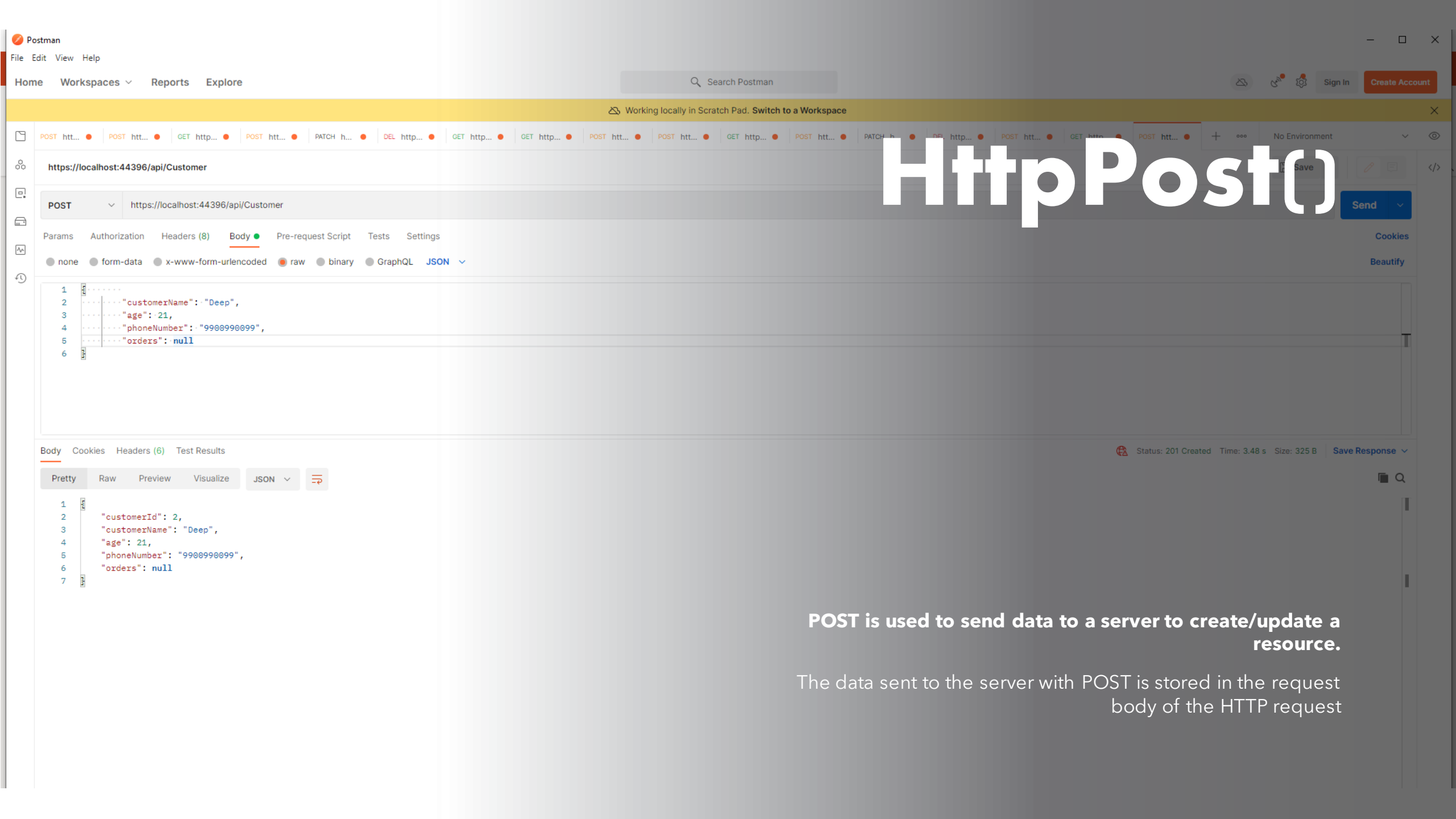




HttpGet()

GET is used to request data from a specified resource.

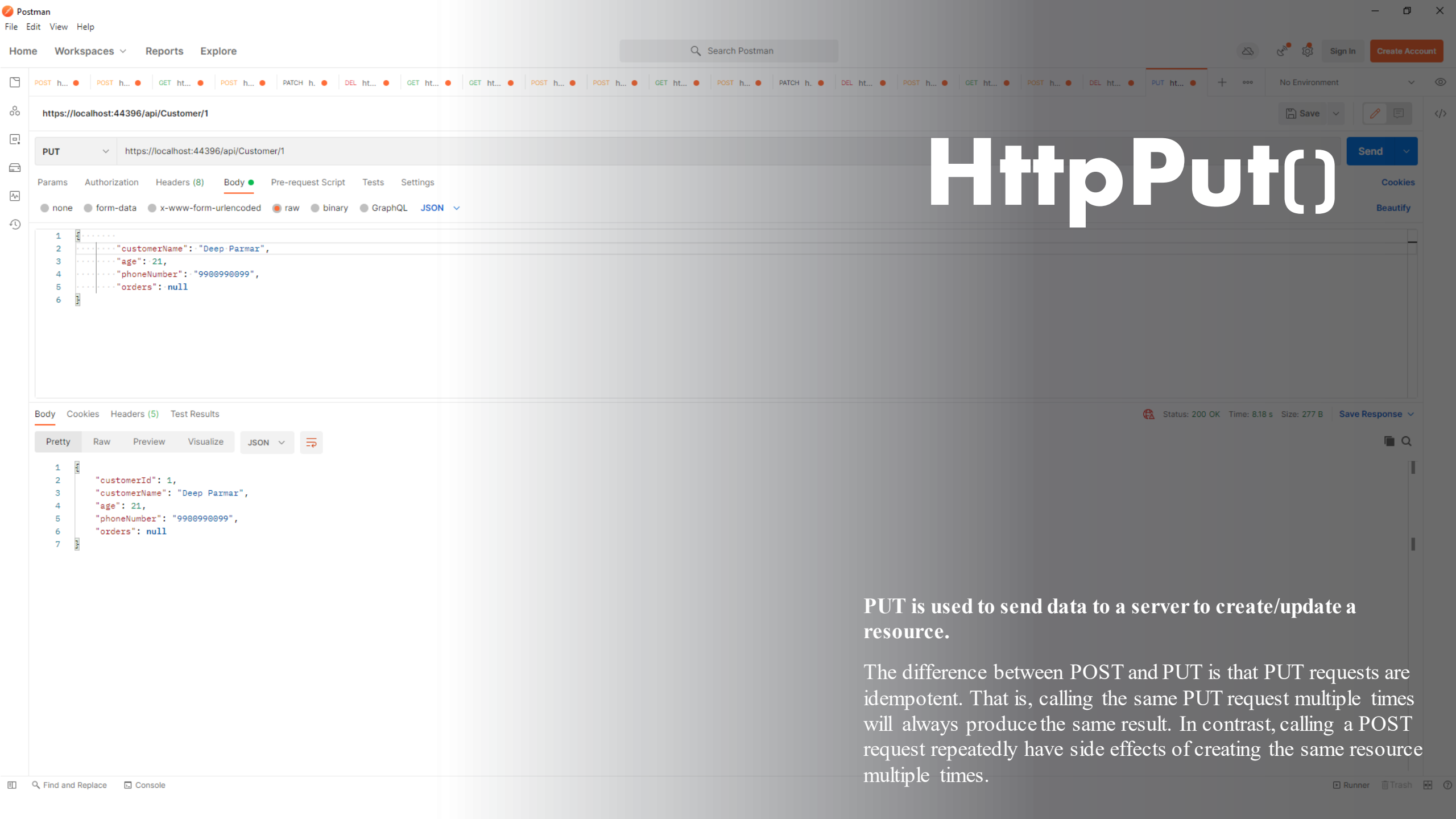
GET is one of the most common HTTP methods.



HttpPost()

POST is used to send data to a server to create/update a resource.

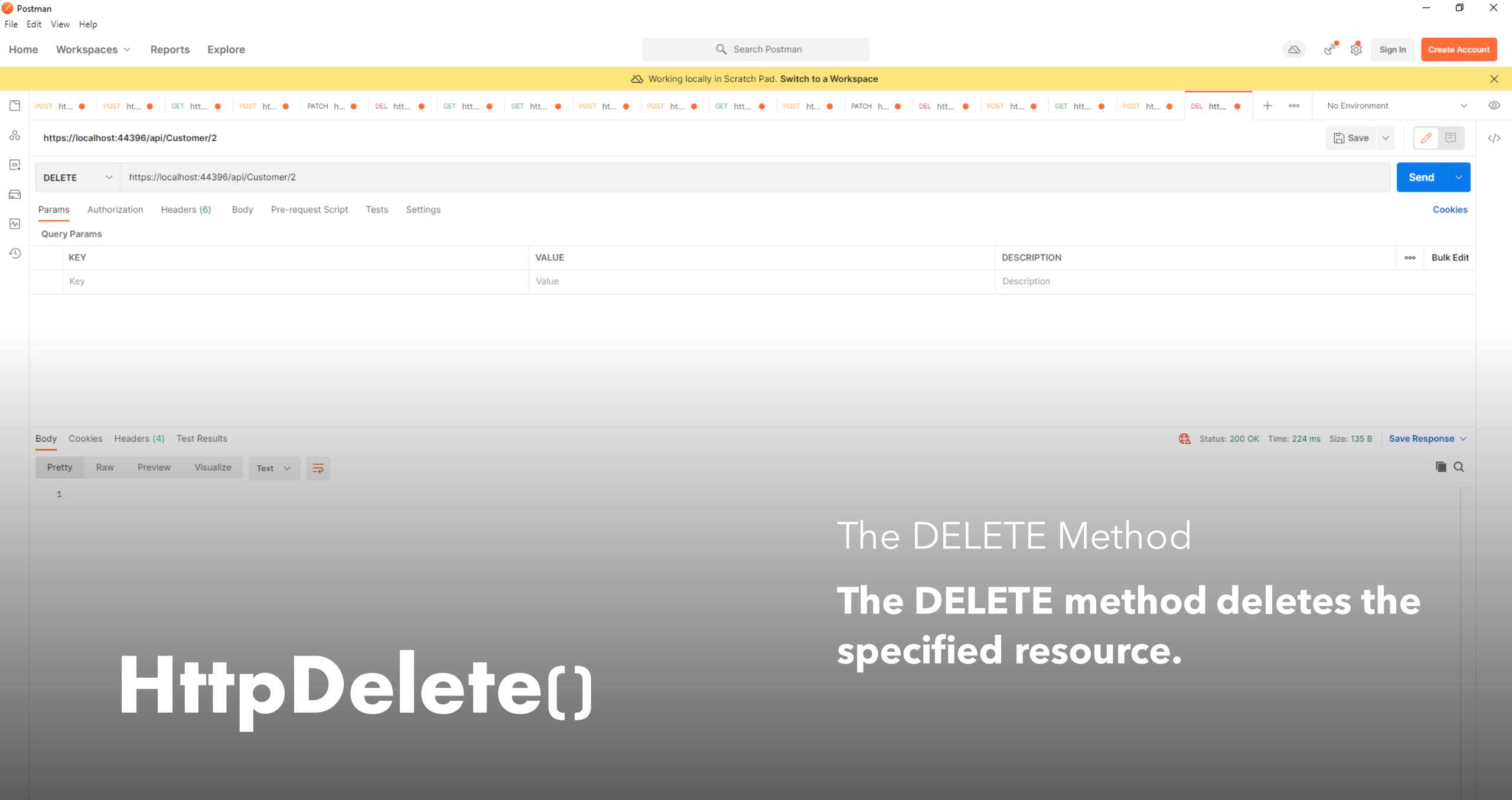
The data sent to the server with POST is stored in the request body of the HTTP request



HttpPut()

PUT is used to send data to a server to create/update a resource.

The difference between POST and PUT is that PUT requests are idempotent. That is, calling the same PUT request multiple times will always produce the same result. In contrast, calling a POST request repeatedly have side effects of creating the same resource multiple times.



HttpDelete()

The DELETE Method

The DELETE method deletes the specified resource.



Passing Headers

API headers are **like an extra source of information for each API call you make**. Their job is to represent the meta-data associated with an API request and response.

Repository Pattern



With the Repository pattern, we create an abstraction layer between the data access and the business logic layer of an application. By using it, we are promoting a more loosely coupled approach to access our data from the database. Also, the code is cleaner and easier to maintain and reuse. Data access logic is in a separate class, or sets of classes called a repository, with the responsibility of persisting the application's business model.