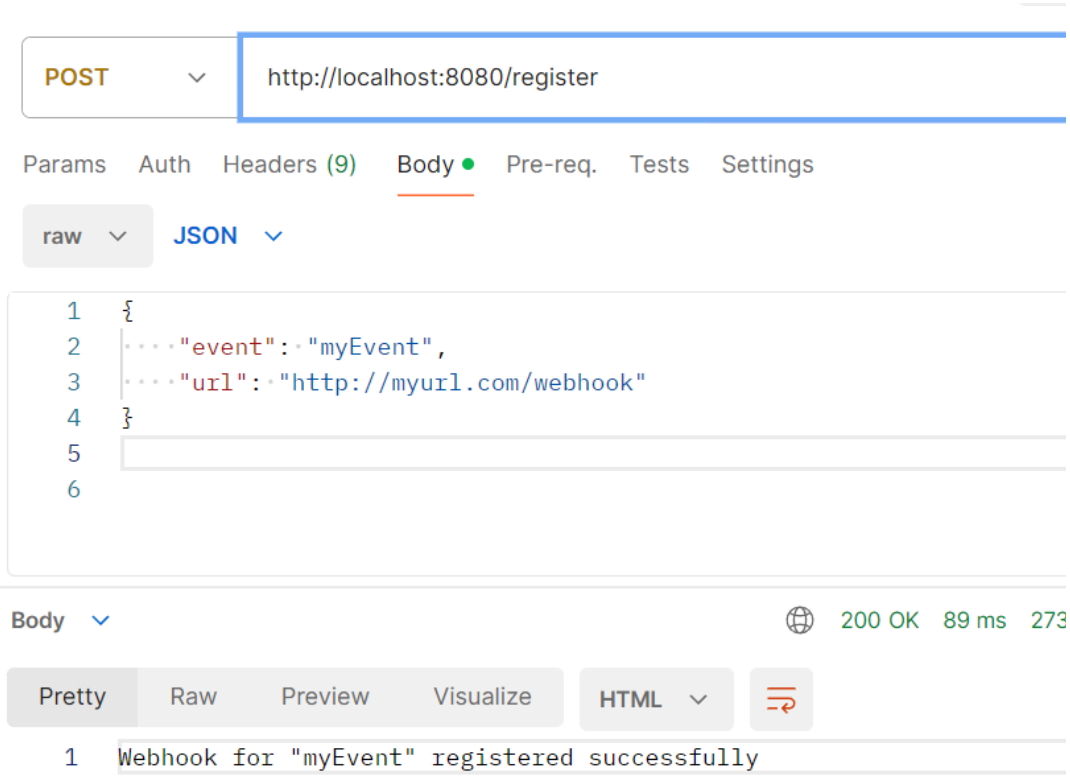
first understanding webhooks:

You have two apps, App A and App B.
App A has some data that App B needs, but App B doesn't know when that data changes.
So, App B sets up a webhook, which is just a URL that App A can send a message to.
App B tells App A about this webhook.
Now, whenever the data changes, App A sends a message (usually an HTTP POST request) to the webhook URL.
When App B receives this message, it knows that the data has changed and can take action accordingly.

the register mechanism:



To register a webhook, you need to send a POST request to the /register endpoint with the event name and the URL where you want to receive the webhook POST requests.

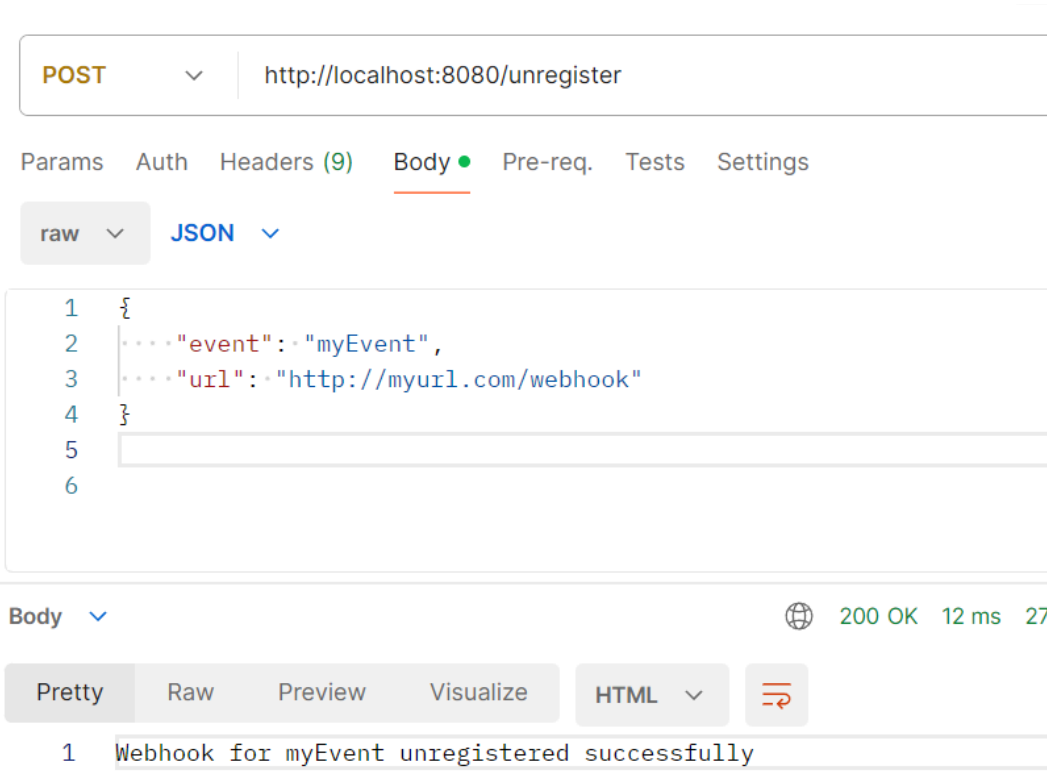
HTTP Request
POST /register

Request Body = json

```
{
  "event": "yourEventName",
  "url": "https://your-webhook-receiver.com/webhook"
}
```

event (required) - The name of the event you want to subscribe to.
url (required) - The URL that will receive the POST requests when the event occurs.

the unregister mechanism:



If you no longer want to receive webhook notifications, you can unregister your webhook.

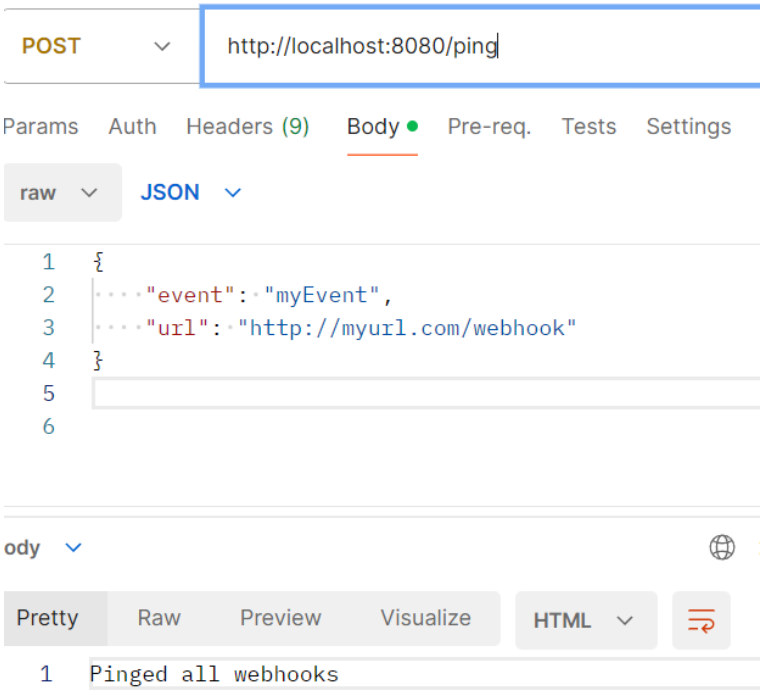
HTTP Request
POST /unregister

Request Body = json

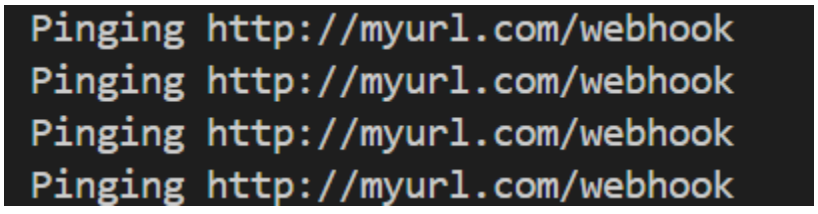
```
{
  "event": "yourEventName",
  "url": "https://your-webhook-receiver.com/webhook"
}
```

event (required) - The name of the event you are unsubscribed from.
url (required) - The URL that was receiving POST requests.

pinging the webhooks:



the result:



i have chosen to make an invoice system

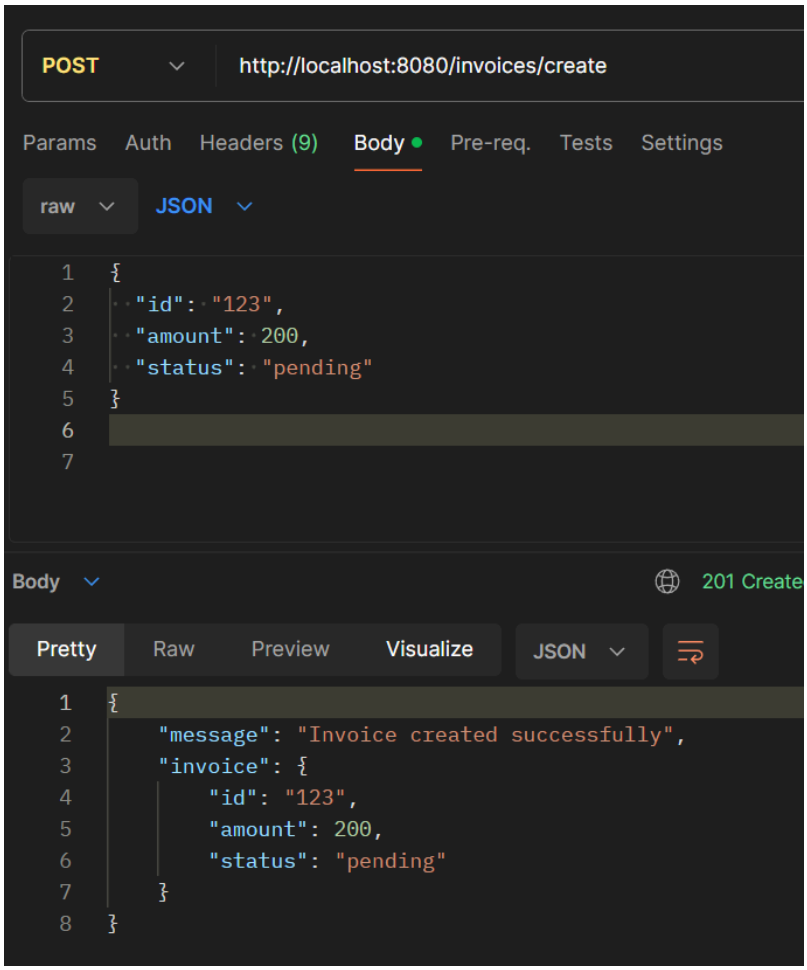
Event: Invoice Created (invoice_created)
Description
This event is triggered when a new invoice is generated in the system.

Webhook Payload

```
{
  "event": "invoice_created",
  "data": {
    "invoice_id": "123",
    "amount": 200,
    "status": "pending"
  }
}
```

Fields:

- invoice_id: Unique identifier for the invoice.
- amount: The total amount of the invoice.
- status: Current status of the invoice (e.g., "pending").



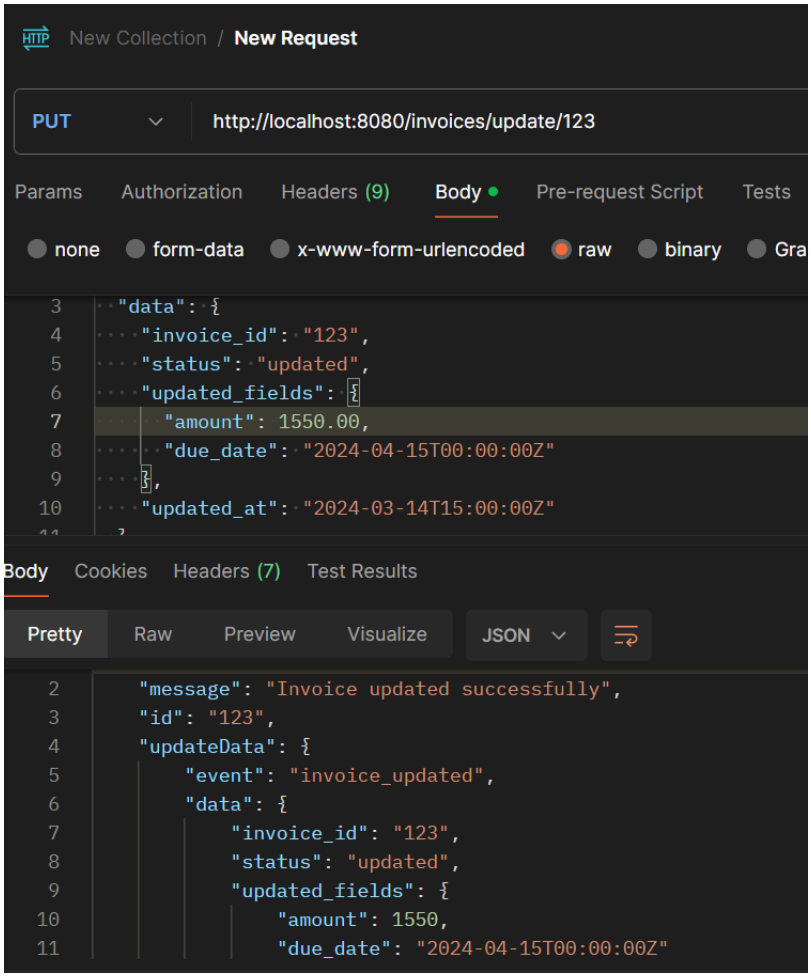
Event: Invoice Updated (invoice_updated)
Description
This event is triggered when details in the invoice are changed.

Webhook Payload

```
{
  "event": "invoice_updated",
  "data": {
    "invoice_id": "12345",
    "status": "updated",
    "updated_fields": {
      "amount": 1550.00,
      "due_date": "2024-04-15T00:00:00Z"
    },
    "updated_at": "2024-03-14T15:00:00Z"
  }
}
```

Fields:

- invoice_id: Unique identifier for the invoice.
- status: New status of the invoice (e.g., "updated").
- updated_fields: A collection of the fields that were updated with their new values.
- updated_at: Timestamp of when the invoice was updated.



Event: Invoice Paid (invoice_paid)
Description
Occurs when the invoice has been marked as paid.

Webhook Payload

```
{
  "event": "invoice_paid",
  "data": {
    "invoice_id": "12345",
```

```
    "status": "paid",
    "amount_received": 1550.00,
    "currency": "USD",
    "paid_at": "2024-03-20T10:00:00Z"
  }
}
```

Fields:

- invoice_id: Unique identifier for the invoice.
- status: New status of the invoice (e.g., "paid").
- amount_received: The total amount received for this invoice.
- currency: Currency in which the payment was made.
- paid_at: Timestamp of when the invoice was marked as paid.