1. Get Product Availability

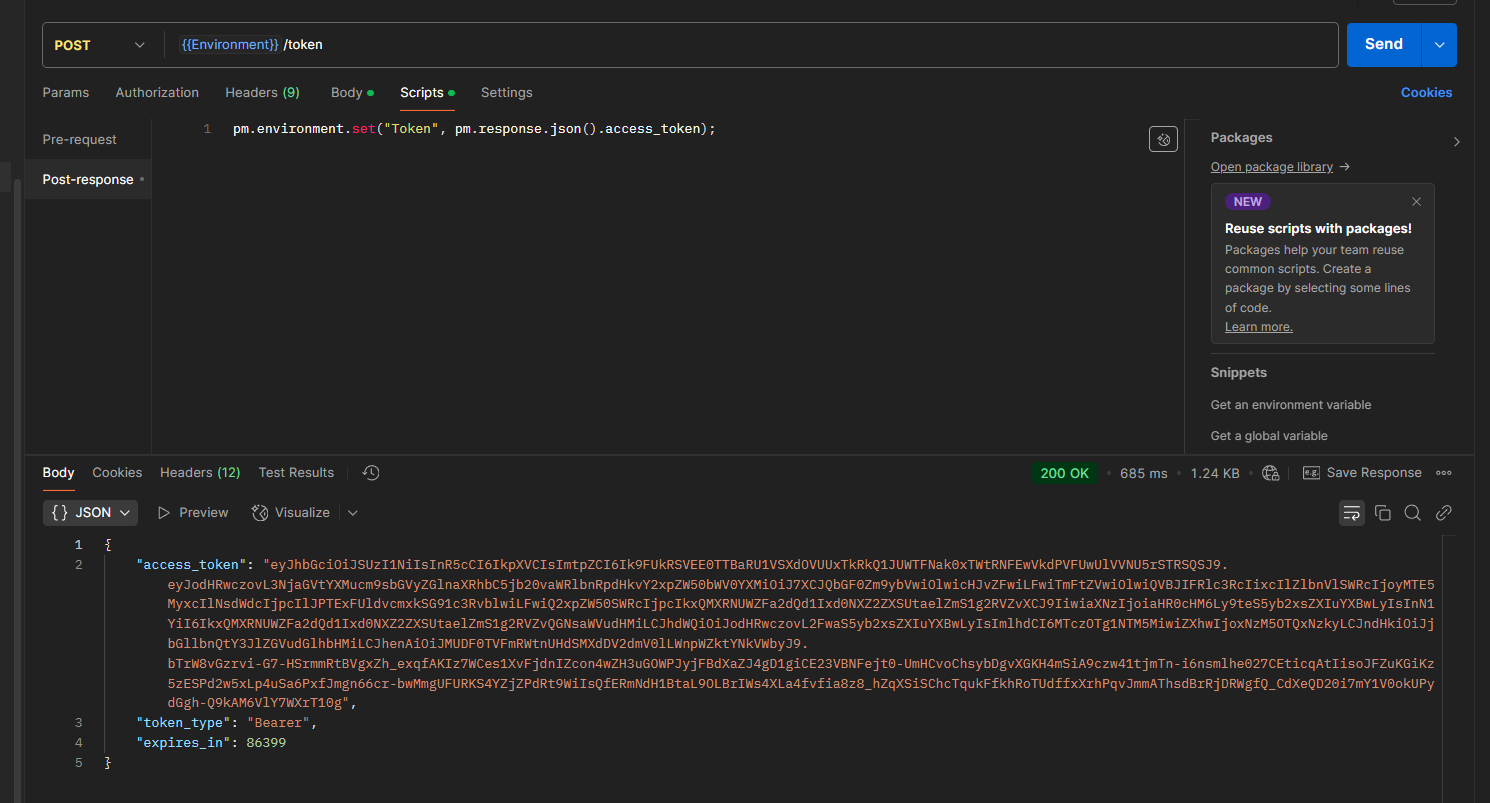
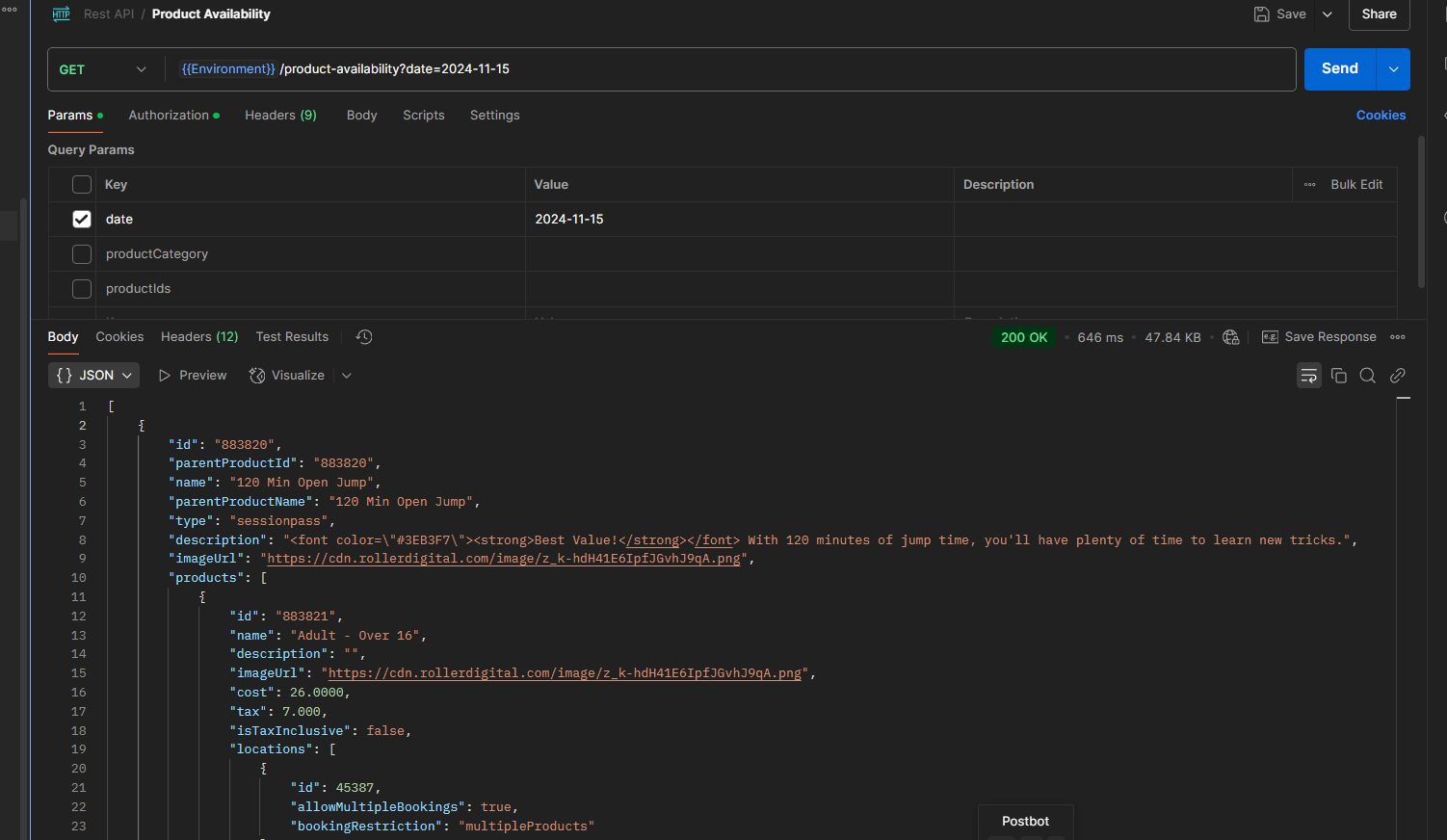
Here is a sample get request done to fetch product availability for November 15th 2024 done using Postman. The request collection was imported using the API docs site. After setting up the authorisation through the provided client id and client secret , the bearer token was automatically set to the variable through postman script. This request can also be done via curl command written here.

curl --request GET \

--url 'https://api.roller.app/product-availability?Date=2024-11-15' \

--header 'Accept: application/json' \

--header 'Authorization: Bearer token\_value\_here'



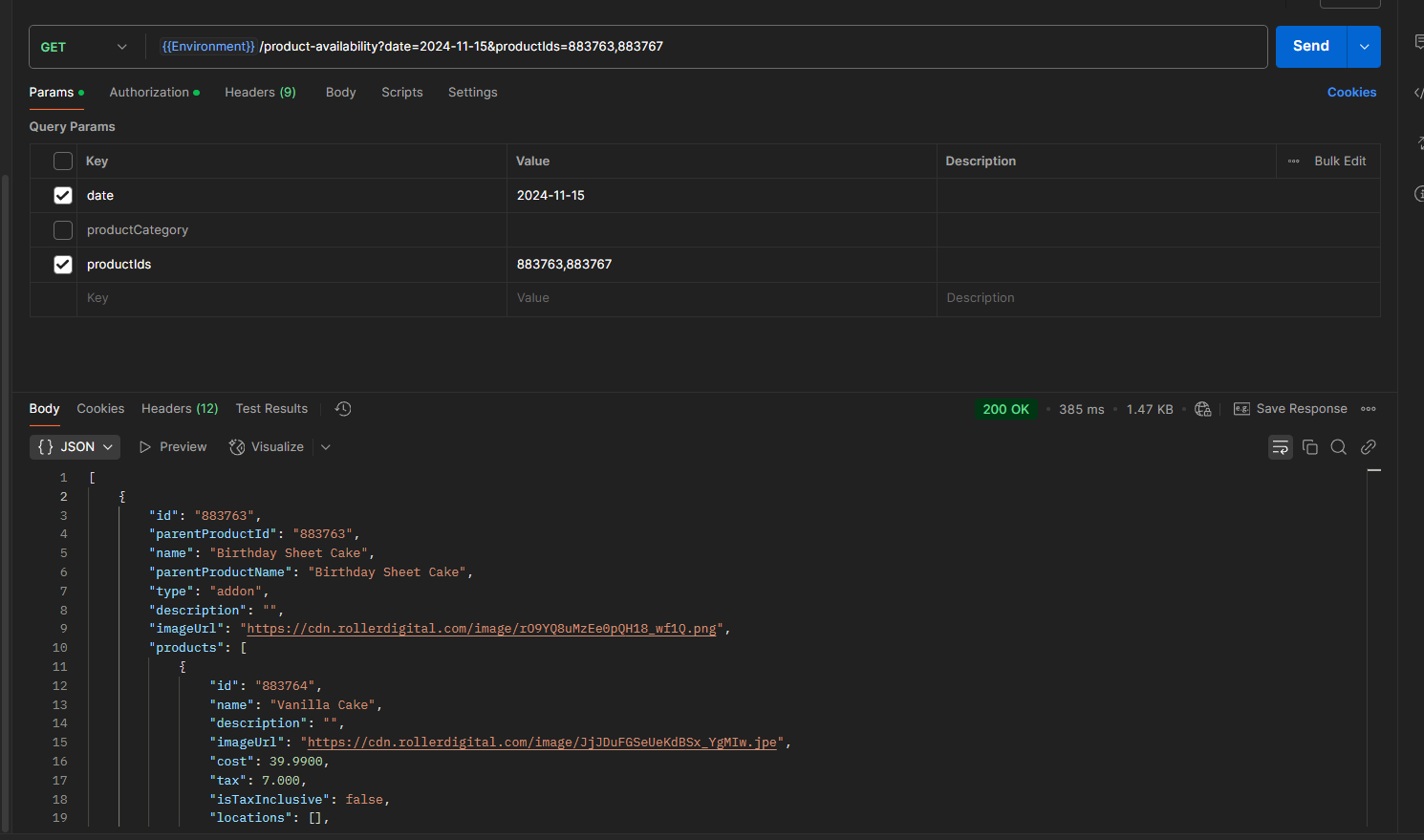
* The **Product Category** of **Parties and Merchandise** can be used, but it also fetches unrelated items.
* To retrieve only **birthday-related items** for the specified date:
  + Perform a **GET request** to the **Product Detail** endpoint.
  + Extract the **productId** related to **birthday products**.
* The **productId** can also be obtained by:
  + Performing a **GET request** to the **Data API - Get Modifiers** endpoint.
* This can be achieved using:
  + **Postman** (by providing parameters).
  + **curl command** (both methods yield the same result).

curl --request GET \

--url 'https://api.roller.app/product-availability?Date=2025-07-14&productIds=883763,883767' \

--header 'Accept: application/json' \

--header 'Authorization: Bearer token\_value\_here'



* **2.POST Booking Creation**
  + Use the POST /Create a Booking endpoint to Create a Booking for January 15th, 2025.
  + This should be for Party “Luke Finn’s Birthday” for 2 Bowling Lanes
  + They will pay via Credit Card and the party costs $500
  + What will be the expected response for this type of booking?

Post request body to do this booking-

Since there is no product id known for the bowling lanes after checking the get product endpoint, we use the locationid as a place holder for the two bowling lanes to showcase the post request body.

This post request was done under the assumption bowling was under the party package is shown here

{

  "externalId": "UJJSK28111",

  "name": "Luke Finn’s Birthday",

  "comments": "Bowling 2 lane",

  "purchaseDate": "2025-01-05T06:57:18.5306035+00:00",

  "capacityReservationId": "301d0d88-fd98-45e1-b1e7-17dbd1c28c92",

  "customerPaysFees": true,

  "customer": {

    "firstName": "Luke",

    "lastName": "Finn",

    "email": "LukeFinn@noemail.com",

    "phone": "123456789"

  },

  "items": [

{

      "productId": 883868,

      "quantity": 1,

      "bookingDate": "2025-03-15",

      "startTime": "16:00",

" priceOverride ": 500

      "tickets": [

        {

          "id": "UJJSK28111",

          "name": "Luke Finn"

        }

      ]

    },

     {

      "productId": 45405,

      "quantity": 1,

      "bookingDate": "2025-03-15",

"startTime": "16:00

    },

{

      "productId": 45406,

      "quantity": 1,

      "bookingDate": "2025-03-15",

      "startTime": "16:00"

    }

  ],

  "payments": [

    {

      "id": "HYA728830013",

      "paymentType": "CreditCard",

      "amount": 500

    }

  ]

}

Post request body Providing only the necessary fields without the party package assumption-

{

  "externalId": "UJJSK28111",

  "name": "Luke Finn’s Birthday",

  "comments": "Bowling 2 lane",

  "purchaseDate": "2025-01-05T06:57:18.5306035+00:00",

  "capacityReservationId": "301d0d88-fd98-45e1-b1e7-17dbd1c28c92",

  "customerPaysFees": true,

  "customer": {

    "firstName": "Luke",

    "lastName": "Finn",

    "email": "LukeFinn@noemail.com",

    "phone": "123456789"

  },

  "items": [

     {

      "productId": 45405,

      "quantity": 1,

      "bookingDate": "2025-03-15",

      "startTime": "16:00"

    },

{

      "productId": 45406,

      "quantity": 1,

      "bookingDate": "2025-03-15",

      "startTime": "16:00"

    }

  ],

  "payments": [

    {

      "id": "HYA728830013",

      "paymentType": "CreditCard",

      "amount": 500

    }

  ]

}

Curl command-

curl -X POST "https://api.roller.app/bookings" \

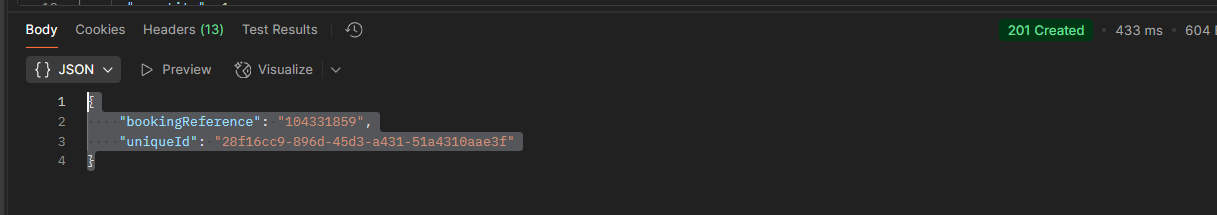
-H "Content-Type: application/json" \

-H "Authorization: Bearer YOUR\_ACCESS\_TOKEN" \

-d '{ "externalId": "UJJSK28111", "name": "Luke Finn’s Birthday", "comments": "Bowling 2 lane", "purchaseDate": "2025-01-05T06:57:18.5306035+00:00", "capacityReservationId": "301d0d88-fd98-45e1-b1e7-17dbd1c28c92", "customerPaysFees": true, "customer": { "firstName": "Luke", "lastName": "Finn", "email": "LukeFinn@noemail.com", "phone": "123456789" }, "items": [ { "productId": 45405, "quantity": 1, "bookingDate": "2025-03-15", "startTime": "16:00" }, { "productId": 45406, "quantity": 1, "bookingDate": "2025-03-15", "startTime": "16:00" } ], "payments": [ { "id": "HYA728830013", "paymentType": "CreditCard", "amount": 500 } ] }

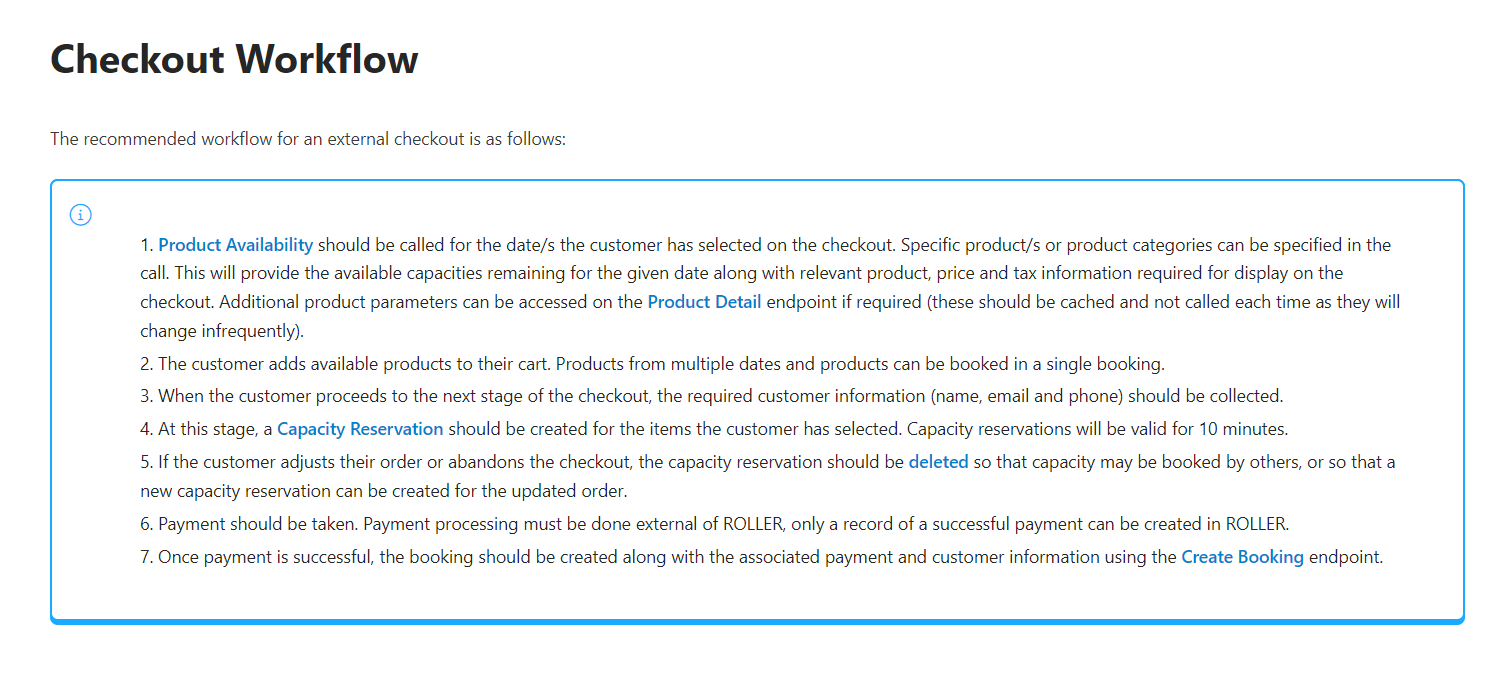
* Followed the **checkout workflow recommendation** by performing a **GET** request on **Product Availability** for the selected date (**January 15, 2025**).
* Observed that **no explicit details** about the **bowling alley** were mentioned in **Product Availability**.
* To confirm, performed a **GET** request on **Product Detail** to find its **productId**.
* The **Product Detail response** also did not explicitly mention the **bowling alley**.
* Further verification was done by searching for the **location ID** of the bowling alley and bowling lanes using the **GET Locations** endpoint.
* Found that the **bowling alley location ID** was not present in **Get Products, Get Product Availability, or Get Product Detail**.
* Had to assume that booking the party package was the correct approach since the bowling alley product ID was unknown and not returned in the **Product Data API**.
* The **product ID** is necessary to perform a **POST Capacity Reservation** before making a **POST Create a Booking** request.
* Without the **capacityReservationId**, the booking **might not be confirmed**.
* Will provide **two POST Create a Booking requests**:
  + **Ideal booking request**, assuming the **product ID is known**.
  + Booking under the assumption of a party package, considering it follows the same workflow
* As mentioned , the name of the booking was done for Luke finn’s Birthday and the payment was done for 500 through credit card
* The request can be sent via **Postman or Bash**, but **Postman** is preferred as it:
  + Provides a **formatted request/response view**.
  + Enables **test automation** for **regression test suites**.

**Expected Response: We expect a success message of 200 series particularly a 201 created to indicate the booking was created and response includes the booking reference and a unique id**

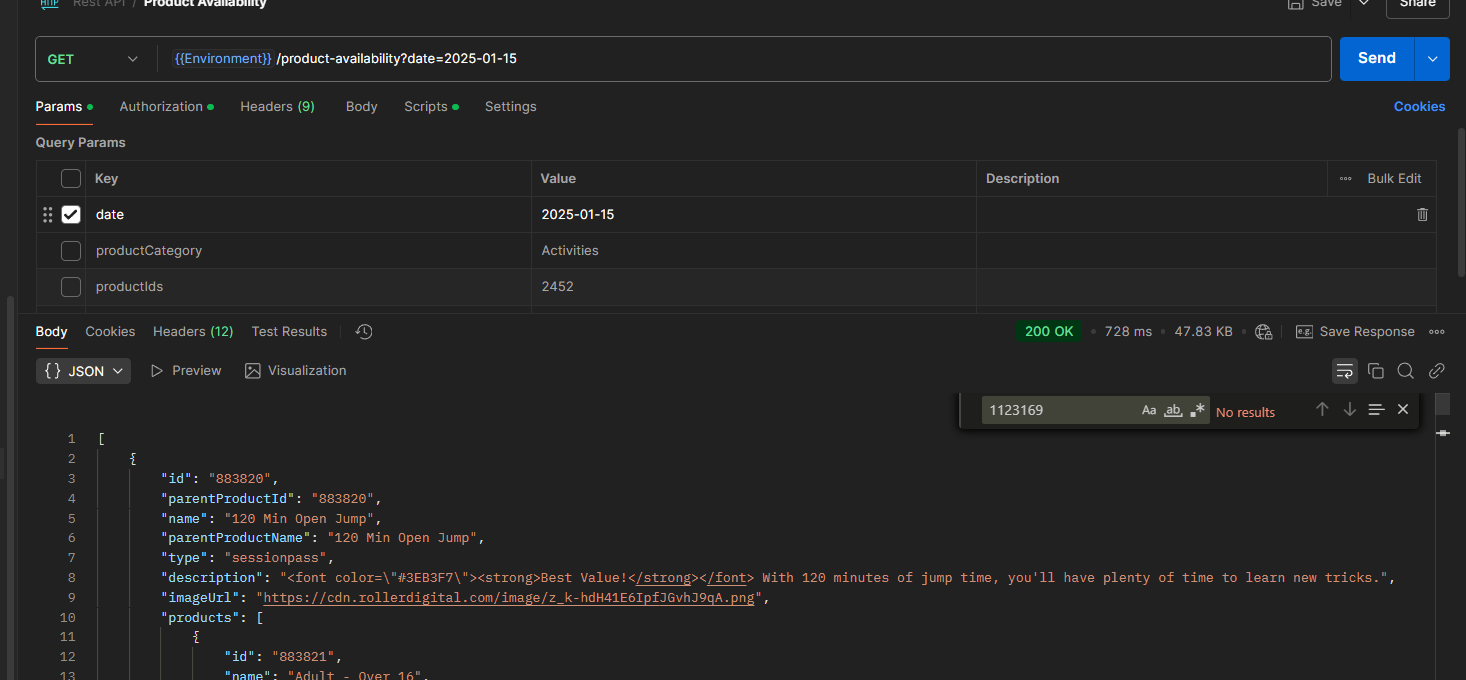
****

Steps to create the Post:

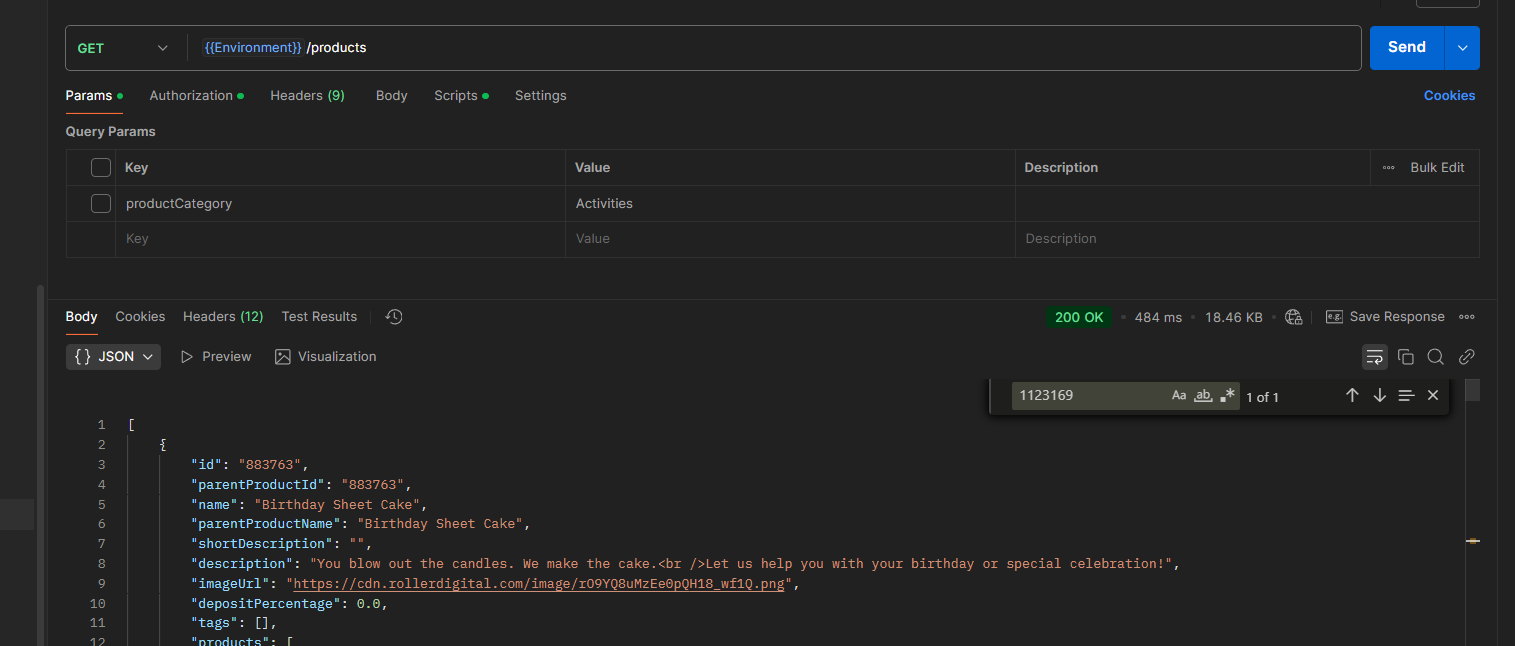
* Get Product availability to get the available products on the selected dates
* Post Capacity Reservation and get the capacityReservationId for the selected Product
* Complete the payment to get the payment details
* Complete the Post Create a booking to complete the workflow



The get Product availability response for that date is shown here which doesn’t have any bowling lane product details. [get product availibility.txt](get%20product%20availibility.txt)



Similary get product detail was done to fetch the bowling alley product id .



Next I do a Post Capacity reservation to block capacity till payment is completed. Here is the post body used for that

{

  "items": [

    {

      "productId": 883868,

      "quantity": 1,

      "bookingDate": "2025-03-15",

      "startTime": "16:00",

      "tickets": [

        {

          "id": "UJJSK28111",

          "name": "Luke Finn"

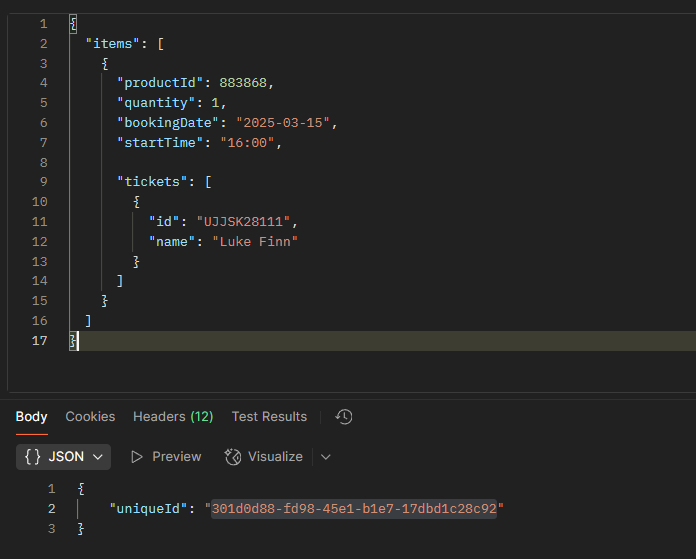
        }

      ]

    }

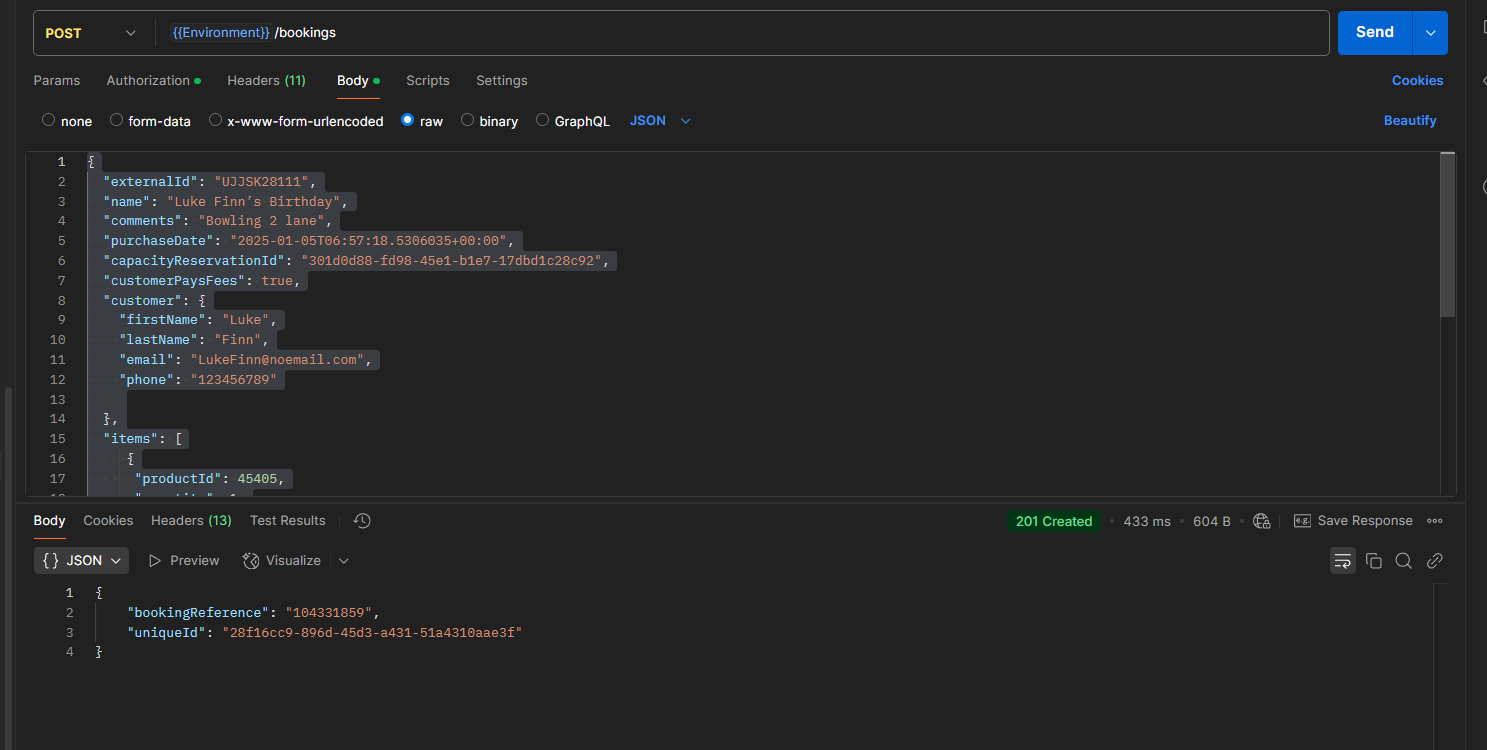
  ]

}



Through this we get the uniqueid needed to complete the post create a booking. Here the request is done for the Ultimate Party Package.

The post is done as shown here



**3. PUT Booking Amendment**

* Update a Luke Finn’s Birthday Party booking using PUT /Update a Booking.
  + The order should include:
    - Change the party to 4 lanes
    - Add 5 extra pizzas to the party for $15 a pizza
* The requirement is to update this booking to 4 lanes and add 5 pizzas
* From docs we see we need uniqueid as path parameters to locate the booking to update.
* From docs we see we can simply add two more lanes under the newItems for the two more lanes or add all 4 lanes here and remove the previous lanes through the idsToRemove option <https://docs.roller.app/docs/rest-api/v4mzj4t4erwa9-update-a-booking#request-body>

JSON Body of Request

{

  "name": "Luke Finn's Birthday Party",

  "comments": "Bowling alley along with birthday party",

  "newItems": [

    {

      "productId": 883789,

      "quantity": 5,

      "bookingDate": "2025-01-15",

      "startTime": "16:00",

      "priceOverride": 15,

      "packageInclusions": [

        {

          "productId": "883868",

          "quantity": 1

        }

      ]

    },

    {

      "productId": 45407,

      "quantity": 1,

      "bookingDate": "2025-01-15",

      "startTime": "16:00"

    },

    {

      "productId": 45408,

      "quantity": 1,

      "bookingDate": "2025-01-15",

      "startTime": "16:00"

    },

    {

      "productId": 45409,

      "quantity": 1,

      "bookingDate": "2025-01-15",

      "startTime": "16:00"

    },

    {

      "productId": 45410,

      "quantity": 1,

      "bookingDate": "2025-01-15",

      "startTime": "16:00"

    }

  ],

      "tickets": [

        {

          "id": "UJJSK28111",

          "name": "Luke Finn"

        }

      ],

  "idsToRemove": [45405, 45406]

}

* We add the productid of pizza and set quantity to 5 and price override it to $15 as specified.
* We also add the 4 lanes under new items and remove the previous two lanes previously booked
* The other non-mandatory fields weren’t given as it wasn’t necessary for the success of the request

Curl command-

curl -X PUT [https://api.roller.app/bookings/{bookingId}](https://api.roller.app/bookings/%7bbookingId%7d)

-H "Content-Type: application/json"

-H "Authorization: Bearer YOUR\_ACCESS\_TOKEN"

-d '{"name": "Luke Finn'\''s Birthday Party","comments": "Bowling alley along with birthday party","newItems": [{"productId": 883789,"quantity": 5,"bookingDate": "2025-01-15","startTime": "16:00","priceOverride": 15,"packageInclusions": [{"productId": "883868","quantity": 1}]},{"productId": 45407,"quantity": 1,"bookingDate": "2025-01-15","startTime": "16:00"},{"productId": 45408,"quantity": 1,"bookingDate": "2025-01-15","startTime": "16:00"},{"productId": 45409,"quantity": 1,"bookingDate": "2025-01-15","startTime": "16:00"},{"productId": 45410,"quantity": 1,"bookingDate": "2025-01-15","startTime": "16:00"}],"tickets": [{"id": "UJJSK28111","name": "Luke Finn"}],"idsToRemove": [45405, 45406]}'

**4. Debug Post Request**

* Review the following sample POST request and note what errors might be causing the request to fail

curl --request GET \

--url https://api.roller.app/bookings/{uniqueIdentifier}/payments \

--header 'Accept: application/json' \

--header 'Authorization: Bearer 123' \

--header 'Content-Type: application/json' \

--data '{

"paymentType": "GiftCard",

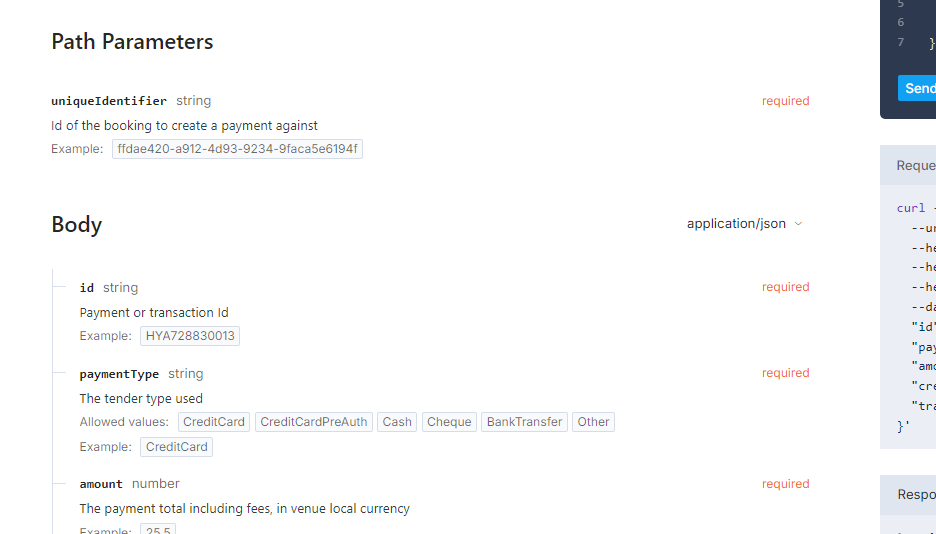
"amount": 25.5,

"creditCardFees": 1.5,

"transactionDate": "2020-07-32T06:57:18.5326035+00:00"

}'

**Causes:**

* The request is using GET instead of POST which is the incorrect method and would cause an error cause get request doesn’t contain a body
* We are missing the id parameter in the body which is a required parameter according to the docs****

The date has an invalid date of 32 July which doesn’t exist and will cause an error