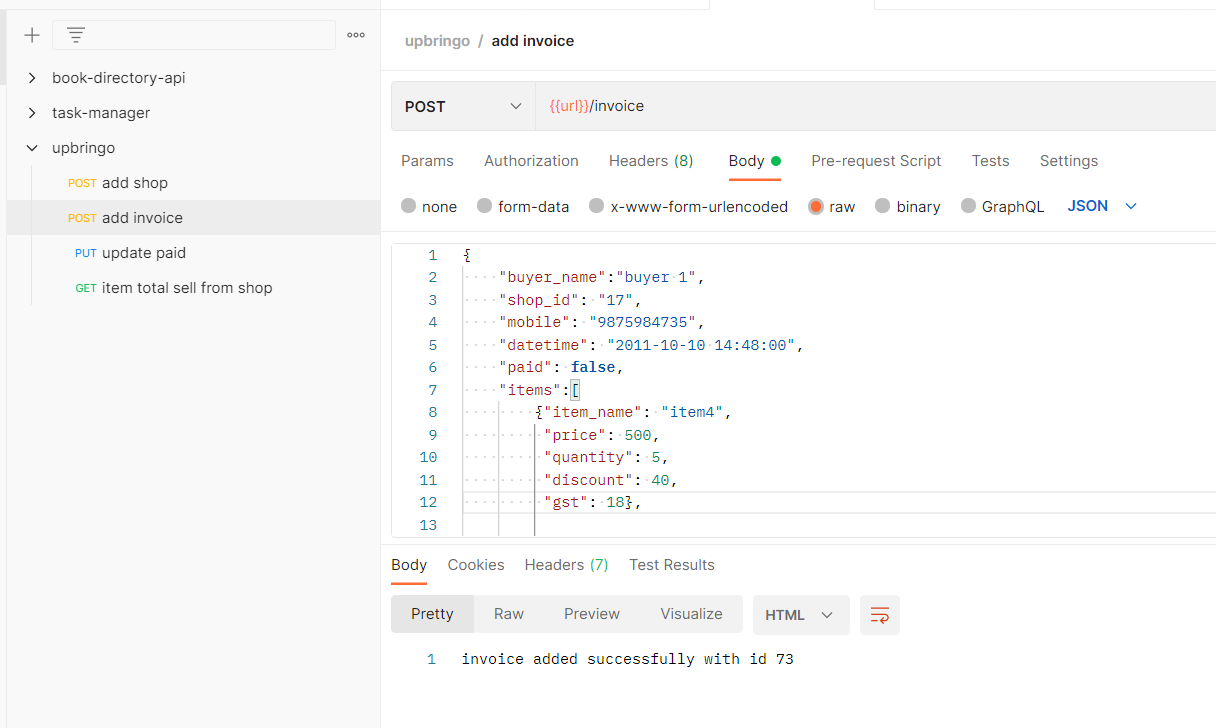
in the video invoice id is not returned, which is updated. Now it returns the invoice id of which shop keeper should take a note. Automated tests to be done using JEST library



README

**# API for storing invoices**

`Node.js` `mysql`

This is an API which allows fascility to store INVOICES of a SHOP.

**## Features**

- Add shop details ( returned shop id should be noted )

- Add Invoice details ( returned invoice id should be noted )

- Update payment status of an Invoice ( using id of the Invoice )

- Get total sale of a particular item for a particular shop ( item name and shop id is required )

**## Install**

npm install

**## Run the app**

npm start

**# API references**

The API to this app is described below.

All the attributes of req body should be given and with same name (case sensitive), along with the format specified below.

**## Add shop details**

**### Request**

`POST /shop`

http://localhost:3000/shop

request body should contain following json object

{

"shop\_name": "laptop repair",

"address": "bhubaneswar",

"mobile": "9876452676",

"email": "sdfld@gmail.com"

}

**### Response**

last added shop\_details id 17

**## Add an Invoice**

Item name is unique. Each item is unique based on its price,gst,discount . If an Item name is not stored already, it will be created and stored.

if an item\_name of any of the item in items array is already existing but the properties (gst,discount,price)

are different. Error occurs. Item name should be changed. response is something like this...

<br/>

{<br/>

"error": "item name already exists , try a difference item\_name specific to its properties",<br/>

"item": {<br/>

"item\_name": "item4",<br/>

"price": 500,<br/>

"quantity": 5,<br/>

"discount": 0,<br/>

"gst": 18<br/>

}<br/>

}

**### Request**

`POST /invoice`

http://localhost:3000/invoice

request body should contain following json object

{

"buyer\_name":"buyer 1",

"shop\_id": "17",

"mobile": "9875984735",

"datetime": "2011-10-10 14:48:00",

"paid": false,

"items":[

{"item\_name": "item4",

"price": 500,

"quantity": 5,

"discount": 40,

"gst": 18},

{"item\_name": "item2",

"price": 350,

"quantity": 6,

"discount": 20,

"gst": 18}

]

}

**### Response**

invoice added successfully with id 73

**## Update an Invoice**

**### Request**

`PUT /invoice/:id/:paid`

http://localhost:3000/invoice/73/true

**### Response**

updated successfully

**## For a shop get total sale of an item**

if shop\_id or item\_name doesn't exist total is 0

**### Request**

`POST /total/:shop\_id/:item\_name`

http://localhost:3000/total/17/item2

**### Response**

{

"item": "item2",

"total": 4008

}