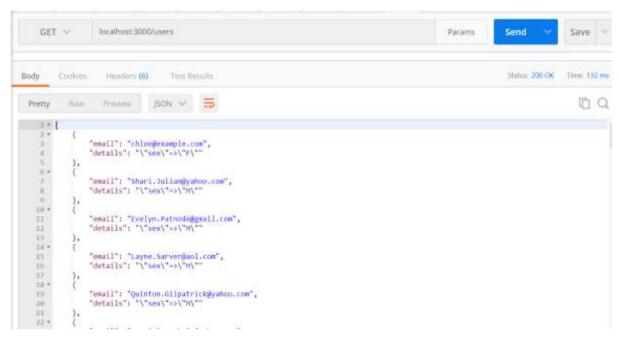
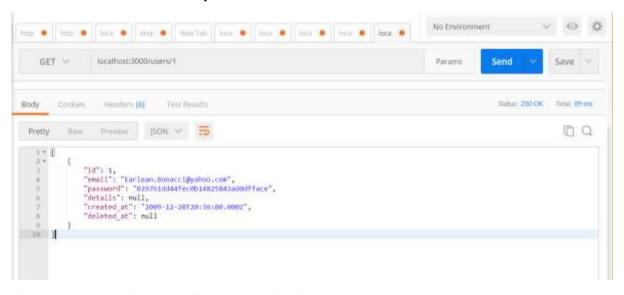
Using Node, Express and Massive create the following HTTP API endpoints serving the following resources as JSON documents

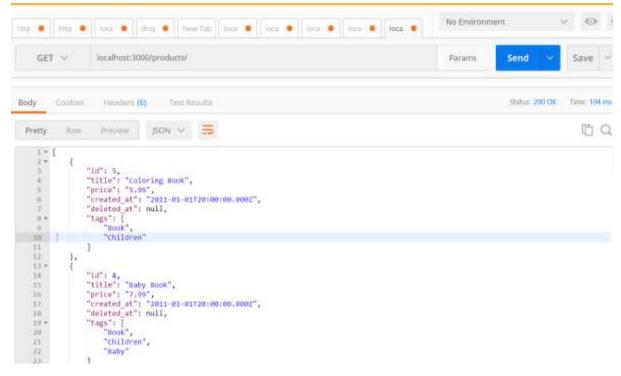
# List all users email and sex in order of most recently created. Do not include password hash in your output



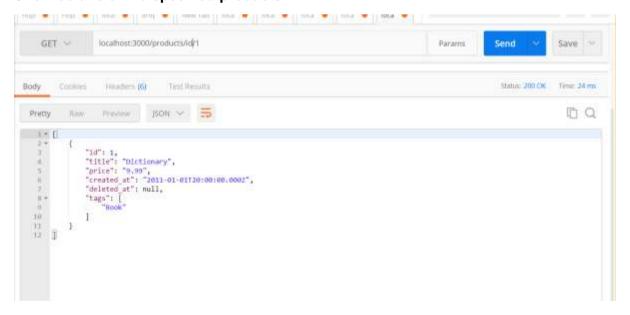
# Show above details of the specified user



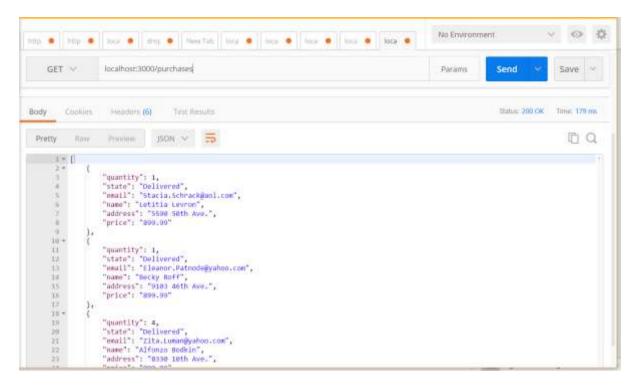
List all products in ascending order of price



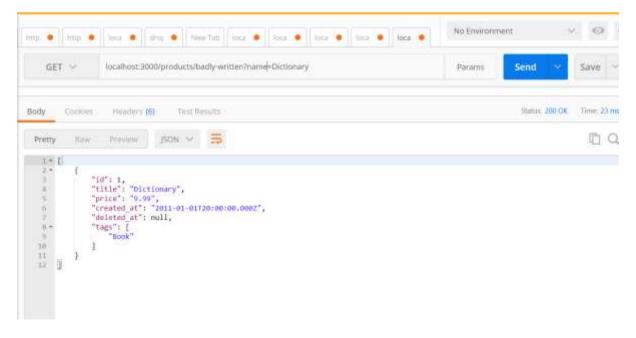
#### Show details of the specified products

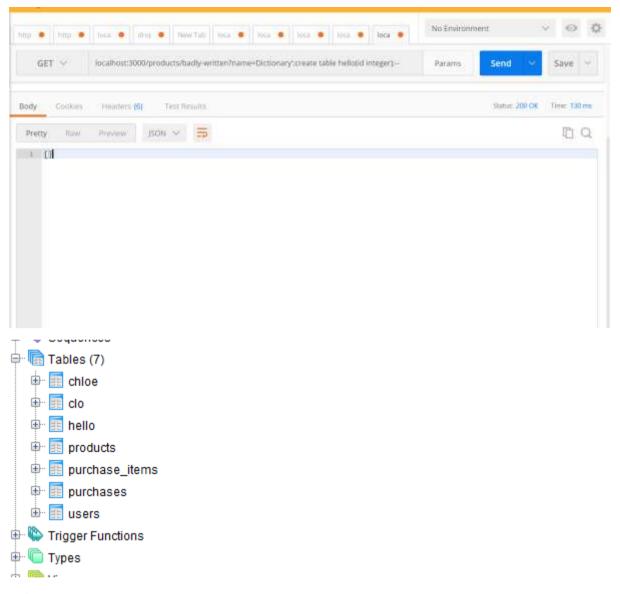


List purchase items to include the receiver's name and, address, the purchaser's email address and the price, quantity and delivery status of the purchased item. Order by price in descending order

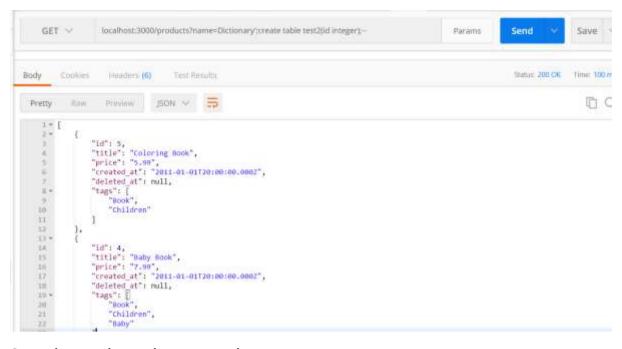


#### Badly implemented get products by name





Safe parametrised query- does not create a table

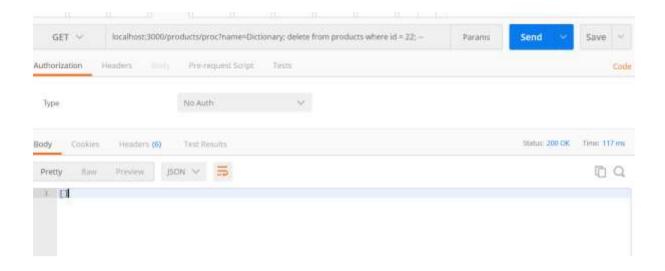


## Created a stored procedure - proc.sql

```
create or replace function get_prods(x character varying)
returns setof products
AS
$$
$$
select * from products where title = x;
$$
language sql
```

```
postgres=#
postgres=# proc.sql
postgres-# \i c:proc.sql
CREATE FUNCTION
postgres-#
```

Run the function and does not allow for sql injection



# **SEQUILIZE PART – CONNECTION SUCCESS**

```
C:\Users\Chocolie\Documents\Final year - semester 2\EAD\labl>node sequalise.js

sequelize deprecated String based operators are now deprecated. Please use Symbol based operators for better security, read more at itp://docs.sequelizejs.com/manual/tutorial/querying.html#operators node_modules\sequelize\lib\sequelize.js:242:13

Executing (default): SELECT 1+1 AS result

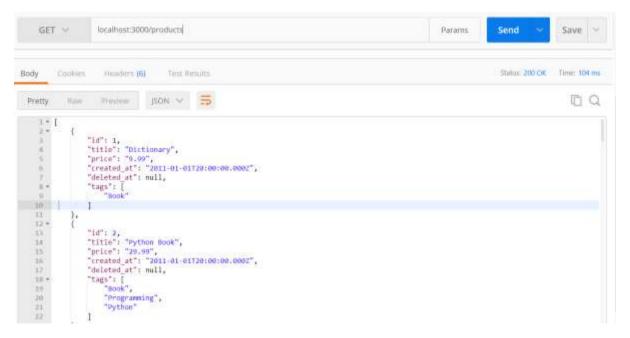
SUCCESS

Example app listening on port 3000)
```

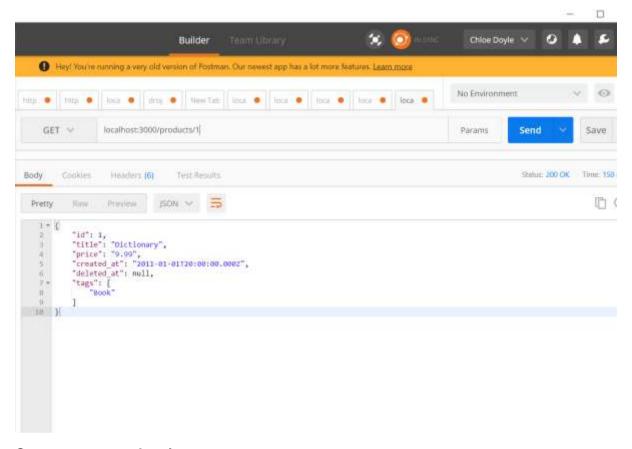
Did this for all tables – purchases, products, users, purchase\_items. And then populated them

Doing the next art – get-put-delete, post requests

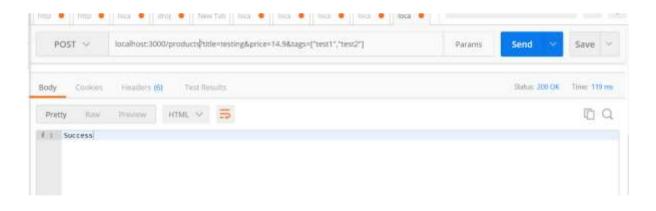
List all products



List all details of specific id

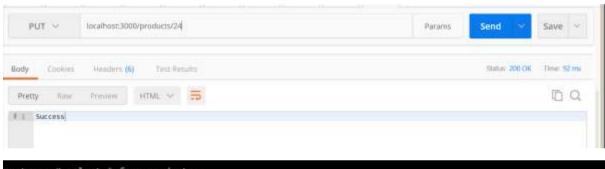


#### Create a new product instance



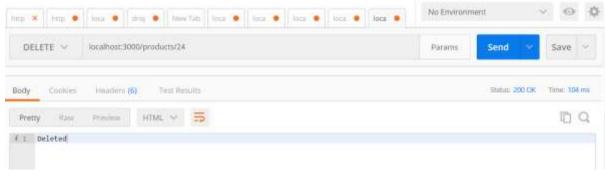
title	price	created_at	deleted_at	tags
Dictionary	9.99	2011-01-01 20:00:00+00		{Book}
Python Book	29.99	2011-01-01 20:00:00+00		{Book, Programming, Python}
Ruby Book	27.99	2011-01-01 20:00:00+00	[	{Book, Programming, Ruby}
Baby Book	7.99	2011-01-01 20:00:00+00	1	{Book,Children,Baby}
Coloring Book	5.99	2011-01-01 20:00:00+00	[	{Book,Children}
Desktop Computer	499.99	2011-01-01 20:00:00+00		{Technology}
Laptop Computer	899.99	2011-01-01 20:00:00+00	[	(Technology)
MP3 Player	108.00	2011-01-01 20:00:00+00	1	{Technology, Music}
42" LCD TV	499.00	2011-01-01 20:00:00+00	1	{Technology,TV}
42" Plasma TV	529.80	2011-01-01 20:00:00+00	1	{Technology,TV}
Classical CD	9.99	2011-01-01 20:00:00+00	1	(Music)
Holiday CD	9.99	2011-01-01 20:00:00+00	1	{Music}
Country CD	9.99	2011-01-01 20:00:00+00	1	{Music}
Pop CD	9.99	2011-01-01 20:00:00+00	1	{Music}
Electronic CD	9.99	2011-01-01 20:00:00+00	1	{Music}
Comedy Movie	14.99	2011-01-01 20:00:00+00	)	{Movie,Comedy}
Documentary	14.99	2011-01-01 20:00:00+00		{Movie}
Romantic	14.99	2011-01-01 20:00:00+00	[	(Movie)
Drama	14.99	2011-01-01 20:00:00+00	1	{Movie}
Action	14.99	2011-01-01 20:00:00+00		{Movie}
Romantic Comedy	15.99	2019-02-16 14:41:38.239936+00		{Movie}
Thriller	9.9	2019-02-17 12:32:17.157213+00		{thriller,scary}
testing	14.9	2019-02-17 14:38:43.415518+00		{test1,test2}

# Updating the price of item 24



id	title	price	created_at	deleted_at	tags
1	Dictionary	9.99	2011-01-01 20:00:00+00		{Book}
2	Python Book	29.99	2011-01-01 20:00:00+00	I. J	{Book, Programming, Python
3	Ruby Book	27.99	2011-01-01 20:00:00+00	1	{Book, Programming, Ruby}
4	Baby Book	7.99	2011-01-01 20:00:00+00	1	{Book,Children,Baby}
5	Coloring Book	5.99	2011-01-01 20:00:00+00	1	{Book,Children}
6	Desktop Computer	499.99	2011-01-01 20:00:00+00		{Technology}
7	Laptop Computer	899.99	2011-01-01 20:00:00+00	i i	{Technology}
8	MP3 Player	108.00	2011-01-01 20:00:00+00	1	{Technology, Music}
9	42" LCD TV	499.00	2011-01-01 20:00:00+00	i i	{Technology,TV}
a	42" Plasma TV	529.00	2011-01-01 20:00:00+00		{Technology, TV}
1	Classical CD	9.99	2011-01-01 20:00:00+00	į i	{Music}
2	Holiday CD	9.99	2011-01-01 20:00:00+00	1	{Music}
3	Country CD	9.99	2011-01-01 20:00:00+00		{Music}
4	Pop CD	9.99	2011-01-01 20:00:00+00	i i	{Music}
5	Electronic CD	9.99	2011-01-01 20:00:00+00	1	{Music}
6	Comedy Movie	14.99	2011-01-01 20:00:00+00		{Movie,Comedy}
7	Documentary	14.99	2011-01-01 20:00:00+00	1	{Movie}
8	Romantic	14.99	2011-01-01 20:00:00+00	1	{Movie}
9	Drama	14.99	2011-01-01 20:00:00+00		{Movie}
0	Action	14.99	2011-01-01 20:00:00+00		{Movie}
1	Romantic Comedy	15.99	2019-02-16 14:41:38.239936+00	i i	{Movie}
2	Thriller	9.9	2019-02-17 12:32:17.157213+00		{thriller,scary}
4	testing	15.99	2019-02-17 14:38:43.415518+00	i i	{test1,test2}

Remove an existing product – deleting item 24



d	title	price	created_at	deleted_at	tags
1	Dictionary	9.99	2011-01-01 20:00:00+00		(Book)
2	Python Book	29.99	2011-01-01 20:00:00+00		(Book, Programming, Python)
з [	Ruby Book	27.99	2011-01-01 20:00:00+00	ĵi i	{Book, Programming, Ruby}
ı İ	Baby Book	7.99	2011-01-01 20:00:00+00		{Book,Children,Baby}
5	Coloring Book	5.99	2011-01-01 20:00:00+00	į i	(Book,Children)
5 j	Desktop Computer	499.99	2011-01-01 20:00:00+00		{Technology}
7	Laptop Computer	899.99	2011-01-01 20:00:00+00	į i	{Technology}
3	MP3 Player	108.00	2011-01-01 20:00:00+00		{Technology, Music}
ρį	42" LCD TV	499.00	2011-01-01 20:00:00+00	į į	{Technology,TV}
9	42" Plasma TV	529.00	2011-01-01 20:00:00+00		{Technology,TV}
ιĺ	Classical CD	9.99	2011-01-01 20:00:00+00	į i	(Music)
	Holiday CD	9.99	2011-01-01 20:00:00+00		{Music}
1	Country CD	9.99	2011-01-01 20:00:00+00		{Music}
H	Pop CD	9,99	2011-01-01 20:00:00+00	į i	{Music}
5	Electronic CD	9.99	2011-01-01 20:00:00+00	10	(Music)
5 İ	Comedy Movie	14.99	2011-01-01 20:00:00+00	i i	{Movie,Comedy}
7	Documentary	14.99	2011-01-01 20:00:00+00		{Movie}
3	Romantic	14.99	2011-01-01 20:00:00+00	ii i	{Movie}
9	Drama	14.99	2011-01-01 20:00:00+00		{Movie}
i	Action	14.99	2011-01-01 20:00:00+00	i i	{Movie}
	Romantic Comedy	15.99	2019-02-16 14:41:38.239936+00		{Movie}
ı İ	Thriller	9.9	2019-02-17 12:32:17.157213+00	i i	{thriller,scary}

```
var purchases = sequelize.define('purchases',
∃ {
     id:
         type: Sequelize.INTEGER,
         primaryKey: true,
         autoIncrement: true
     created at:
      type: Sequelize.DATE,
       defaultValue: sequelize.literal('NOW()')
     },
     name:
         type: Sequelize.STRING
     address:
         type: Sequelize.STRING
     },
     state:
3
         type: Sequelize.STRING
     zipcode:
         type: Sequelize.INTEGER
     },
     user id:
```

#### **Inserting into products**

```
C:\Users\Chooolie\Documents\Final year - semester 2\EAD\lab1>
C:\Users\Chooolie\Documents\Final year - semester 2\EAD\lab1> node sequalise.js
sequalize deprecated String based operators are now deprecated. Please use Symbol based operators for better security, read more at ht
tp://docs.sequalizejs.com/nanual/tutorial/tutorial/querying.html#operators-node_modular\sequalize\lib\sequalize_js:242:13
Executing (default): SELECT 1+1 AS result
SUCCESS
Example app listening on port 3000!
Executing (default): INSERT INTO "products" ("id", "title", "price", "created_at", "deleted_at", "tags") VALUES (DEFAULT, 'chloe', '14.9', NOW
(), NULL, ARRAY['test1', 'test2']::TEXT[]) RETURNINS *;
```

New product create = romantic comedy

```
ostgres=# select * from products;
          title
                        price
id |
                                              created at
                                                                     | deleted_at |
                                                                                                tags
     Dictionary
                            9.99
                                    2011-01-01 20:00:00+00
                                                                                     {Book, Programming, Python}
{Book, Programming, Ruby}
{Book, Children, Baby}
{Book, Children}
     Python Book
Ruby Book
                           29.99
                                    2011-01-01 20:00:00+00
                           27.99
                                    2011-01-01 20:00:00+00
     Baby Book
                            7.99
                                    2011-01-01 20:00:00+00
     Coloring Book
                            5.99
                                    2011-01-01 20:00:00+00
                                                                                      (Technology)
     Desktop Computer
                          499.99
                                    2011-01-01 20:00:00+00
                                                                                      (Technology)
                                    2011-01-01 20:00:00+00
     Laptop Computer
                          899.99
     MP3 Player
                                                                                      {Technology, Music}
                          108.00
                                    2011-01-01 20:00:00+00
     42" LCD TV
                                                                                      (Technology, TV)
                          499.00
                                    2011-01-01 20:00:00+00
     42" Plasma TV
                          529.00
                                    2011-01-01 20:00:00+00
                                                                                      (Technology, TV)
11
     Classical CD
                            9.99
                                    2011-01-01 20:00:00+00
                                                                                     {Music}
12
13
                            9.99
                                    2011-01-01 20:00:00+00
     Holiday CD
                                                                                     {Music}
                                    2011-01-01 20:00:00+00
     Country CD
                            9.99
                                                                                     {Music}
     Pop CD
                            9.99
                                    2011-01-01 20:00:00+00
                                                                                     {Music}
                                                                                     {Music}
{Movie,Comedy}
15
     Electronic CD
                            9.99
                                    2011-01-01 20:00:00+00
     Comedy Movie
                                    2011-01-01 20:00:00+00
16
                           14.99
     Documentary
                           14.99
                                    2011-01-01 20:00:00+00
                                                                                      (Movie)
18
     Romantic
                           14.99
                                    2011-01-01 20:00:00+00
                                                                                     (Movie)
19
     Drama
                           14.99
                                    2011-01-01 20:00:00+00
                                                                                     (Movie)
     Action
                                    2011-01-01 20:00:00+00
20
                           14.99
                                                                                      (Movie)
                           15.99
                                    2019-02-16 14:41:38.239936+00
    Romantic Comedy
                                                                                     (Movie)
```

#### Did and tested for all others too - like this

```
users.create(
{
  email: 'chloe@example.com',
  password: '1234',
  details: 'sex=>F'
}).then( users => {
  users.save()
}).catch(err => {
  console.log(err)
})
```