

Sunny Day Continues

Today we have two assignments 😊

Long but easy Lab 😊 😊 😊

### **Assignment A:**

#### **Creating a RESTful web service**

**The ABD Glasses shop grew bigger so the owner decide to create a mobile application beside his web application to manage it's items (show item details, add new item, edit and delete existing items)**

**To do all that we need a RESTFUL web service for the shop which will create all CRUD operations for the items**

**The web service will be used by both imaginary front end and mobile team so we need to do our best so the imaginary teams are happy with the web service they will all use 😊 😊 😊**

#### **Let see how the web service will be used:**

- Handle your webservice to allow only 4 HTTPverbs GET, POST, PUT and DELETE . If another HTTP verb is used it should give status code 405 with the following results :

*{"error": "method not allowed!"}*

- If the database connection failed for any reason, the response should be with status 500 internal server error and the following custom error .

*{"error": "internal server error!"}*

## **1- Get Single Product**

url : [GET] Localhost/GlassShopAPI.php/items/{id}

url example: [GET] Localhost/GlassShopAPI.php/items/100

- If the ID is correct and the verb is GET then the result should be all fields of that item like the following :

```
{  
  "id": "100",  
  "product_code": "NWTCTFV-100",  
  "product_name": "new_glass very new1 ",  
  "photo": "09.png",  
  "list_price": "14.00",  
  "reorder_level": "10",  
  "units_in_stock": "4",  
  "category": "sunglasses",  
  "country": "USA",  
  "rating": "4.60",  
  "discontinued": "1",  
  "date": "2018-08-28 22:52:14"  
}
```

**If the ID is not correct for any reason the API should produce the following response with Result code: 404:**

```
{"error": "Resource doesn't exist"}
```

The same response should be returned if the resource requested is anything other than items .

## **2- Create New Product**

url : [POST] Localhost/GlassShopAPI.php/items

Request :

```
{  
    "id" : 177,  
    "product_name" : "my new glasses"  
}
```

Response status code 201:

```
{  
    "status": "Resource was added successfully!"  
}
```

Adding a wrong column in the request such as

```
{  
    "wrong_column" : "my new glasses 123"  
}
```

Should return 400 status code and

```
{  
    "error": "Bad request"  
}
```

### **3- Editing an existing item**

url : [PUT] Localhost/GlassShopAPI.php/items/{id}

url example: [PUT] Localhost/GlassShopAPI.php/items/100

Request :

```
{  
    "product_name": "my new glasses 123"  
}
```

Response will return the edited resource with status code 200

```
{  
    "id": "177",  
    "product_code": "test",  
    "product_name": "my new glasses 123",  
    "photo": null,  
    "list_price": "20.00", ....  
}
```

Remember that editing a Non existing resource (invalid id)

Should return 404 code and

```
{  
    "error": "Resource not found!"  
}
```

Also adding a wrong column in the request such as

```
{  
    "wrong_column" : "my new glasses 123"  
}
```

Should return 400 status code and

```
{  
    "error": "Bad request"  
}
```

#### **4- Deleting an existing item**

**url : [DELETE] Localhost/GlassShopAPI.php/items/{id}**

**url example : [DELETE] Localhost/GlassShopAPI.php/items/80**

No request body

Response :

```
{  
  "status": "Resource was deleted successfully!"  
}
```

#### **Implementation notes :**

You need the MySQLHandler class in the Resources directory not to loose time working with SQL queries, it has methods for select, update, delete and update so use them , and feel free to change any thing you like .

**Bonus: (Do it after finishing the Assignment B)**

**Use apidoc.js tool, use it to generate Help for your API for your colleagues in the Mobile and front end to understand your API which they will have to consume .**

**>> Generating API help require documenting your API well using the help published online at :  
<http://apidocjs.com/>**

## **Assignment B**

Use the following wsdl file :

<http://api.radioreference.com/soap2/?wsdl&v=latest>

which is for the radio reference API, to get the country list supported in this api  
and their codes both by PHP SOAPClient as follow

**Country : Afghanistan**

Code : Afghanistan

**Country : Albania**

Code : Albania

**Country : Algeria**

Code : Algeria

**Country : American Samoa**

Code : American Samoa

**Country : Andorra**

Code : Andorra

**Country : Angola**

Code : Angola

Good Luck 😊😊