

## PART 1

## GET/users

```
app.get("/users", (req, res) => {
  massiveInstance.query("select email, details from users order by created_at")
    .then((result) => {
      res.send(result);
    })
});
```

```
C:\Users\Ira\Desktop\Lab1>node index.js
```

localhost:3000/users

```
{
  "email": "Romaine.Birdsell@aol.com",
  "details": "\sex\=>\F\\"",
  "email": "Ivana.Sosnowski@aol.com",
  "details": "\sex\=>\M\\"",
  "email": "Cherryl.Tarnowski@gmail.com",
  "details": null,
  "email": "Zita.Luman@yahoo.com",
  "details": null,
  "email": "Takako.Gilpatrick@aol.com",
  "details": "\sex\=>\M\\"",
  "email": "Derek.Crenshaw@gmail.com",
  "details": "\sex\=>\F\\"",
  "email": "Jeremiah.Buonocore@yahoo.com",
  "details": null,
  "email": "Samatha.Pellegrin@yahoo.com",
  "details": null,
  "email": "Danny.Crays@gmail.com",
  "details": "\sex\=>\M\\"",
  "email": "Shanell.Lichtenstein@aol.com",
  "details": "\sex\=>\M\\"",
  "email": "Wynona.Greening@aol.com",
  "details": "\sex\=>\M\\"",
  "email": "Carmel.Bulfer@aol.com",
  "details": "\sex\=>\F\\"",
  "state\=>\Florida\\"",
  "email": "Edmund.Roles@yahoo.com",
  "details": "\sex\=>\F\\"",
  "state\=>\New York\\"",
  "email": "Cortney.Strayer@gmail.com",
  "details": "\sex\=>\M\\"",
  "state\=>\Virginia\\"",
  "email": "Harrison.Puett@yahoo.com",
  "details": "\sex\=>\M\\"",
  "email": "Granville.Hedgpeth@gmail.com",
  "details": null,
  "email": "Zita.Breeding@gmail.com",
  "details": null,
  "email": "Claud.Cousineau@gmail.com",
  "details": "\sex\=>\F\\"",
  "email": "Mauro.Pung@yahoo.com",
  "details": "\sex\=>\F\\"",
  "state\=>\Illinois\\"",
  "email": "Eve.Kump@yahoo.com",
  "details": "\sex\=>\M\\"",
  "email": "Angel.Lessley@yahoo.com",
  "details": "\sex\=>\F\\"",
  "email": "Salvatore.Arends@aol.com",
  "details": "\sex\=>\F\\"",
  "state\=>\Virginia\\"",
  "email": "Humberto.Jonson@yahoo.com",
  "details": null,
  "email": "Vivian.Westmoreland@yahoo.com",
  "details": "\sex\=>\F\\"",
  "state\=>\South Carolina\\"",
  "email": "Wan.Dilks@gmail.com",
  "details": "\sex\=>\M\\"",
}
```

## GET/users/:id

```
app.get("/users/:id", (req, res) => {
  var id = req.params.id
  massiveInstance.query("select details from users where id = $1", [id])
    .then((result) => {
      res.send(result);
    })
});
```

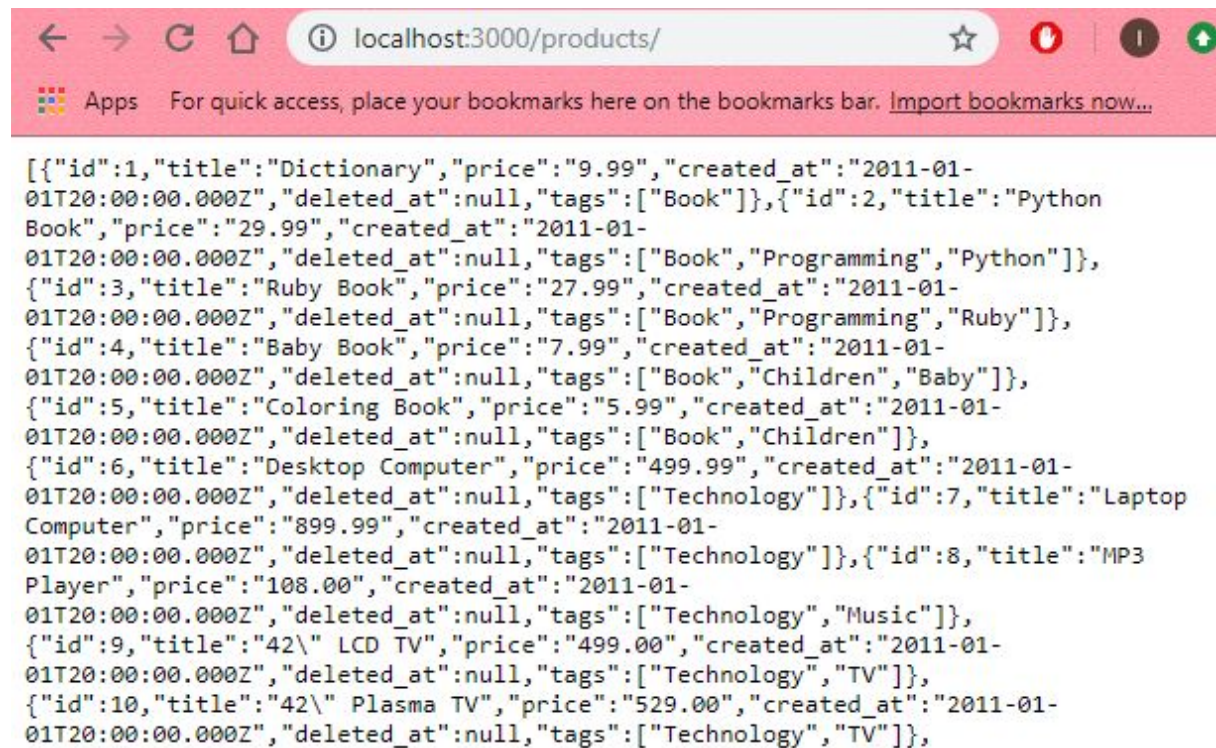
localhost:3000/users/7

Apps For quick access, place your bookmarks here on the

```
[{"details": "\"sex\"=>\"M\""}]
```

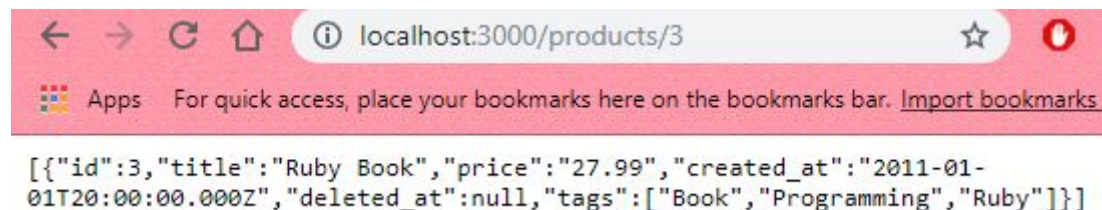
## GET/products

```
app.get("/products", (req, res) => {
  if (req.query.name == undefined) {
    massiveInstance.query("select * from products")
      .then((result) => {
        res.send(result);
      })
  }
});
```



## GET/products/:id

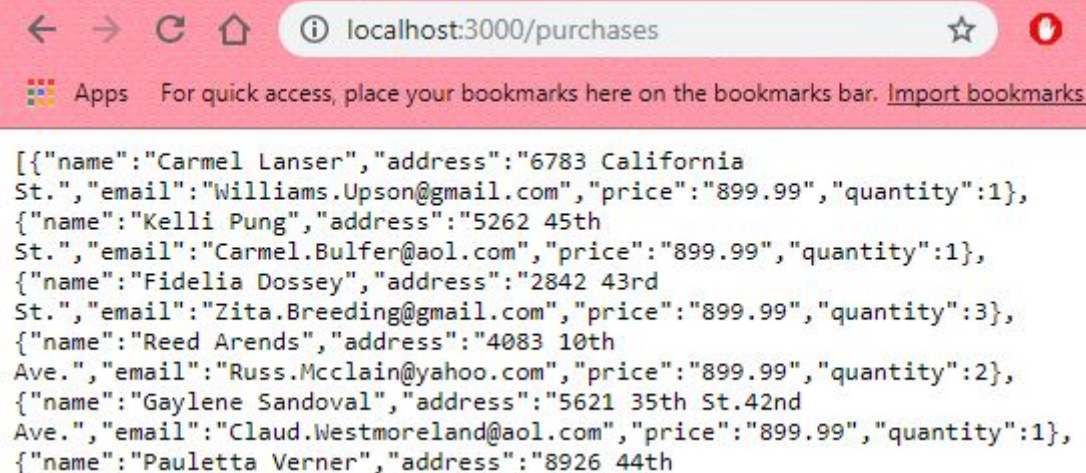
```
app.get("/products/:id", (req, res) => {
  var id = req.params.id
  massiveInstance.query("select * from products where id = $1", [id])
    .then((result) => {
      res.send(result);
    })
});
```





## GET/purchases

```
app.get("/purchases", (req, res) => {
  massiveInstance.query("select name, address, email, price,
    quantity FROM purchases A JOIN users B on (A.user_id = B.id)
    join purchase_items on (purchase_id = A.id) order by price desc;
  ").then((result) => {
    res.send(result);
  });
});
```



localhost:3000/purchases

Apps For quick access, place your bookmarks here on the bookmarks bar. [Import bookmarks](#)

```
[{"name": "Carmel Lanser", "address": "6783 California St.", "email": "Williams.Upson@gmail.com", "price": "899.99", "quantity": 1}, {"name": "Kelli Pung", "address": "5262 45th St.", "email": "Carmel.Bulfer@aol.com", "price": "899.99", "quantity": 1}, {"name": "Fidelia Dossey", "address": "2842 43rd St.", "email": "Zita.Breeding@gmail.com", "price": "899.99", "quantity": 3}, {"name": "Reed Arends", "address": "4083 10th Ave.", "email": "Russ.Mcclain@yahoo.com", "price": "899.99", "quantity": 2}, {"name": "Gaylene Sandoval", "address": "5621 35th St. 42nd Ave.", "email": "Claud.Westmoreland@aol.com", "price": "899.99", "quantity": 1}, {"name": "Pauletta Verner", "address": "8926 44th
```

## PART 2

```
// PART 2 EXTENDED
// else
// {
//   massiveInstance.query("//http://localhost:3000/
products?name=Dictionary"; Delete from products where title = Dictionary;--
//   "select * from products where title = '" + req.query.name + "';")
//   .then(items => {
//     res.json(items);
//   });
// }
// }
```






## PART 3

```
// PART 3 - PARAMETERISED
// massiveInstance.query("select * from products where title = $1", [title])





// PART 3 - STORED PROCEDURE
// massiveInstance.getProductbytitle(req.query.name)
// .then(items => {
//   res.json(items);
// });
```

## PART 4

» Desktop » Lab1 » node\_modules » .bin » models

Name	Date modified	Type
 index	17/02/2019 12:19	JS File
 products	17/02/2019 16:42	JS File
 purchases	17/02/2019 16:43	JS File
 purchases_items	17/02/2019 16:43	JS File
 users	17/02/2019 16:40	JS File

» Desktop » Lab1 » node\_modules » .bin » migrations

Name	Date modified	Type
 20190217164020-create-users	17/02/2019 16:40	JS File
 20190217164253-create-products	17/02/2019 16:42	JS File
 20190217164300-create-purchases	17/02/2019 16:43	JS File
 20190217164306-create-purchases-items	17/02/2019 16:43	JS File

## PART 5

```
C:\Users\Ira\Desktop\Lab1\node_modules\.bin>sequelize db:seed:all

Sequelize CLI [Node: 10.15.1, CLI: 5.4.0, ORM: 4.42.0]





Loaded configuration file "config\config.json".
Using environment "development".
sequelize deprecated String based operators are now deprecated. Please use
read more at http://docs.sequelizejs.com/manual/tutorial/querying.html#opera
== 20190217211620-users: migrating =====
== 20190217211620-users: migrated (0.020s)

== 20190217211631-products: migrating =====
== 20190217211631-products: migrated (0.012s)

== 20190217211640-purchases: migrating =====
== 20190217211640-purchases: migrated (0.030s)

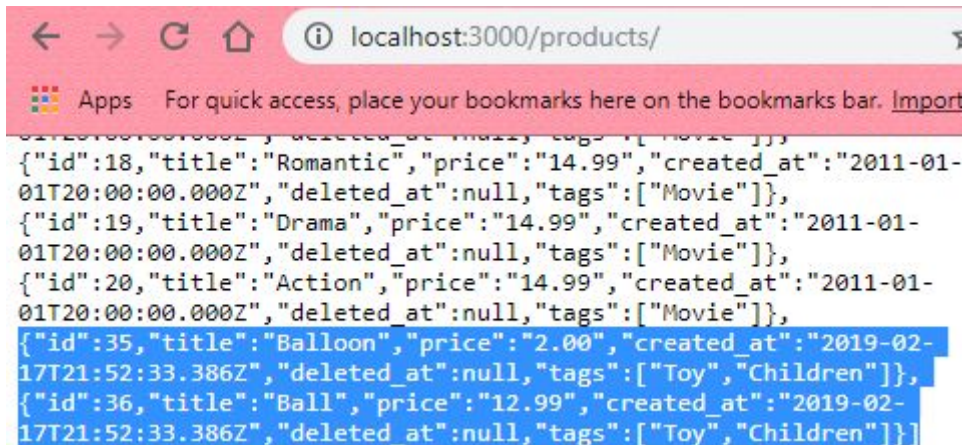
== 20190217211648-purchases-items: migrating =====
== 20190217211648-purchases-items: migrated (0.004s)
```

» Desktop » Lab1 » node\_modules » .bin » seeders

Name	Date modified	Type
 20190217211620-users	17/02/2019 21:33	JS File
 20190217211631-products	17/02/2019 21:36	JS File
 20190217211640-purchases	17/02/2019 21:41	JS File
 20190217211648-purchases-items	17/02/2019 21:44	JS File

## PART 6

### GET/products (including new data)



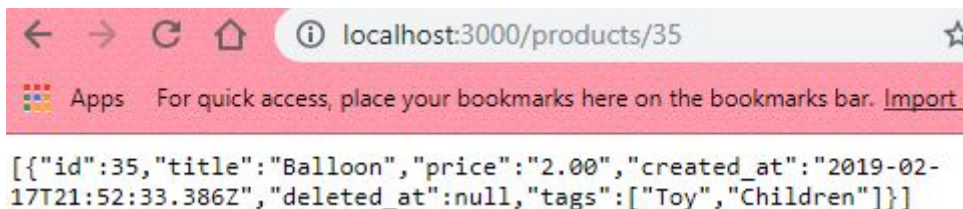
```
{
  "id": 18, "title": "Romantic", "price": "14.99", "created_at": "2011-01-01T20:00:00.000Z", "deleted_at": null, "tags": ["Movie"]},
  {"id": 19, "title": "Drama", "price": "14.99", "created_at": "2011-01-01T20:00:00.000Z", "deleted_at": null, "tags": ["Movie"]},
  {"id": 20, "title": "Action", "price": "14.99", "created_at": "2011-01-01T20:00:00.000Z", "deleted_at": null, "tags": ["Movie"]},
  {"id": 35, "title": "Balloon", "price": "2.00", "created_at": "2019-02-17T21:52:33.386Z", "deleted_at": null, "tags": ["Toy", "Children"]},
  {"id": 36, "title": "Ball", "price": "12.99", "created_at": "2019-02-17T21:52:33.386Z", "deleted_at": null, "tags": ["Toy", "Children"]}
]
```



```
'use strict';

module.exports = {
  up: (queryInterface, Sequelize) => {
    return queryInterface.bulkInsert('products', [
      {
        title: 'Balloon',
        price: '2.00',
        created_at: new Date(),
        tags: ['Toy', 'Children']
      },
      {
        title: 'Ball',
        price: '12.99',
        created_at: new Date(),
        tags: ['Toy', 'Children']
      }
    ], {});
  },
  down: (queryInterface, Sequelize) => {
    return queryInterface.bulkDelete('products', {
      title: [
        'Balloon',
        'Ball'
      ]
    }, {});
  }
};
```

### GET/products/:id



```
[{"id": 35, "title": "Balloon", "price": "2.00", "created_at": "2019-02-17T21:52:33.386Z", "deleted_at": null, "tags": ["Toy", "Children"]}]
```



## POST/products

```
router.post('/products', function(req, res) {
  models.product.create({
    title: req.body.title,
    price: req.body.price,
    created_at: new Date(),
    deleted_at: null,
    tags: req.body.tags
  }).then(function(product) {
    res.json(product);
  });
});
```

## PUT/products/:id

```
router.put('/products/:id', function(req, res) {
  models.product.find({
    where: {
      id: req.params.id
    }
  }).then(function(product) {
    if(product){
      product.updateAttributes({
        title: req.body.title,
        price: req.body.price,
        created_at: req.body.created_at,
        deleted_at: req.body.deleted_at,
        tags: req.body.tags
      }).then(function(product) {
        res.send(product);
      });
    }
  });
});
```

## DELETE/products/:id

```
router.delete('/products/:id', function(req, res) {

  models.product_item.destroy({
    where: {
      id: req.params.id
    }
  }).then(function(product) {
    res.json(product);
  }).catch(function(error) {
    res.json(error['message']);
  });

  models.product.destroy({
    where: {
      id: req.params.id
    }
  }).then(function(product) {
    res.json(product);
  }).catch(function(error) {
    res.json(error['message']);
  });
});
```