

Couchbase NoSQL Developer Workshop

ラボハンドブック

共通（凡例・付録）

凡例 - お読みください!

すべてのラボ モジュールで次の書式指定規則を使用します。

- **黄色** の強調表示された値 (テキストまたはスクリーンショット) は、環境での正しい値に置き換える必要があります(例: **192.168.61.101**)
- **オレンジ色** の強調表示された値は、タスクのコンテキストでの追加の強調に使用されます。
- **courier** のコマンド **bold** の **新しい** フォントは、実行する予定のものです (下記参照)。
 - プレフィックスは、ローカルコンピュータ (MacおよびLinux)のターミナルで実行されるコマンドを示します。
 - ">"接頭辞は、Local コンピュータで実行される Windows 固有のコマンドを示すためだけに使用されます。すべてのラボでは **Windows PowerShell** を "ターミナル" と呼び、すべてのコマンドの汎用例では、Mac や Linuxのプレフィックス、つまり"\$"と仮定し、Windows で実行する前に一部のコマンドを変更する必要があります。
 - プレフィックスのないテキストは、コマンドの出力です。

```
$ docker -v
Docker version 19.03.8, build afacb8b
$ git --version
Git version 2.23.0
$ docker container ls --all
```

CONTAINER ID	IMAGE	COMMAND	CREATED	STATUS	PORTS	NAMES
--------------	-------	---------	---------	--------	-------	-------

- **赤のフォント** は重要な指示を示しています。細心の注意を払ってください。
- **太字**または**斜体**フォントの項目は、注意を払ったりクリックしたりするための名前またはUI要素です。

特に明記されていない限り、すべてのコマンド、パラメータ、設定、ログイン、パスワードなどでは**大文字と小文字が区別されます!**

端末のコマンドの出力は、ラボで示されている出力とは異なる可能性があります。一般的に、出力が類似している限り、これは問題ではないはずです。立ち往生した場合は、助けを求めてください。

注意:

コードやコマンドをコピーして貼り付けると、テキストエディタの自動フォーマットが原因でエラーが発生する可能性があります。自動フォーマットされた文字を使用すると、コードが正しく実行されません。この場合は、コピーして貼り付ける代わりにコマンドを入力してみてください。

node.js API プロジェクト構造

```
|— configuration
|   |— config.js
|   |— config.json
|
|— controllers
|   |— productController.js
|   |— testController.js
|   |— userController.js
|
|— library
|   |— outputMessage.js
|   |— verifyToken.js
|
|— repository
|   |— repository.js
|
|— resources
|   |— labs
|
|— service
|   |— productService.js
|   |— userService.js
|
|— Dockerfile.dev
|— package.json
|— server.js
```

サンプル顧客ドキュメント

```
{
  "doc": {
    "type": "customer",
    "schema": "1.0.0",
    "created": 1559780511352,
    "createdBy": 195,
    "modified": 1587833486406,
    "modifiedBy": 702
  },
  "_id": "customer_100",
  "custId": 100,
  "custName": {
    "firstName": "Hilton",
    "lastName": "Schinner"
  },
  "username": null,
  "email": "Karianne39@gmail.com",
  "createdOn": "2019-06-08",
```

```

"address": {
  "home": {
    "address1": "4810 Hegmann Manors Burgs",
    "city": "North Ivahfort",
    "state": "WY",
    "zipCode": "01961",
    "country": "KG"
  },
  "work": {
    "address1": "198 Molly Mountain Plains",
    "city": "South Clifton",
    "state": "CT",
    "zipCode": "77161",
    "country": "SA"
  }
},
"mainPhone": {
  "phone_number": "682126158733138",
  "extension": null
},
"additionalPhones": {
  "type": "Other",
  "phone_number": "987601914834116",
  "extension": null
}
}

```

サンプル Order ドキュメント

```

{
  "doc": {
    "type": "order",
    "schema": "1.0.0",
    "created": 1584603465196,
    "createdBy": 4075,
    "modified": 1570944661302,
    "modifiedBy": 4991
  },
  "_id": "order_100",
  "orderId": 100,
  "custId": 658,
  "orderDate": 1577676607455,
  "orderStatus": "Pending",
  "billingInfo": {
    "name": "Linwood Emard",
    "phone": "1-229-156-9636 ",
    "email": "Daisy42@hotmail.com",
    "address": {
      "address1": "6989 Frami Way Wall",
      "city": "Lake Nathanielside",
      "state": "UT",
      "zipCode": "73043-1470",

```

```

    "country": "MZ"
  },
  "shippingInfo": {
    "name": "Leatha Swaniawski",
    "address": {
      "address1": "956 Percy Port Spurs",
      "city": "Kuhicport",
      "state": "MS",
      "zipCode": "71913",
      "country": "AX"
    },
    "shippingMethod": "UPS Ground"
  },
  "shippingTotal": 20.36,
  "tax": 4.72,
  "lineItems": [
    {
      "prodId": "181f86cd-49f3-4a4f-8ac6-48e28fce955e",
      "dispName": "Refined Frozen Chicken",
      "shortDescr": "Odit et officiis. Eum voluptatibus voluptatem vel. Non ut officia aut sed. Aut quaerat molestiae sint sit quia ut. Nostrum amet aut occaecati non ex voluptatem in asperiores.",
      "image": "http://lorempixel.com/640/480/transport",
      "price": 35.66,
      "qty": 1,
      "subTotal": 35.66
    }
  ],
  "grandTotal": 60.74
}

```

Swagger UIでの承認

API ロジックの一部をテストするために、一部のエンドポイントではswagger ページを使用して承認が必要になります。 エンドポイントを認証するには、次の手順に従います。

1. [Swagger UI] ページに移動します: <http://localhost:3000/api-docs/>
2. /test/testLogin エンドポイントをクリックします。
3. パネルが展開されたら、[試す] ボタン
4. ユーザー名とパスワードを入力します(ユーザー名とパスワードが登録されていることを確認する手順は、実習1の手順1を参照してください)
5. [実行] ボタンをクリックします。
6. 応答コードは200 にする必要があり、応答本文には `userInfo` オブジェクトを含むデータ プロパティを含め、`userInfo` オブジェクト内はトークンプロパティである必要があります。
7. `token` 値をコピーします。
8. Swagger UI ページの右上隅にある [Authorize] ボタンをクリックします。

9. ポップアップで、(ステップ#7でコピーされた)トークン値を[value]フィールドに貼り付けます。
10. [Authorize]ボタン
11. [close] ボタン
12. Swagger UI ページは、承認を必要とするエンドポイントの横に閉じたロックアイコンを表示するようになりました

ログイン応答の例:

領域を節約するために、*userInfo* オブジェクトと *customerInfo* オブジェクトが削除されました。*userInfo*オブジェクトのトークンプロパティのみが表示されます。

```
{
  "data": {
    "userInfo": {
      ... sample userInfo data ...
      "token":
"eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.eyJpZCI6IjIjMGM0YTM0LTUyOGEtNDg2Yi04ZTM1LTQ0ZjgyNTIwZDBhZCIsIm1hdCI6MTU4OTM4NzcyMX0.1pFgZrr01TBRQFTA_5IUd7TzMoiEhhv0weuCX-70sxk"
    },
    "customerInfo": {
      ... sample customerInfo data ...
    }
  },
  "message": "Successfully logged in and created session.",
  "error": null,
  "authorized": true
}
```