Due Date: Monday July 20 12pm PST.  
  
Task 1: Add a REST API endpoint to allow a user to add a rating to a beer.

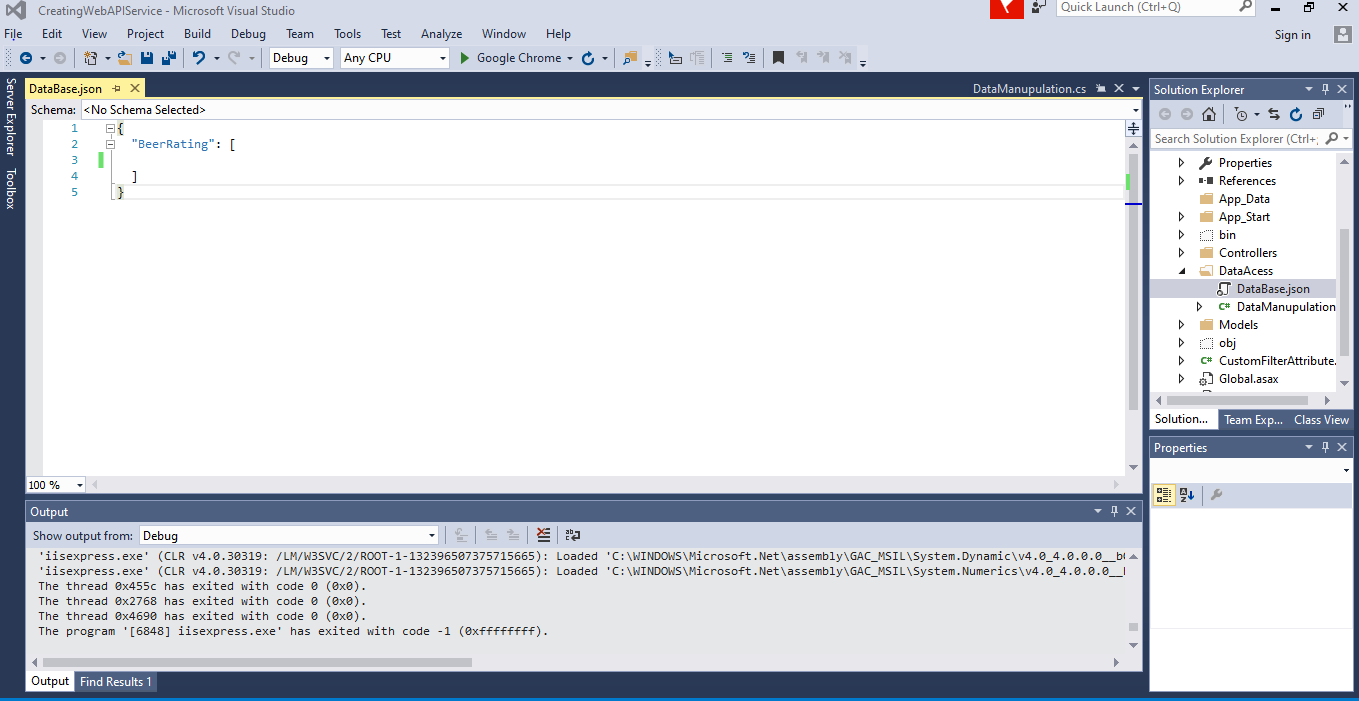
URL: <http://localhost:56290/api/Beer/AddBeerInfo>

Type: GET

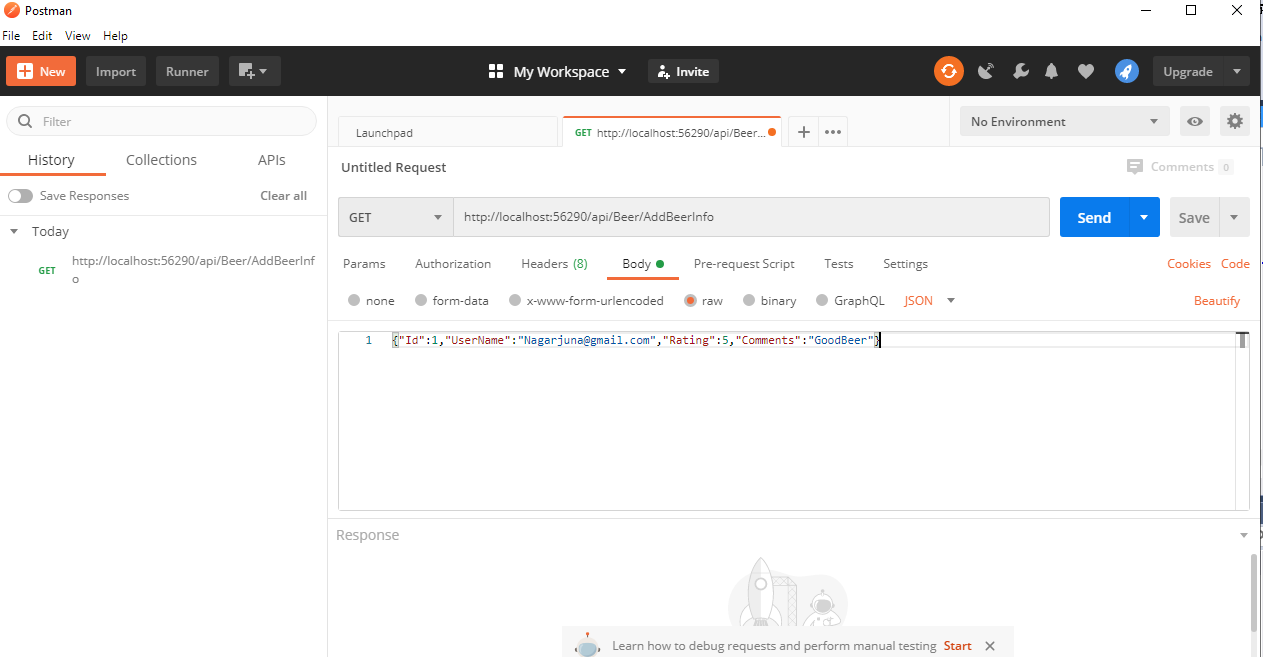
InputData: {"Id":1,"UserName":"Nagarjuna@gmail.com","Rating":5,"Comments":"GoodBeer"}

OutPut: "{\"Id\":1,\"UserName\":\"Nagarjuna@gmail.com\",\"Rating\":5,\"Comments\":\"GoodBeer\",\"BeeName\":\"Buzz\",\"Status\":\"Sucess\",\"ReultData\":null}"

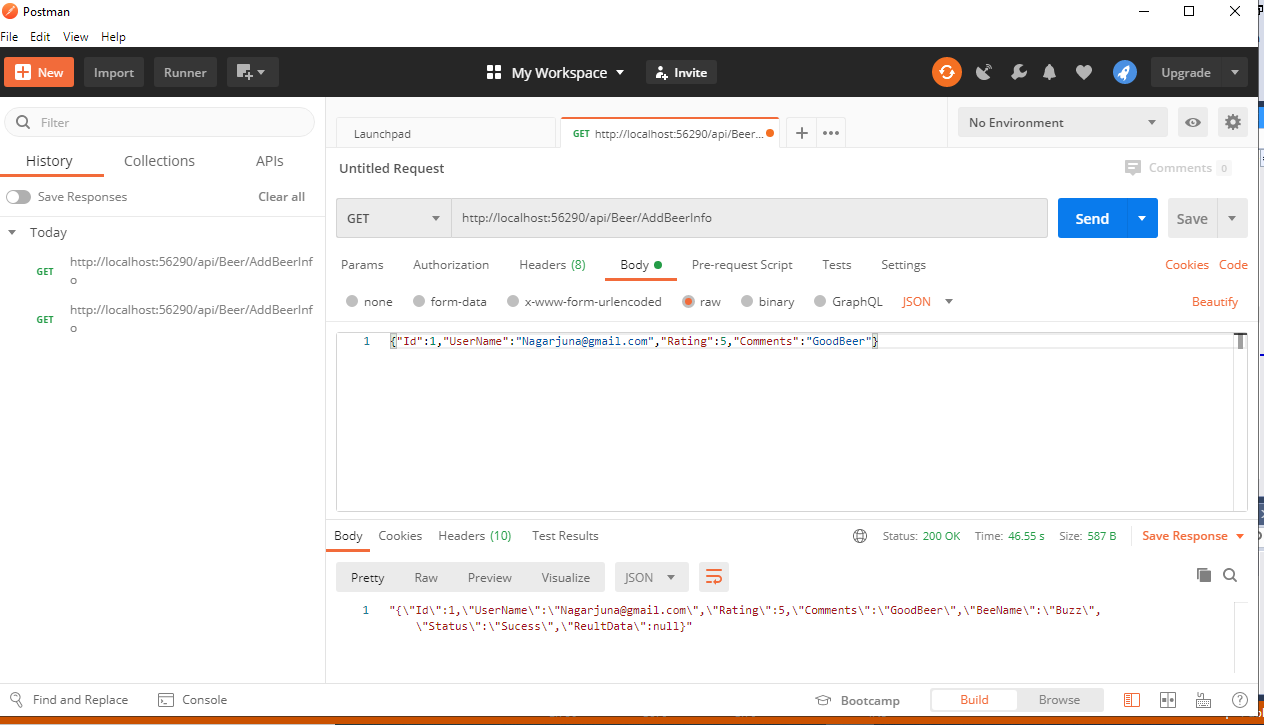
Step: 1 - As you Can see below in picture there is no rating review in DataBase.json



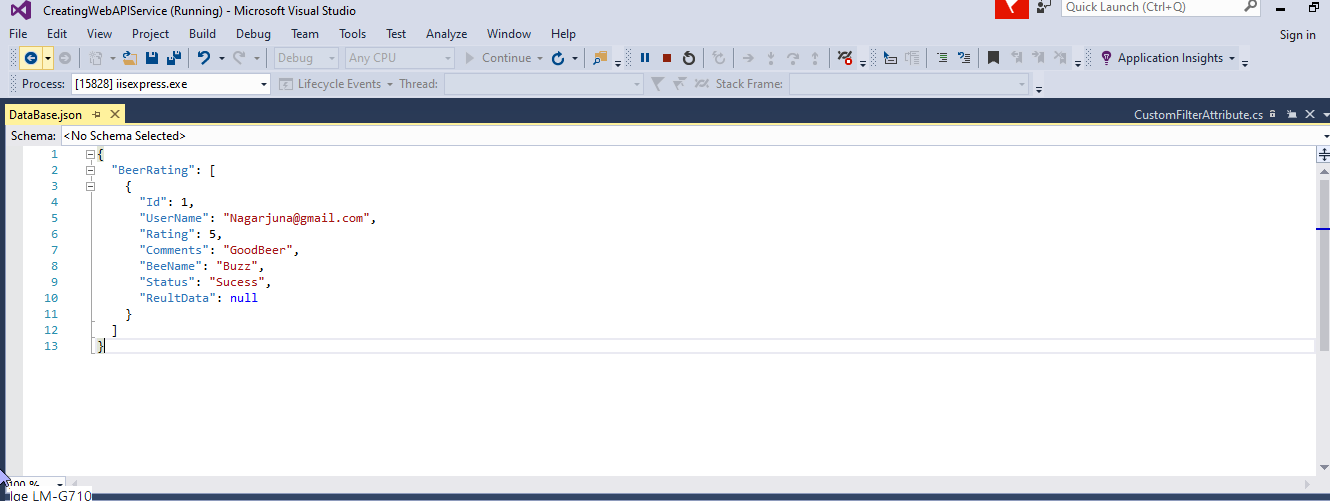
Input: Json code below “Id”:1, “UserName”: “Nagarjuna@gmail.com”, “Rating”:5 “comments”: “GoodBeer”.



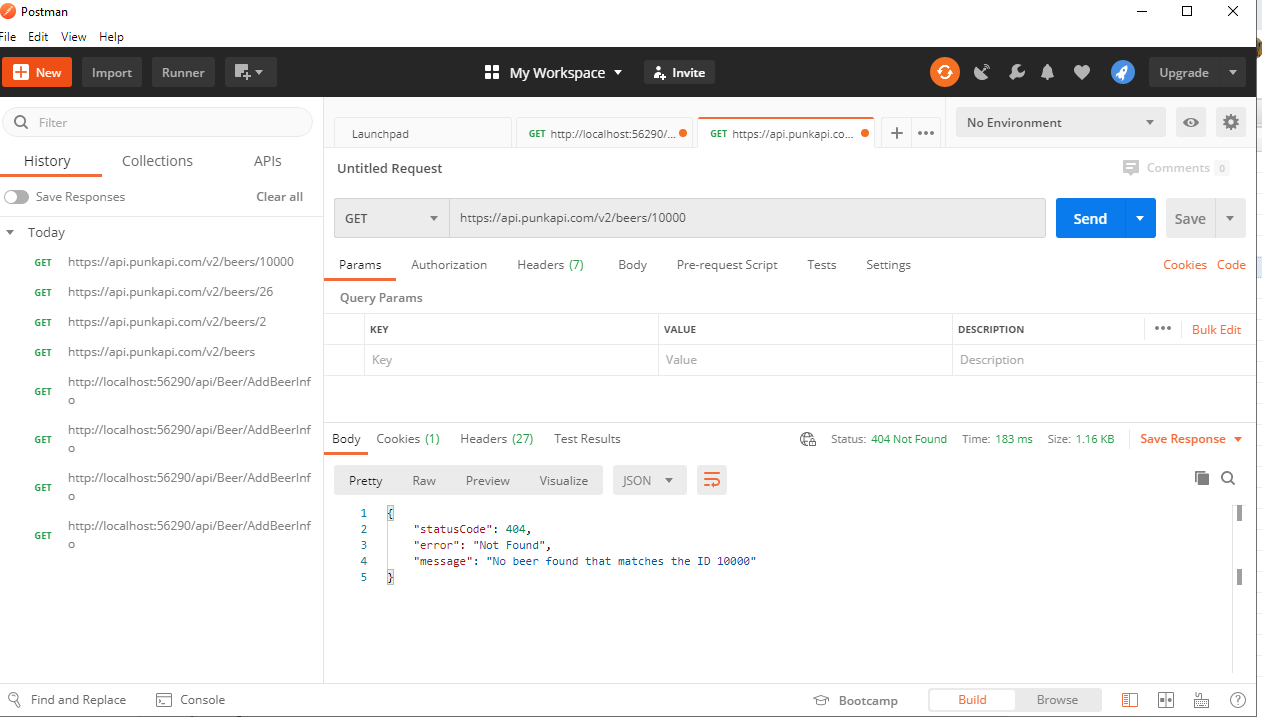
Output Image below.



User rating now inserted in DataBase.Json.



Validation With 10000(Record Not Found in punk api) -please see below picture

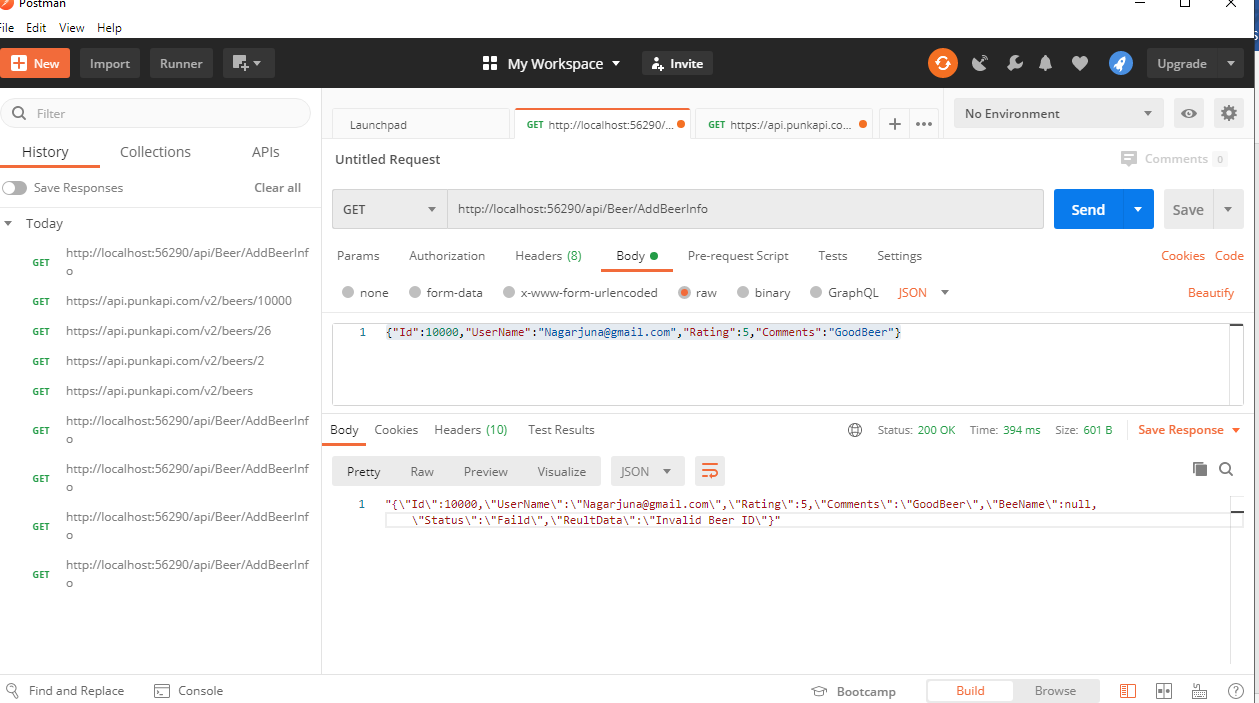


Add validation to ensure the id parameter is a valid beer id by querying the Punk API.

Validation for 10000 No data found

Input: {"Id":10000,"UserName":"Nagarjuna@gmail.com","Rating":5,"Comments":"GoodBeer"}

OutPut: "{\"Id\":10000,\"UserName\":\"Nagarjuna@gmail.com\",\"Rating\":5,\"Comments\":\"GoodBeer\",\"BeeName\":null,\"Status\":\"Faild\",\"ReultData\":\"Invalid Beer ID\"}"

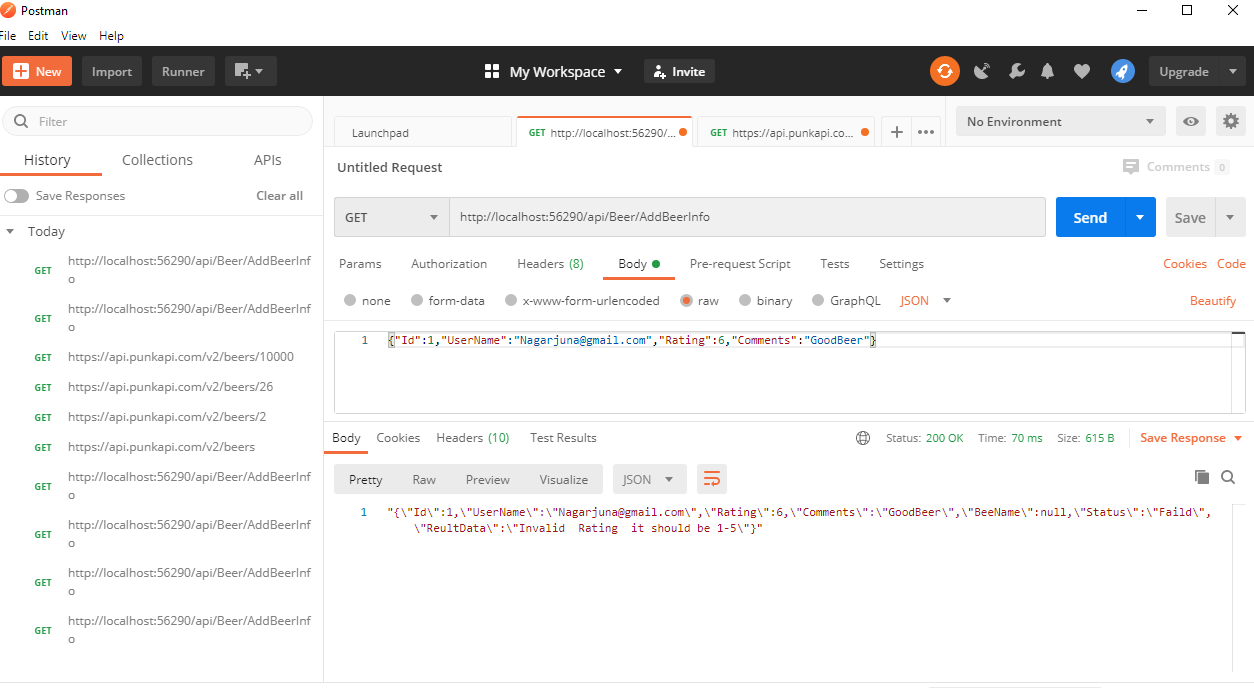


Add validation to ensure that the rating is a valid value in the range of 1 to 5.

To check this validation, I gave rating:6. Results were Failed. See below in picture

Input: {"Id":1,"UserName":"Nagarjuna@gmail.com","Rating":6,"Comments":"GoodBeer"}

OutPut: "{\"Id\":1,\"UserName\":\"Nagarjuna@gmail.com\",\"Rating\":6,\"Comments\":\"GoodBeer\",\"BeeName\":null,\"Status\":\"Faild\",\"ReultData\":\"Invalid  Rating  it should be 1-5\"}"

  
  
Task 2: Add a REST API endpoint to retrieve a list of beers.

1. **I am passing Input as BeerName**

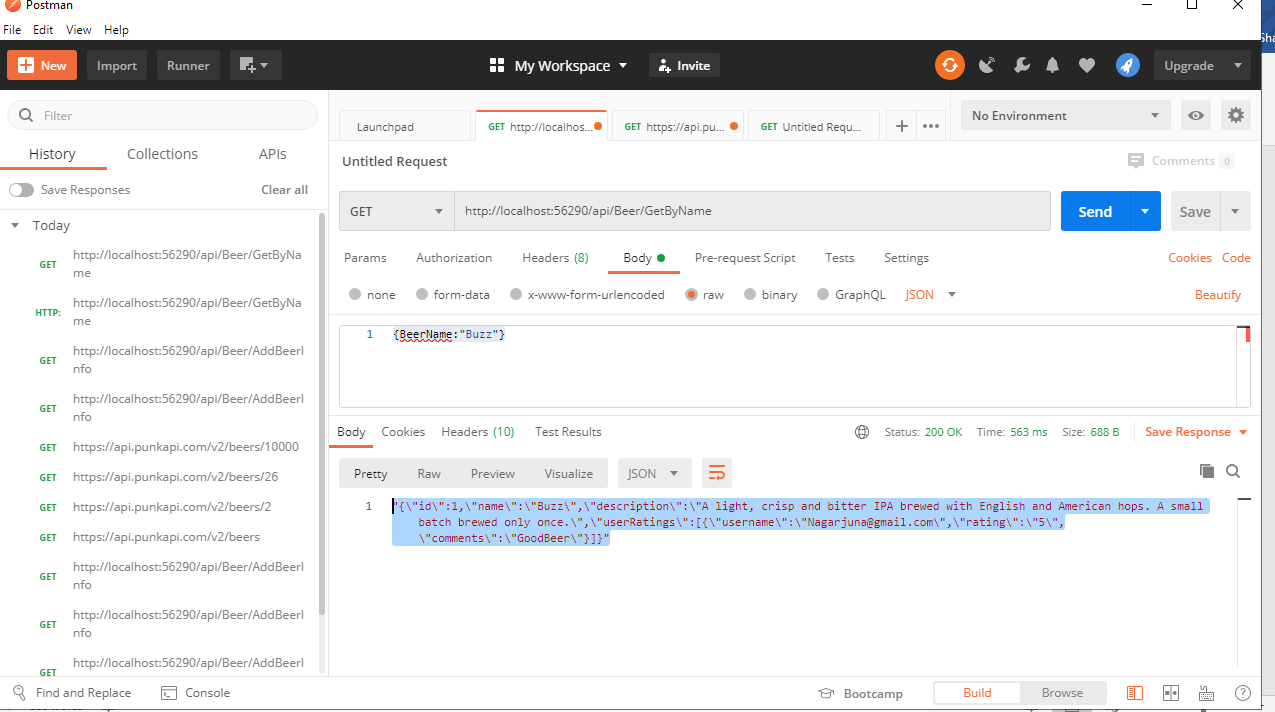
URL: http://localhost:56290/api/Beer/GetByName

Type: GET

I have Choose BeerName as Buzz. In the Output below you can see full details of Buzz beer, and rating for Buzz beer.

InputData: {BeerName:"Buzz"}

OutPut: "{\"id\":1,\"name\":\"Buzz\",\"description\":\"A light, crisp and bitter IPA brewed with English and American hops. A small batch brewed only once.\",\"userRatings\":[{\"username\":\"Nagarjuna@gmail.com\",\"rating\":\"5\",\"comments\":\"GoodBeer\"}]}"

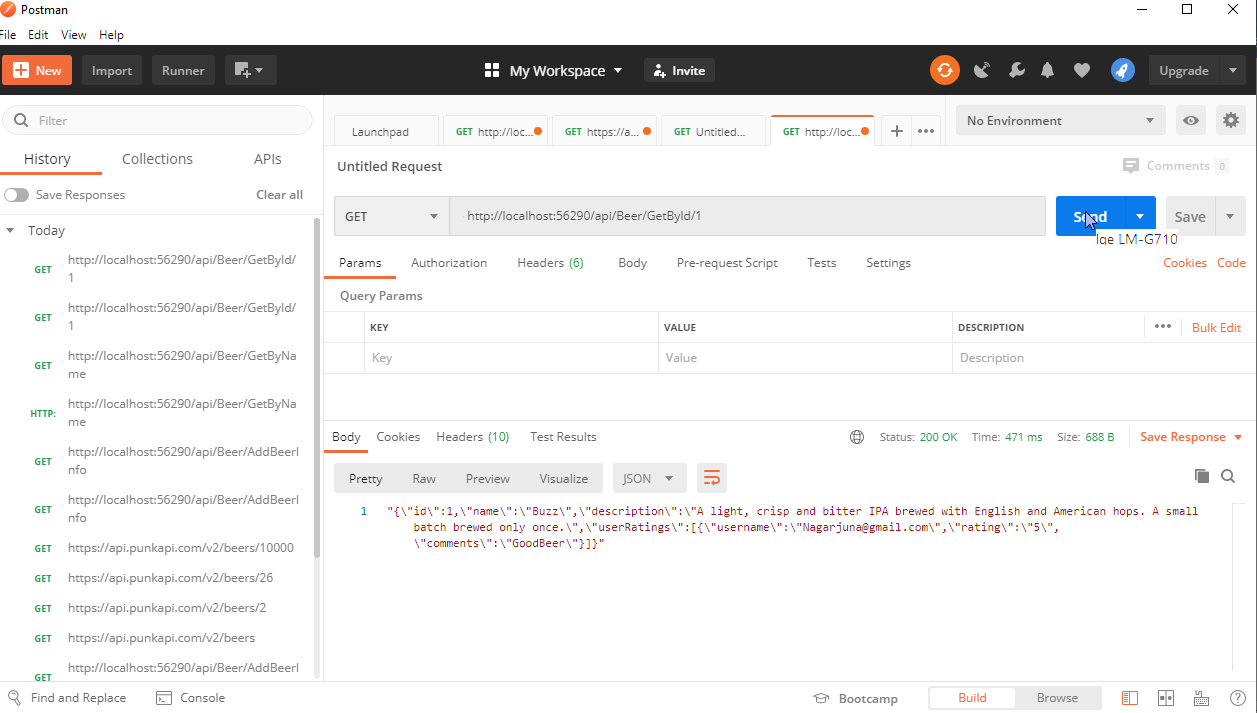


1. **I am passing Input as Beer ID**

URL: <http://localhost:56290/api/Beer/GetById/1>

Here I am calling by Id:1 directly.

Type: GET

Output: "{\"id\":1,\"name\":\"Buzz\",\"description\":\"A light, crisp and bitter IPA brewed with English and American hops. A small batch brewed only once.\",\"userRatings\":[{\"username\":\"Nagarjuna@gmail.com\",\"rating\":\"5\",\"comments\":\"GoodBeer\"}]}"  
  


Task 3: Add a custom Web API Action Filter Attribute

To Check UserName without regular expression, I gave UserName: Nagarjuna only with out domain name. In the Below Output and Image you can see status: "Invalid User Name. User Name  Must be  Eamil id",

Input: {"Id":1,"UserName":"Nagarjuna","Rating":5,"Comments":"GoodBeer"}

OutPut: {

    "Id": 1,

    "UserName": "Nagarjuna",

    "Rating": 5,

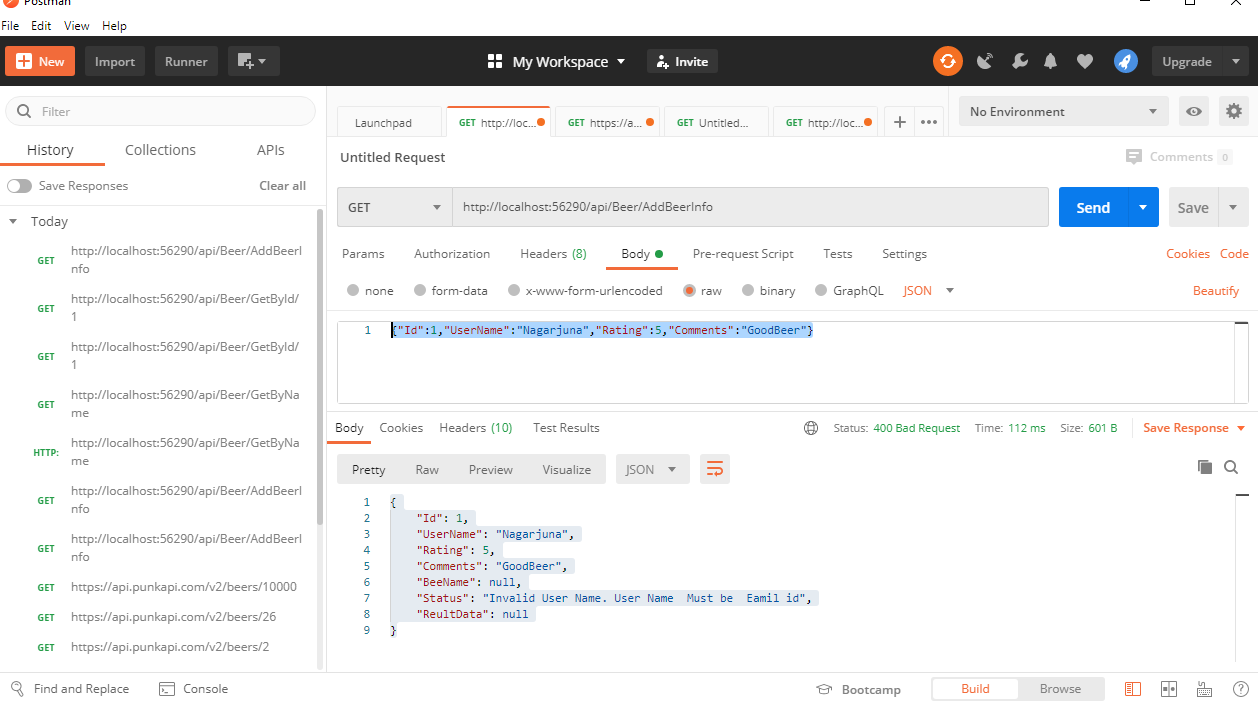
    "Comments": "GoodBeer",

    "BeeName": null,

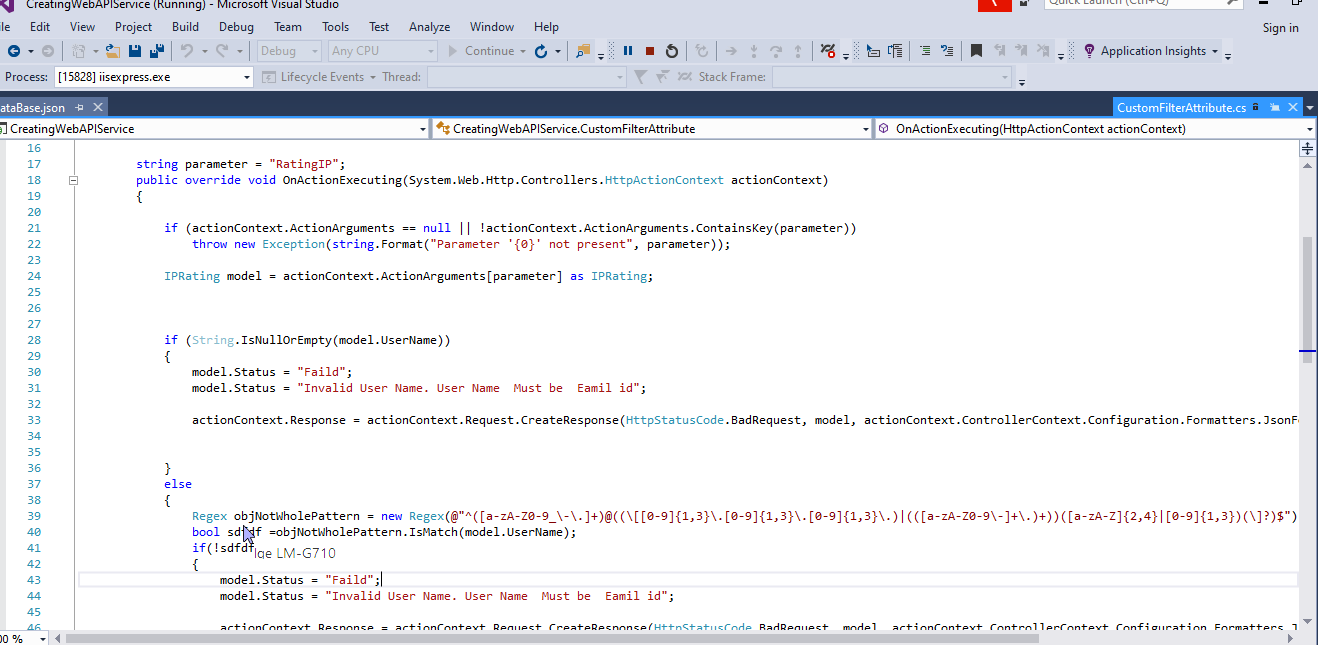
    "Status": "Invalid User Name. User Name  Must be  Eamil id",

    "ReultData": null

}

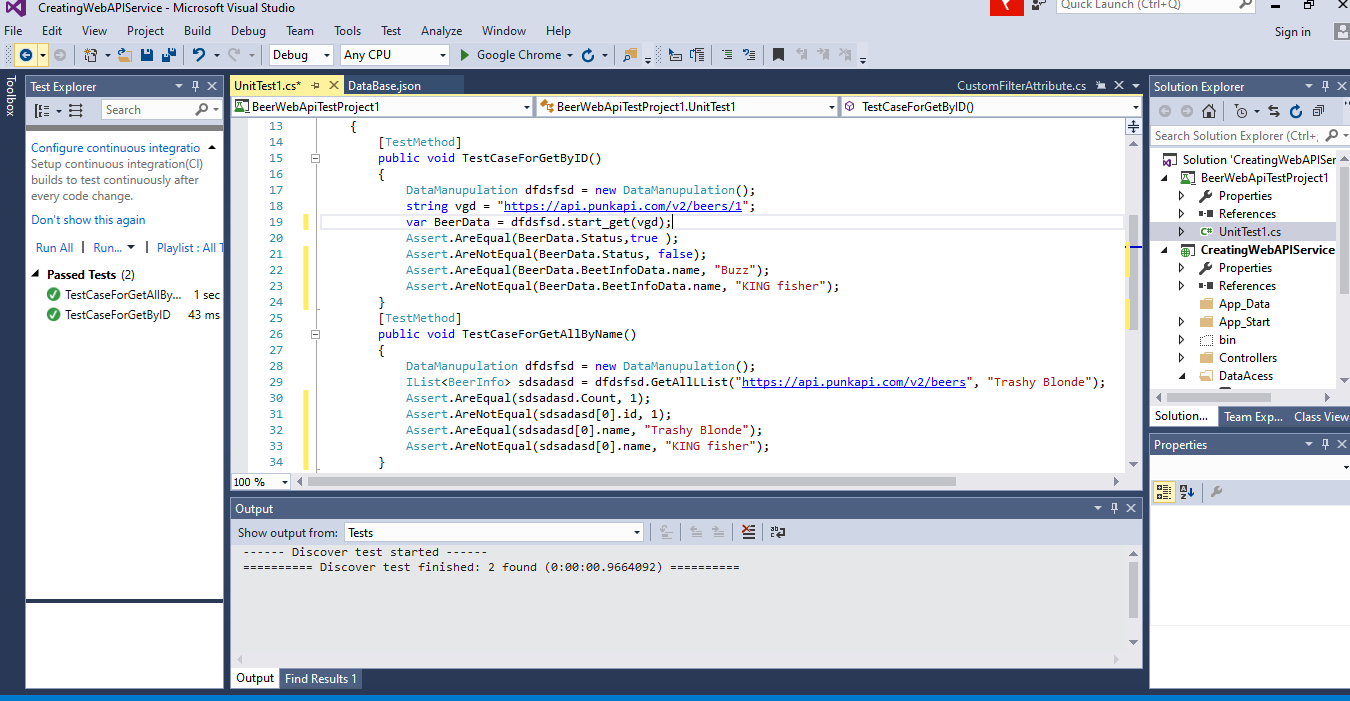


Custom Action Filter added to validated the UserName. Regular Expression applied for the username



Task 4: Add unit tests  
I wrote 2 test cases. You can see the results below in pictures.

Task 4 No.1



I Enjoyed doing this project. Please let me know if you have any questions. I will be very happy to assist you.

Thanks,

Nagarjuna