

CODEOFNINJA < <https://www.codeofninja.com/>>

How To Create A Simple REST API in PHP? Step By Step Guide!



By using this site, you agree to our [Privacy Policy](https://www.codeofninja.com/privacy-policy) < <https://www.codeofninja.com/privacy-policy>> and [Terms of Use](https://www.codeofninja.com/terms-of-use) < <https://www.codeofninja.com/terms-of-use>> .

OK



Previously, we learned how to create, read, update and delete database records (CRUD operations) with our [**PHP, MySQL & OOP CRUD Tutorial <**](#)
[**https://www.codeofaninja.com/2014/06/php-object-oriented-crud-example-oop.html>**](https://www.codeofaninja.com/2014/06/php-object-oriented-crud-example-oop.html).

Today, before we go to JavaScript programming, we will learn how to create a simple REST API in PHP. Enjoy our step-by-step tutorial below!

Contents [hide]

- 1 Project Overview**
 - 1.1 What is REST API?**
 - 1.2 Why do we need REST API?**
 - 1.3 Where REST API is used?**
 - 1.4 REST API in our tutorials**
- 2 File structure**
- 3 Setup the database**
 - 3.1 Create categories table**
 - 3.2 Dump data for categories table**

By using this site, you agree to our [Privacy Policy < https://www.codeofaninja.com/privacy-policy>](#) and [Terms of Use < https://www.codeofaninja.com/terms-of-use>](#).

OK

- 4.1 Product object**
- 4.2 Create file to read products**
- 4.3 Connect to database and products table**
- 4.4 Read products from the database**
- 4.5 Add product “read()” method**
- 4.6 Tell the user no products found**
- 4.7 Output**
- 5 Create Product**
 - 5.1 Create create.php file**
 - 5.2 Product create() method**
 - 5.3 Output**
- 6 Read One Product**
 - 6.1 Create read_one.php file**
 - 6.2 Product readOne() method**
 - 6.3 Output**
- 7 Update product**
 - 7.1 Create “update.php” file**
 - 7.2 Product update() method**
 - 7.3 Output**
- 8 Delete Product**
 - 8.1 Create “delete.php” file**
 - 8.2 Product delete() method**
 - 8.3 Output**
- 9 Search Products**
 - 9.1 Create “search.php” file**
 - 9.2 Create search() method**
 - 9.3 Output**
- 10 Paginate Products**
 - 10.1 Create “read_paging.php” file**
 - 10.2 Create “core.php” file**
 - 10.3 Create “readPaging()” method**
 - 10.4 Create “count()” method**
 - 10.5 Get “paging” array**
 - 10.6 Output**
- 11 Read Categories**
 - 11.1 Create “category.php” file**
 - 11.2 Create “read.php” file**
 - 11.3 Add Category “read()” method**

By using this site, you agree to our **Privacy Policy** <<https://www.codeofaninja.com/privacy-policy>> and **Terms of Use** <<https://www.codeofaninja.com/terms-of-use>> .

OK

13 What's Next?**14 Related Tutorials****15 Notes****15.1 Found An Issue?****15.2 Subscribe to CodeOfaNinja****15.3 Thank You!****15.4 Share our tutorial**

Project Overview

What is REST API?

To define "REST API", we have to know what is "REST" and what is "API" first. I'll do my best to explain it in simple terms because REST has a lot of concepts inside of it that could mean a lot of things <

<https://stackoverflow.com/questions/4663927/what-is-rest-slightly-confused> .

REST stands for "REpresentational State Transfer". It is a concept or architecture for managing information over the internet. REST concepts are referred to as resources. A representation of a resource must be stateless. It is usually represented by [JSON](https://stackoverflow.com/questions/383692/what-is-json-and-why-would-i-use-it) < <https://stackoverflow.com/questions/383692/what-is-json-and-why-would-i-use-it> > . This post is worth reading: [How I Explained REST to My Wife?](http://www.looah.com/source/view/2284) < <http://www.looah.com/source/view/2284> >

API stands for "Application Programming Interface". It is a set of rules that allows one piece of software application to talk to another. Those "rules" can include the create, read, update and delete operations.

By using this site, you agree to our [Privacy Policy](https://www.codeofaninja.com/privacy-policy) < <https://www.codeofaninja.com/privacy-policy> > and [Terms of Use](https://www.codeofaninja.com/terms-of-use) < <https://www.codeofaninja.com/terms-of-use> > .

OK

REST API enables your application to cooperate with one or several different applications using REST concepts. If you want to learn more, watch the video below.

By using this site, you agree to our [Privacy Policy](https://www.codeofaninja.com/privacy-policy) < <https://www.codeofaninja.com/privacy-policy>> and [Terms of Use](https://www.codeofaninja.com/terms-of-use) < <https://www.codeofaninja.com/terms-of-use>> .

OK

In many applications, REST API is a need because this is the lightest way to create, read, update or delete information between different applications over the internet or HTTP protocol. This information is presented to the user in an instant especially if you use JavaScript to render the data on a webpage.

Where REST API is used?

REST API can be used by any application that can connect to the internet. If data from an application can be created, read, updated or deleted using another application, it usually means a REST API is used.

REST API in our tutorials

A REST API is needed for our [**AJAX CRUD Tutorial <**](#)
[**https://www.codeofaninja.com/2015/06/php-crud-with-ajax-and-oop.html> .**](https://www.codeofaninja.com/2015/06/php-crud-with-ajax-and-oop.html> .)

But don't mind it for now. We will do it one step at a time. You don't need to learn all of it as well. Just choose what you need to learn.

But one thing is for sure, this source code is good enough and works for our JavaScript tutorials.

File structure

At the end of this tutorial, we will have the following folders and files.

```
|── api/  
    ├── config/  
    |   ├── core.php – file used for core configuration  
    |   └── database.php – file used for connecting to the database.  
    └── objects/  
        └── product.php – contains properties and methods for “product” database
```

By using this site, you agree to our [Privacy Policy < https://www.codeofaninja.com/privacy-policy>](#)
[and Terms of Use < https://www.codeofaninja.com/terms-of-use> .](#)

OK

```
└── product/
    ├── create.php – a file that will accept posted product data to be saved to
    │   the database.
    ├── delete.php – a file that will accept a product ID to delete a database
    │   record.
    ├── read.php – a file that will output JSON data based on “products”
    │   database records.
    ├── read_paging.php – a file that will output “products” JSON data with
    │   pagination.
    ├── read_one.php – a file that will accept product ID to read a record from
    │   the database.
    ├── update.php – a file that will accept a product ID to update a database
    │   record.
    └── search.php – a file that will accept keywords parameter to search
        “products” database.

└── category/
    ├── read.php – a file that will output JSON data based on “categories”
    │   database records.

└── shared/
    └── utilities.php – a file that will return pagination array.
```

Setup the database

Using PhpMyAdmin, create a new `api_db` database. Yes, `api_db` is the database name. After that, run the following SQL queries to create new tables with sample data.

Create categories table

By using this site, you agree to our [Privacy Policy](https://www.codeofaninja.com/privacy-policy) <<https://www.codeofaninja.com/privacy-policy>> and [Terms of Use](https://www.codeofaninja.com/terms-of-use) <<https://www.codeofaninja.com/terms-of-use>> .

OK

```
CREATE TABLE IF NOT EXISTS `categories` (
  `id` int(11) NOT NULL AUTO_INCREMENT,
  `name` varchar(256) NOT NULL,
  `description` text NOT NULL,
  `created` datetime NOT NULL,
  `modified` timestamp NOT NULL DEFAULT CURRENT_TIMESTAMP,
  PRIMARY KEY (`id`)
) ENGINE=InnoDB DEFAULT CHARSET=utf8 AUTO_INCREMENT=19 ;
```

Dump data for categories table

```
INSERT INTO `categories` (`id`, `name`, `description`, `created` ,
(1, 'Fashion', 'Category for anything related to fashion.', '2014-06-01 00:35:07'),
(2, 'Electronics', 'Gadgets, drones and more.', '2014-06-01 00:35:07'),
(3, 'Motors', 'Motor sports and more', '2014-06-01 00:35:07', '2014-06-01 00:35:07'),
(5, 'Movies', 'Movie products.', '0000-00-00 00:00:00', '2016-01-09 02:24:24'),
(6, 'Books', 'Kindle books, audio books and more.', '0000-00-00 00:00:00'),
(13, 'Sports', 'Drop into new winter gear.', '2016-01-09 02:24:24');
```

Products table

```
CREATE TABLE IF NOT EXISTS `products` (
  `id` int(11) NOT NULL AUTO_INCREMENT,
  `name` varchar(32) NOT NULL,
  `description` text NOT NULL,
  `price` decimal(10,0) NOT NULL,
  `category_id` int(11) NOT NULL,
  `created` datetime NOT NULL,
  `modified` timestamp NOT NULL DEFAULT CURRENT_TIMESTAMP,
  PRIMARY KEY (`id`)
) ENGINE=InnoDB DEFAULT CHARSET=latin1 AUTO_INCREMENT=65 ;
```

Dump data for products table

By using this site, you agree to our [Privacy Policy](https://www.codeofaninja.com/privacy-policy) <<https://www.codeofaninja.com/privacy-policy>> and [Terms of Use](https://www.codeofaninja.com/terms-of-use) <<https://www.codeofaninja.com/terms-of-use>> .

OK

```
INSERT INTO `products` (`id`, `name`, `description`, `price`, `category_id`, `image_url`)
VALUES
(1, 'LG P880 4X HD', 'My first awesome phone!', '336', 3, '2014-06-01 01:12:34', 'https://www.codeofaninja.com/images/lg-p880.jpg'),
(2, 'Google Nexus 4', 'The most awesome phone of 2013!', '299', 2, '2014-06-01 01:12:34', 'https://www.codeofaninja.com/images/google-nexus-4.jpg'),
(3, 'Samsung Galaxy S4', 'How about no?', '600', 3, '2014-06-01 01:12:34', 'https://www.codeofaninja.com/images/samsung-galaxy-s4.jpg'),
(6, 'Bench Shirt', 'The best shirt!', '29', 1, '2014-06-01 01:12:34', 'https://www.codeofaninja.com/images/bench-shirt.jpg'),
(7, 'Lenovo Laptop', 'My business partner.', '399', 2, '2014-06-01 01:12:34', 'https://www.codeofaninja.com/images/lenovo-laptop.jpg'),
(8, 'Samsung Galaxy Tab 10.1', 'Good tablet.', '259', 2, '2014-06-01 01:12:34', 'https://www.codeofaninja.com/images/samsung-galaxy-tab-10.1.jpg'),
(9, 'Spalding Watch', 'My sports watch.', '199', 1, '2014-06-01 01:12:34', 'https://www.codeofaninja.com/images/spalding-watch.jpg'),
(10, 'Sony Smart Watch', 'The coolest smart watch!', '300', 2, '2014-06-01 01:12:34', 'https://www.codeofaninja.com/images/sony-smart-watch.jpg'),
(11, 'Huawei Y300', 'For testing purposes.', '100', 2, '2014-06-01 01:12:34', 'https://www.codeofaninja.com/images/huawei-y300.jpg'),
(12, 'Abercrombie Lake Arnold Shirt', 'Perfect as gift!', '60', 1, '2014-06-01 01:12:34', 'https://www.codeofaninja.com/images/abercrombie-lake-arnold-shirt.jpg'),
(13, 'Abercrombie Allen Brook Shirt', 'Cool red shirt!', '70', 1, '2014-06-01 01:12:34', 'https://www.codeofaninja.com/images/abercrombie-allen-brook-shirt.jpg'),
(26, 'Another product', 'Awesome product!', '555', 2, '2014-11-22 11:11:11', 'https://www.codeofaninja.com/images/another-product.jpg'),
(28, 'Wallet', 'You can absolutely use this one!', '799', 6, '2014-11-22 11:11:11', 'https://www.codeofaninja.com/images/wallet.jpg'),
(31, 'Amanda Waller Shirt', 'New awesome shirt!', '333', 1, '2014-11-22 11:11:11', 'https://www.codeofaninja.com/images/amanda-waller-shirt.jpg'),
(42, 'Nike Shoes for Men', 'Nike Shoes', '12999', 3, '2015-12-12 12:12:12', 'https://www.codeofaninja.com/images/nike-shoes-for-men.jpg'),
(48, 'Bristol Shoes', 'Awesome shoes.', '999', 5, '2016-01-08 06:06:06', 'https://www.codeofaninja.com/images/bristol-shoes.jpg'),
(60, 'Rolex Watch', 'Luxury watch.', '25000', 1, '2016-01-11 15:15:15', 'https://www.codeofaninja.com/images/rolex-watch.jpg');
```

Connect to database

The code below shows the database credentials and a method to get a database connection using PDO. If you're not yet familiar with PDO, please learn from our

[PHP OOP CRUD Tutorial < https://www.codeofaninja.com/2014/06/php-object-oriented-crud-example-oop.html >](https://www.codeofaninja.com/2014/06/php-object-oriented-crud-example-oop.html) first.

- Create api folder. Open api folder.
- Create config folder. Open config folder.
- Create a database.php file. Place the following code inside it.

By using this site, you agree to our [Privacy Policy < https://www.codeofaninja.com/privacy-policy >](https://www.codeofaninja.com/privacy-policy) and [Terms of Use < https://www.codeofaninja.com/terms-of-use >](https://www.codeofaninja.com/terms-of-use).

OK

```

<?php
class Database{

    // specify your own database credentials
    private $host = "localhost";
    private $db_name = "api_db";
    private $username = "root";
    private $password = "";
    public $conn;

    // get the database connection
    public function getConnection(){

        $this->conn = null;

        try{
            $this->conn = new PDO("mysql:host=" . $this->host . "
                $this->conn->exec("set names utf8");
            }catch(PDOException $exception){
                echo "Connection error: " . $exception->getMessage();
            }

            return $this->conn;
        }
    }
?>

```

Read products

Product object

The code below shows a class named Product with several of its properties. It also shows a constructor method that will accept the database connection.

We will use this class to read data from the database.

- Open api folder.

By using this site, you agree to our **Privacy Policy** <<https://www.codeofaninja.com/privacy-policy>>
and **Terms of Use** <<https://www.codeofaninja.com/terms-of-use>> .

OK

```
<?php
class Product{

    // database connection and table name
    private $conn;
    private $table_name = "products";

    // object properties
    public $id;
    public $name;
    public $description;
    public $price;
    public $category_id;
    public $category_name;
    public $created;

    // constructor with $db as database connection
    public function __construct($db){
        $this->conn = $db;
    }
}
?>
```

Create file to read products

The code below shows headers about who can read this file and which type of content it will return.

In this case, our `read.php` file can be read by anyone (asterisk * means all) and will return a data in [JSON format < https://www.json.org/>](https://www.json.org/).

- Open api folder.
- Create product folder.
- Open product folder.
- Create `read.php` file.
- Place the following code inside it.

By using this site, you agree to our [Privacy Policy < https://www.codeofaninja.com/privacy-policy>](https://www.codeofaninja.com/privacy-policy) and [Terms of Use < https://www.codeofaninja.com/terms-of-use>](https://www.codeofaninja.com/terms-of-use).

OK

```
<?php
// required headers
header("Access-Control-Allow-Origin: *");
header("Content-Type: application/json; charset=UTF-8");

// database connection will be here
```

Connect to database and products table

In the code below, we included the `database.php` and `product.php` files. These are the files we created earlier.

We need to use the `getConnection()` method of the `Database` class to get a database connection. We pass this connection to the `Product` class.

Replace of `// database connection will be here` comment of `read.php` file with the following code.

```
// include database and object files
include_once '../config/database.php';
include_once '../objects/product.php';

// instantiate database and product object
$database = new Database();
$db = $database->getConnection();

// initialize object
$product = new Product($db);

// read products will be here
```

Read products from the database

In the code below, we used the `read()` method of `Product` class to read data

By using this site, you agree to our [Privacy Policy](https://www.codeofaninja.com/privacy-policy) <<https://www.codeofaninja.com/privacy-policy>>
and [Terms of Use](https://www.codeofaninja.com/terms-of-use) <<https://www.codeofaninja.com/terms-of-use>> .

OK

If there are records found, we loop through it using the `while` loop, add each record to the `$products_arr` array, set a `200 OK` response code and show it to the user in JSON format.

Replace of `// read products will be here` comment of `read.php` file with the following code.

By using this site, you agree to our **Privacy Policy** <<https://www.codeofaninja.com/privacy-policy>> and **Terms of Use** <<https://www.codeofaninja.com/terms-of-use>> .

OK

```
// query products
$stmt = $product->read();
$num = $stmt->rowCount();

// check if more than 0 record found
if($num>0){

    // products array
    $products_arr=array();
    $products_arr["records"]=array();

    // retrieve our table contents
    // fetch() is faster than fetchAll()
    // http://stackoverflow.com/questions/2770630/pdofetchall-vs-
    while ($row = $stmt->fetch(PDO::FETCH_ASSOC)){
        // extract row
        // this will make $row['name'] to
        // just $name only
        extract($row);

        $product_item=array(
            "id" => $id,
            "name" => $name,
            "description" => html_entity_decode($description),
            "price" => $price,
            "category_id" => $category_id,
            "category_name" => $category_name
        );
        array_push($products_arr["records"], $product_item);
    }

    // set response code - 200 OK
    http_response_code(200);

    // show products data in json format
    echo json_encode($products_arr);
}

// no products found will be here
```

By using this site, you agree to our [Privacy Policy](https://www.codeofaninja.com/privacy-policy) <<https://www.codeofaninja.com/privacy-policy>>
and [Terms of Use](https://www.codeofaninja.com/terms-of-use) <<https://www.codeofaninja.com/terms-of-use>> .

OK

We used the `read()` method in the previous section but it does not exist yet in the `Product` class. We need to add this `read()` method. The code below shows the query to get records from the database.

- Open objects folder.
- Open `product.php` file.
- Place the following code inside the `Product` class.
- To make sure you added it correctly, place the code before the last closing curly brace.

```
// read products
function read(){

    // select all query
    $query = "SELECT
        c.name as category_name, p.id, p.name, p.description
    FROM
        " . $this->table_name . " p
    LEFT JOIN
        categories c
    ON p.category_id = c.id
    ORDER BY
        p.created DESC";

    // prepare query statement
    $stmt = $this->conn->prepare($query);

    // execute query
    $stmt->execute();

    return $stmt;
}
```

Tell the user no products found

If the `$num` variable has a value of zero or negative, it means there are no records

By using this site, you agree to our [Privacy Policy](https://www.codeofaninja.com/privacy-policy) <<https://www.codeofaninja.com/privacy-policy>>
and [Terms of Use](https://www.codeofaninja.com/terms-of-use) <<https://www.codeofaninja.com/terms-of-use>> .

OK

On the code below, we set the response code to 404 - Not found and a message that says No products found.

Replace of // no products found will be here comment of read.php file with the following code.

```
else{

    // set response code - 404 Not found
    http_response_code(404);

    // tell the user no products found
    echo json_encode(
        array("message" => "No products found.")
    );
}
```

Output

You need to use **[POSTMAN < https://www.getpostman.com/ >](https://www.getpostman.com/)** to test our API. Download your version of POSTMAN **[here < https://www.getpostman.com/apps >](https://www.getpostman.com/apps)**.

Launch POSTMAN. Enter the following as the request URL.



Click the blue "Send" button.

- Output if there are product data.

By using this site, you agree to our [Privacy Policy < https://www.codeofaninja.com/privacy-policy >](https://www.codeofaninja.com/privacy-policy) and [Terms of Use < https://www.codeofaninja.com/terms-of-use >](https://www.codeofaninja.com/terms-of-use).

OK

The screenshot shows a Postman interface with a POST request to `http://localhost/api/product/read.php`. The response status is 200 OK, and the response body is a JSON object with the following structure:

```

1  {
2   "records": [
3     {
4       "id": "60",
5       "name": "Rolex Watch Men",
6       "description": "Luxury watch.",
7       "price": "25000",
8       "category_id": "1",
9       "category_name": "Fashion"
10    },
11    {
12      "id": "48",
13      "name": "Bristol Shoes Men",
14      "description": "Awesome shoes. Seen in movies.",
15      "price": "999",
16      "category_id": "5",
17      "category_name": "Movies"
18    },
19    {
20      "id": "42",
21      "name": "Nike Shoes for Men",
22      "description": "Nike Shoes",
23      "price": "12999",
24      "category_id": "3",
25      "category_name": "Motors"
26    }
].

```

< https://www.codeofaninja.com/wp-content/uploads/2017/02/read-products-200.jpg>

- Output if there are no product data.

The screenshot shows a Postman interface with a POST request to `http://localhost/api/product/read.php`. The response status is 404 Not Found, and the response body is a JSON object with the following structure:

```

1  {
2   "message": "No products found."
3 }

```

< https://www.codeofaninja.com/wp-content/uploads/2017/02/read-products-404.jpg>

Create Product

Create create.php file

- Open product folder.
- Create a new `create.php` file.
- Open that file and put the following code inside it.

By using this site, you agree to our [Privacy Policy](#) < https://www.codeofaninja.com/privacy-policy>
and [Terms of Use](#) < https://www.codeofaninja.com/terms-of-use> .

OK

```

<?php
// required headers
header("Access-Control-Allow-Origin: *");
header("Content-Type: application/json; charset=UTF-8");
header("Access-Control-Allow-Methods: POST");
header("Access-Control-Max-Age: 3600");
header("Access-Control-Allow-Headers: Content-Type, Access-Control-Allow-Headers");

// get database connection
include_once '../config/database.php';

// instantiate product object
include_once '../objects/product.php';

$database = new Database();
$db = $database->getConnection();

$product = new Product($db);

// get posted data
$data = json_decode(file_get_contents("php://input"));

// make sure data is not empty
if(
    !empty($data->name) &&
    !empty($data->price) &&
    !empty($data->description) &&
    !empty($data->category_id)
){

    // set product property values
    $product->name = $data->name;
    $product->price = $data->price;
    $product->description = $data->description;
    $product->category_id = $data->category_id;
    $product->created = date('Y-m-d H:i:s');

    // create the product
    if($product->create()){

        // set response code - 201 created
        http_response_code(201);
    }
}

```

By using this site, you agree to our [Privacy Policy](https://www.codeofaninja.com/privacy-policy) <<https://www.codeofaninja.com/privacy-policy>>
and [Terms of Use](https://www.codeofaninja.com/terms-of-use) <<https://www.codeofaninja.com/terms-of-use>> .

OK

```
// if unable to create the product, tell the user
else{

    // set response code - 503 service unavailable
    http_response_code(503);

    // tell the user
    echo json_encode(array("message" => "Unable to create product"));
}

// tell the user data is incomplete
else{

    // set response code - 400 bad request
    http_response_code(400);

    // tell the user
    echo json_encode(array("message" => "Unable to create product"));
}
?>
```

Product create() method

- Open objects folder.
- Open product.php file.
- The previous section will not work without the following code inside the Product (objects/product.php) class.

By using this site, you agree to our [Privacy Policy](https://www.codeofaninja.com/privacy-policy) <<https://www.codeofaninja.com/privacy-policy>> and [Terms of Use](https://www.codeofaninja.com/terms-of-use) <<https://www.codeofaninja.com/terms-of-use>> .

OK

```
// create product
function create(){

    // query to insert record
    $query = "INSERT INTO
        " . $this->table_name . "
    SET
        name=:name, price=:price, description=:description

    // prepare query
    $stmt = $this->conn->prepare($query);

    // sanitize
    $this->name=htmlspecialchars(strip_tags($this->name));
    $this->price=htmlspecialchars(strip_tags($this->price));
    $this->description=htmlspecialchars(strip_tags($this->description));
    $this->category_id=htmlspecialchars(strip_tags($this->category_id));
    $this->created=htmlspecialchars(strip_tags($this->created));

    // bind values
    $stmt->bindParam(":name", $this->name);
    $stmt->bindParam(":price", $this->price);
    $stmt->bindParam(":description", $this->description);
    $stmt->bindParam(":category_id", $this->category_id);
    $stmt->bindParam(":created", $this->created);

    // execute query
    if($stmt->execute()){
        return true;
    }

    return false;

}
```

Output

To test for the successful creation of a product, open POSTMAN. Enter the following as the request URL

By using this site, you agree to our [Privacy Policy](https://www.codeofaninja.com/privacy-policy) <<https://www.codeofaninja.com/privacy-policy>>
and [Terms of Use](https://www.codeofaninja.com/terms-of-use) <<https://www.codeofaninja.com/terms-of-use>> .

OK

- Click the "Body" tab.
- Click "raw".
- Enter this JSON value:

```
{
  "name" : "Amazing Pillow 2.0",
  "price" : "199",
  "description" : "The best pillow for amazing programmers.",
  "category_id" : 2,
  "created" : "2018-06-01 00:35:07"
}
```

- It should look like this:

The screenshot shows the Postman interface. The top bar has 'POST' selected, the URL is 'http://localhost/api/product/create.php', and the 'Send' button is highlighted. Below the URL, the 'Body' tab is active, showing the JSON payload. The 'Text' dropdown is selected. The JSON content is:

```

1 {
2   "name" : "Amazing Pillow 2.0",
3   "price" : "199",
4   "description" : "The best pillow for amazing programmers.",
5   "category_id" : 2,
6   "created" : "2018-06-01 00:35:07"
7 }

```

At the bottom, the 'Body' tab is active, showing the JSON response:

```

1 {
2   "message": "Product was created."
3 }

```

The status bar at the bottom right indicates 'Status: 201 Created' and 'Time: 23 ms'.

< https://www.codeofaninja.com/wp-content/uploads/2017/02/create-product-201.jpg>

- If the system is unable to create the product, it should look like this:

By using this site, you agree to our [Privacy Policy](https://www.codeofaninja.com/privacy-policy) < https://www.codeofaninja.com/privacy-policy>
and [Terms of Use](https://www.codeofaninja.com/terms-of-use) < https://www.codeofaninja.com/terms-of-use> .

OK

POST http://localhost/api/product/create.php

Body

```
{
  "name": "Amazing Pillow 2.0",
  "price": "199",
  "description": "The best pillow for amazing programmers.",
  "category_id": 2,
  "created": "2018-06-01 00:35:07"
}
```

Status: 503 Service Unavailable Time: 23 ms Size: 484 B

503 Service Unavailable

The server is currently unavailable (because it is overloaded or down for maintenance). Generally, this is a temporary state.

< <https://www.codeofaninja.com/wp-content/uploads/2017/02/create-product-503.jpg> >

- If the sent data is incomplete, for example, it is missing the price data, the output should look like this:

POST http://localhost/api/product/create.php

Body

```
{
  "name": "Amazing Pillow 2.0",
  "description": "The best pillow for amazing programmers.",
  "category_id": 2,
  "created": "2018-06-01 00:35:07"
}
```

Status: 400 Bad Request Time: 20 ms Size: 496 B

400 Bad Request

The request cannot be fulfilled due to bad syntax.

Read One Product

By using this site, you agree to our [Privacy Policy](#) < <https://www.codeofaninja.com/privacy-policy> >
and [Terms of Use](#) < <https://www.codeofaninja.com/terms-of-use> > .

OK

- Open that file and put the following code.

By using this site, you agree to our **Privacy Policy** <<https://www.codeofaninja.com/privacy-policy>> and **Terms of Use** <<https://www.codeofaninja.com/terms-of-use>> .

OK

```

<?php
// required headers
header("Access-Control-Allow-Origin: *");
header("Access-Control-Allow-Headers: access");
header("Access-Control-Allow-Methods: GET");
header("Access-Control-Allow-Credentials: true");
header('Content-Type: application/json');

// include database and object files
include_once '../config/database.php';
include_once '../objects/product.php';

// get database connection
$database = new Database();
$db = $database->getConnection();

// prepare product object
$product = new Product($db);

// set ID property of record to read
$product->id = isset($_GET['id']) ? $_GET['id'] : die();

// read the details of product to be edited
$product->readOne();

if($product->name!=null){
    // create array
    $product_arr = array(
        "id" => $product->id,
        "name" => $product->name,
        "description" => $product->description,
        "price" => $product->price,
        "category_id" => $product->category_id,
        "category_name" => $product->category_name
    );
}

// set response code - 200 OK
http_response_code(200);

// make it json format
echo json_encode($product);

```

By using this site, you agree to our [Privacy Policy](https://www.codeofaninja.com/privacy-policy) <<https://www.codeofaninja.com/privacy-policy>>
and [Terms of Use](https://www.codeofaninja.com/terms-of-use) <<https://www.codeofaninja.com/terms-of-use>> .

OK

```
http_response_code(404);

// tell the user product does not exist
echo json_encode(array("message" => "Product does not exist."
})

?>
```

Product readOne() method

- Open objects folder.
- Open product.php file.
- The previous section will not work without the following code inside the Product class.

By using this site, you agree to our [Privacy Policy](https://www.codeofaninja.com/privacy-policy) <<https://www.codeofaninja.com/privacy-policy>> and [Terms of Use](https://www.codeofaninja.com/terms-of-use) <<https://www.codeofaninja.com/terms-of-use>> .

OK

```

// used when filling up the update product form
function readOne(){

    // query to read single record
    $query = "SELECT
        c.name AS category_name, p.id, p.name, p.description
    FROM
        " . $this->table_name . " p
    LEFT JOIN
        categories c
        ON p.category_id = c.id
    WHERE
        p.id = ?
    LIMIT
        0,1";

    // prepare query statement
    $stmt = $this->conn->prepare( $query );

    // bind id of product to be updated
    $stmt->bindParam(1, $this->id);

    // execute query
    $stmt->execute();

    // get retrieved row
    $row = $stmt->fetch(PDO::FETCH_ASSOC);

    // set values to object properties
    $this->name = $row['name'];
    $this->price = $row['price'];
    $this->description = $row['description'];
    $this->category_id = $row['category_id'];
    $this->category_name = $row['category_name'];
}


```

Output

- First, we will test for a product that exists. Open POSTMAN. Enter the following as the request URL. Click the blue "Send" button.

By using this site, you agree to our Privacy Policy <<https://www.codeofaninja.com/privacy-policy>> and Terms of Use <<https://www.codeofaninja.com/terms-of-use>>.

OK

POST http://localhost/api/product/read_one.php?id=60

Body Headers (11) Test Results

Pretty Raw Preview JSON

```

1 {
  "id": "60",
  "name": "Rolex Watch Men",
  "description": "Luxury watch.",
  "price": "25000",
  "category_id": "1",
  "category_name": "Fashion"
}

```

Status: 200 OK Time: 28 ms Size: 517 B

200 OK

Standard response for successful HTTP requests. The actual response will depend on the request method used. In a GET request, the response will contain an entity corresponding to the requested resource. In a POST request the response will contain an entity describing or containing the result of the action.

< https://www.codeofaninja.com/wp-content/uploads/2017/02/read-one-200.jpg>

- Next, we will test for a product that does not exist. Enter the following as the request URL. Click the blue “Send” button.

POST http://localhost/api/product/read_one.php?id=999

Body Headers (11) Test Results

Pretty Raw Preview JSON

```

1 {
  "message": "Product does not exist."
}

```

Status: 404 Not Found Time: 22 ms Size: 434 B

404 Not Found

The requested resource could not be found but may be available again in the future. Subsequent requests by the client are permissible.

< https://www.codeofaninja.com/wp-content/uploads/2017/02/read-one-404.jpg>

Update product

Create “update.php” file

- Open product folder.
- Create a new update.php file.
- Open that file and put the following code inside it.

By using this site, you agree to our [Privacy Policy](https://www.codeofaninja.com/privacy-policy) < https://www.codeofaninja.com/privacy-policy> and [Terms of Use](https://www.codeofaninja.com/terms-of-use) < https://www.codeofaninja.com/terms-of-use> .

OK

```

<?php
// required headers
header("Access-Control-Allow-Origin: *");
header("Content-Type: application/json; charset=UTF-8");
header("Access-Control-Allow-Methods: POST");
header("Access-Control-Max-Age: 3600");
header("Access-Control-Allow-Headers: Content-Type, Access-Control-Allow-Headers");

// include database and object files
include_once '../config/database.php';
include_once '../objects/product.php';

// get database connection
$database = new Database();
$db = $database->getConnection();

// prepare product object
$product = new Product($db);

// get id of product to be edited
$data = json_decode(file_get_contents("php://input"));

// set ID property of product to be edited
$product->id = $data->id;

// set product property values
$product->name = $data->name;
$product->price = $data->price;
$product->description = $data->description;
$product->category_id = $data->category_id;

// update the product
if($product->update()){

    // set response code - 200 ok
    http_response_code(200);

    // tell the user
    echo json_encode(array("message" => "Product was updated."));
}

// if unable to update the product tell the user

```

By using this site, you agree to our [Privacy Policy](https://www.codeofaninja.com/privacy-policy) <<https://www.codeofaninja.com/privacy-policy>>
and [Terms of Use](https://www.codeofaninja.com/terms-of-use) <<https://www.codeofaninja.com/terms-of-use>> .

OK

```
// tell the user
echo json_encode(array("message" => "Unable to update product
})
?>
```

Product update() method

- Open objects folder.
- Open product.php file.
- The previous section will not work without the following code inside the Product class.

By using this site, you agree to our [Privacy Policy](https://www.codeofaninja.com/privacy-policy) <<https://www.codeofaninja.com/privacy-policy>> and [Terms of Use](https://www.codeofaninja.com/terms-of-use) <<https://www.codeofaninja.com/terms-of-use>> .

OK

```

// update the product
function update(){

    // update query
    $query = "UPDATE
        " . $this->table_name . "
        SET
            name = :name,
            price = :price,
            description = :description,
            category_id = :category_id
        WHERE
            id = :id";

    // prepare query statement
    $stmt = $this->conn->prepare($query);

    // sanitize
    $this->name=htmlspecialchars(strip_tags($this->name));
    $this->price=htmlspecialchars(strip_tags($this->price));
    $this->description=htmlspecialchars(strip_tags($this->description));
    $this->category_id=htmlspecialchars(strip_tags($this->category_id));
    $this->id=htmlspecialchars(strip_tags($this->id));

    // bind new values
    $stmt->bindParam(':name', $this->name);
    $stmt->bindParam(':price', $this->price);
    $stmt->bindParam(':description', $this->description);
    $stmt->bindParam(':category_id', $this->category_id);
    $stmt->bindParam(':id', $this->id);

    // execute the query
    if($stmt->execute()){
        return true;
    }

    return false;
}

```

Output

By using this site, you agree to our [Privacy Policy](https://www.codeofaninja.com/privacy-policy) <<https://www.codeofaninja.com/privacy-policy>>
 and [Terms of Use](https://www.codeofaninja.com/terms-of-use) <<https://www.codeofaninja.com/terms-of-use>> .

OK

<http://localhost/api/product/update.php> < <http://localhost/api/pr>

- Click the "Body" tab.
- Click "raw".
- Enter the following JSON value.
- Make sure the ID exists in your database.
- Click the blue "Send" button.

```
{  
    "id" : "106",  
    "name" : "Amazing Pillow 3.0",  
    "price" : "255",  
    "description" : "The best pillow for amazing programmers.",  
    "category_id" : 2,  
    "created" : "2018-08-01 00:35:07"  
}
```

The product ID 106, is just an example. You need to specify a product ID that exists in your database.

If you specify an ID that does not exist in the database, it might still say that the product was updated. It does not update anything on the database but the query was executed successfully without any syntax errors.

To prevent this, you need an extra validation where you check if an ID exists in the database. This feature is not yet part of our tutorial.

- If updating a product is successful, it should look like this:

By using this site, you agree to our [Privacy Policy](#) <<https://www.codeofaninja.com/privacy-policy>>
and [Terms of Use](#) <<https://www.codeofaninja.com/terms-of-use>> .

OK

POST http://localhost/api/product/update.php

Body (Text)

```

1 {
2   "id" : "106",
3   "name" : "Amazing Pillow 3.0",
4   "price" : "255",
5   "description" : "The best pillow for amazing programmers.",
6   "category_id" : 2,
7   "created" : "2018-08-01 00:35:07"
8 }

```

Body Cookies Headers (11) Test Results

Status: 200 OK Time: 22 ms Size: 499 B

200 OK

Standard response for successful HTTP requests. The actual response will depend on the request method used. In a GET request, the response will contain an entity corresponding to the requested resource. In a POST request the response will contain an entity describing or containing the result of the action.

< https://www.codeofaninja.com/wp-content/uploads/2017/02/update-product-200.jpg >

- If the system fails to update the product, the output will look like this:

POST http://localhost/api/product/update.php

Body (Text)

```

1 {
2   "id" : "106",
3   "name" : "Amazing Pillow 3.0",
4   "price" : "255",
5   "description" : "The best pillow for amazing programmers.",
6   "category_id" : 2,
7   "created" : "2018-08-01 00:35:07"
8 }

```

Body Cookies Headers (10) Test Results

Status: 503 Service Unavailable Time: 21 ms Size: 484 B

503 Service Unavailable

The server is currently unavailable (because it is overloaded or down for maintenance). Generally, this is a temporary state.

By using this site, you agree to our [Privacy Policy](https://www.codeofaninja.com/privacy-policy) < https://www.codeofaninja.com/privacy-policy >
and [Terms of Use](https://www.codeofaninja.com/terms-of-use) < https://www.codeofaninja.com/terms-of-use > .

OK

Create "delete.php" file

- Open product folder.
- Create new delete.php file.
- Open that file and put the following code inside it.

By using this site, you agree to our **Privacy Policy** <<https://www.codeofaninja.com/privacy-policy>> and **Terms of Use** <<https://www.codeofaninja.com/terms-of-use>> .

OK

```

<?php
// required headers
header("Access-Control-Allow-Origin: *");
header("Content-Type: application/json; charset=UTF-8");
header("Access-Control-Allow-Methods: POST");
header("Access-Control-Max-Age: 3600");
header("Access-Control-Allow-Headers: Content-Type, Access-Control-Allow-Headers");

// include database and object file
include_once '../config/database.php';
include_once '../objects/product.php';

// get database connection
$database = new Database();
$db = $database->getConnection();

// prepare product object
$product = new Product($db);

// get product id
$data = json_decode(file_get_contents("php://input"));

// set product id to be deleted
$product->id = $data->id;

// delete the product
if($product->delete()){

    // set response code - 200 ok
    http_response_code(200);

    // tell the user
    echo json_encode(array("message" => "Product was deleted."));
}

// if unable to delete the product
else{

    // set response code - 503 service unavailable
    http_response_code(503);

    // tell the user
}

```

By using this site, you agree to our [Privacy Policy](https://www.codeofaninja.com/privacy-policy) <<https://www.codeofaninja.com/privacy-policy>>
and [Terms of Use](https://www.codeofaninja.com/terms-of-use) <<https://www.codeofaninja.com/terms-of-use>> .

OK

Product delete() method

- Open objects folder.
- Open product.php file.
- The previous section will not work without the following code inside the Product class.

```
// delete the product
function delete(){

    // delete query
    $query = "DELETE FROM " . $this->table_name . " WHERE id = ?"

    // prepare query
    $stmt = $this->conn->prepare($query);

    // sanitize
    $this->id=htmlspecialchars(strip_tags($this->id));

    // bind id of record to delete
    $stmt->bindParam(1, $this->id);

    // execute query
    if($stmt->execute()){
        return true;
    }

    return false;
}
```

Output

Open POSTMAN. Enter the following as the request URL.

```
http://localhost/api/product/delete.php < http://localhost/api/pr
```

By using this site, you agree to our [Privacy Policy](https://www.codeofaninja.com/privacy-policy) < <https://www.codeofaninja.com/privacy-policy>>
and [Terms of Use](https://www.codeofaninja.com/terms-of-use) < <https://www.codeofaninja.com/terms-of-use>> .

OK

- Click the blue “Send” button.

```
{
  "id" : "106"
}
```

- If a product was successfully deleted, it should look like this:

The screenshot shows the Postman interface. The top bar has 'POST' selected, the URL is 'http://localhost/api/product/delete.php', and the 'Body' tab is active with 'Text' selected. The request body contains the following JSON:

```

1 {
2   "id" : "106"
3 }

```

The response section shows a 200 OK status with the message: "Standard response for successful HTTP requests. The actual response will depend on the request method used. In a GET request, the response will contain an entity corresponding to the requested resource. In a POST request the response will contain an entity describing or containing the result of the action." Below the message, the response body is shown as:

```

1 {
2   "message": "Product was deleted."
3 }

```

<https://www.codeofaninja.com/wp-content/uploads/2017/02/delete-200.jpg>

- If the system fails to delete the product, the output will look like this:

By using this site, you agree to our [Privacy Policy](https://www.codeofaninja.com/privacy-policy) <https://www.codeofaninja.com/privacy-policy> and [Terms of Use](https://www.codeofaninja.com/terms-of-use) <https://www.codeofaninja.com/terms-of-use> .

OK

The screenshot shows a Postman interface. At the top, it says "POST" and "http://localhost/api/product/delete.php". Below that, under the "Body" tab, there is a JSON payload:

```

1 {
2   "id" : "106"
3 }

```

On the right, the response details are shown:

- Status: 503 Service Unavailable
- Time: 41 ms
- Size: 484 B

The response body is:

```

503 Service Unavailable
The server is currently unavailable (because it is overloaded or down for maintenance). Generally, this is a temporary state.

```

< <https://www.codeofaninja.com/wp-content/uploads/2017/02/delete-503-1.jpg> >

Search Products

Create "search.php" file

- Open product folder.
- Create a search.php file.
- Open that file and place the following code.

By using this site, you agree to our [Privacy Policy](https://www.codeofaninja.com/privacy-policy) < <https://www.codeofaninja.com/privacy-policy> >
and [Terms of Use](https://www.codeofaninja.com/terms-of-use) < <https://www.codeofaninja.com/terms-of-use> > .

OK

```

<?php
// required headers
header("Access-Control-Allow-Origin: *");
header("Content-Type: application/json; charset=UTF-8");

// include database and object files
include_once '../config/core.php';
include_once '../config/database.php';
include_once '../objects/product.php';

// instantiate database and product object
$database = new Database();
$db = $database->getConnection();

// initialize object
$product = new Product($db);

// get keywords
$keywords=isset($_GET["s"]) ? $_GET["s"] : "";

// query products
$stmt = $product->search($keywords);
$num = $stmt->rowCount();

// check if more than 0 record found
if($num>0){

    // products array
    $products_arr=array();
    $products_arr["records"]=array();

    // retrieve our table contents
    // fetch() is faster than fetchAll()
    // http://stackoverflow.com/questions/2770630/pdofetchall-vs-
    while ($row = $stmt->fetch(PDO::FETCH_ASSOC)){
        // extract row
        // this will make $row['name'] to
        // just $name only
        extract($row);

        $product_item=array(
            "id" -- <id

```

By using this site, you agree to our [Privacy Policy](https://www.codeofaninja.com/privacy-policy) <<https://www.codeofaninja.com/privacy-policy>>
and [Terms of Use](https://www.codeofaninja.com/terms-of-use) <<https://www.codeofaninja.com/terms-of-use>> .

OK

```
        "category_name" => $category_name
    );
    array_push($products_arr["records"], $product_item);
}

// set response code - 200 OK
http_response_code(200);

// show products data
echo json_encode($products_arr);
}

else{
    // set response code - 404 Not found
    http_response_code(404);

    // tell the user no products found
    echo json_encode(
        array("message" => "No products found.")
    );
}
?>
```

Create search() method

- Open objects folder.
- Open product.php file.
- Add the following search() method.

By using this site, you agree to our [Privacy Policy](https://www.codeofaninja.com/privacy-policy) <<https://www.codeofaninja.com/privacy-policy>>
and [Terms of Use](https://www.codeofaninja.com/terms-of-use) <<https://www.codeofaninja.com/terms-of-use>> .

OK

```

// search products
function search($keywords){

    // select all query
    $query = "SELECT
        c.name AS category_name, p.id, p.name, p.description
    FROM
        " . $this->table_name . " p
    LEFT JOIN
        categories c
        ON p.category_id = c.id
    WHERE
        p.name LIKE ? OR p.description LIKE ? OR c.name LIKE ?
    ORDER BY
        p.created DESC";

    // prepare query statement
    $stmt = $this->conn->prepare($query);

    // sanitize
    $keywords=htmlspecialchars(strip_tags($keywords));
    $keywords = "%{$keywords}%";

    // bind
    $stmt->bindParam(1, $keywords);
    $stmt->bindParam(2, $keywords);
    $stmt->bindParam(3, $keywords);

    // execute query
    $stmt->execute();

    return $stmt;
}

```

Output

Open POSTMAN. Enter the following as the request URL.

<http://localhost/api/product/search.php?c-shirt> < <http://localhost/api/product/search.php?c-shirt>

By using this site, you agree to our [Privacy Policy](https://www.codeofaninja.com/privacy-policy) < <https://www.codeofaninja.com/privacy-policy>>
and [Terms of Use](https://www.codeofaninja.com/terms-of-use) < <https://www.codeofaninja.com/terms-of-use>> .

OK

- If there was a product found, it should look like this:

The screenshot shows a Postman interface with a POST request to `http://localhost/api/product/search.php?&s=shirt`. The response status is **200 OK**. The response body is a JSON object with the following structure:

```

1 | {
2 |   "records": [
3 |     {
4 |       "id": "31",
5 |       "name": "Amanda Waller Shirt Men",
6 |       "description": "New awesome shirt!",
7 |       "price": "333",
8 |       "category_id": "1",
9 |       "category_name": "Fashion"
10|     },
11|     {
12|       "id": "13",
13|       "name": "Abercrombie Allen Brook Shirt",
14|       "description": "Cool red shirt!",
15|       "price": "70",
16|       "category_id": "1",
17|       "category_name": "Fashion"
18|     },
19|     {
20|       "id": "12",
21|       "name": "Abercrombie Lake Arnold Shirt",
22|       "description": "Perfect as gift!",
23|       "price": "60",
24|       "category_id": "1",
25|       "category_name": "Fashion"
26|     },
27|     {
28|       "id": "6",
29|       "name": "Bench Shirt",
30|     }
31|   ]
32| }

```

The response message states: "Standard response for successful HTTP requests. The actual response will depend on the request method used. In a GET request, the response will contain an entity corresponding to the requested resource. In a POST request the response will contain an entity describing or containing the result of the action."

< https://www.codeofaninja.com/wp-content/uploads/2017/02/search-200.jpg>

- If there are no products found, the output will look like this:

The screenshot shows a Postman interface with a POST request to `http://localhost/api/product/search.php?&s=shirt`. The response status is **404 Not Found**. The response body is a JSON object with the following structure:

```

1 | {
2 |   "message": "No products found."
3 | }

```

The response message states: "The requested resource could not be found but may be available again in the future. Subsequent requests by the client are permissible."

< https://www.codeofaninja.com/wp-content/uploads/2017/02/search-404.jpg>

Paginate Products

Create "read_paging.php" file

By using this site, you agree to our [Privacy Policy](#) < <https://www.codeofaninja.com/privacy-policy> >
and [Terms of Use](#) < <https://www.codeofaninja.com/terms-of-use> > .

OK

```
<?php
// required headers
header("Access-Control-Allow-Origin: *");
header("Content-Type: application/json; charset=UTF-8");

// include database and object files
include_once '../config/core.php';
include_once '../shared/utilities.php';
include_once '../config/database.php';
include_once '../objects/product.php';

// utilities
$utilities = new Utilities();

// instantiate database and product object
$database = new Database();
$db = $database->getConnection();

// initialize object
$product = new Product($db);

// query products
$stmt = $product->readPaging($from_record_num, $records_per_page);
$num = $stmt->rowCount();

// check if more than 0 record found
if($num>0){

    // products array
    $products_arr=array();
    $products_arr["records"]=array();
    $products_arr["paging"]=array();

    // retrieve our table contents
    // fetch() is faster than fetchAll()
    // http://stackoverflow.com/questions/2770630/pdofetchall-vs-
    while ($row = $stmt->fetch(PDO::FETCH_ASSOC)){
        // extract row
        // this will make $row['name'] to
        // just $name only
        extract($row);

        // do what you need to do with $row here
    }
}
```

By using this site, you agree to our [Privacy Policy](https://www.codeofaninja.com/privacy-policy) <<https://www.codeofaninja.com/privacy-policy>>
and [Terms of Use](https://www.codeofaninja.com/terms-of-use) <<https://www.codeofaninja.com/terms-of-use>> .

OK

```

        "price" => $price,
        "category_id" => $category_id,
        "category_name" => $category_name
    );

    array_push($products_arr["records"], $product_item);
}

// include paging
$total_rows=$product->count();
$page_url="{$home_url}product/read_paging.php?";
$paging=$utilities->getPaging($page, $total_rows, $records_per_page);
$products_arr["paging"]=$paging;

// set response code - 200 OK
http_response_code(200);

// make it json format
echo json_encode($products_arr);
}

else{
    // set response code - 404 Not found
    http_response_code(404);

    // tell the user products does not exist
    echo json_encode(
        array("message" => "No products found.")
    );
}
?>

```

Create “core.php” file

This file holds our core configuration like the home URL and pagination variables.

- Open the config folder.

By using this site, you agree to our [Privacy Policy](https://www.codeofaninja.com/privacy-policy) <<https://www.codeofaninja.com/privacy-policy>>
and [Terms of Use](https://www.codeofaninja.com/terms-of-use) <<https://www.codeofaninja.com/terms-of-use>> .

OK

```
<?php
// show error reporting
ini_set('display_errors', 1);
error_reporting(E_ALL);

// home page url
$home_url="http://localhost/api/ < http://localhost/api/> ";

// page given in URL parameter, default page is one
$page = isset($_GET['page']) ? $_GET['page'] : 1;

// set number of records per page
$records_per_page = 5;

// calculate for the query LIMIT clause
$from_record_num = ($records_per_page * $page) - $records_per_page;
?>
```

Create "readPaging()" method

- Open objects folder.
- Open product.php file.
- Add the following method inside the product class.
- This method will return a list of records limited to what we set in \$records_per_page of the previous section.

By using this site, you agree to our [Privacy Policy](https://www.codeofaninja.com/privacy-policy) <<https://www.codeofaninja.com/privacy-policy>> .
and [Terms of Use](https://www.codeofaninja.com/terms-of-use) <<https://www.codeofaninja.com/terms-of-use>> .

OK

```

// read products with pagination
public function readPaging($from_record_num, $records_per_page){

    // select query
    $query = "SELECT
        c.name AS category_name, p.id, p.name, p.description
    FROM
        " . $this->table_name . " p
    LEFT JOIN
        categories c
        ON p.category_id = c.id
    ORDER BY p.created DESC
    LIMIT ?, ?";

    // prepare query statement
    $stmt = $this->conn->prepare( $query );

    // bind variable values
    $stmt->bindParam(1, $from_record_num, PDO::PARAM_INT);
    $stmt->bindParam(2, $records_per_page, PDO::PARAM_INT);

    // execute query
    $stmt->execute();

    // return values from database
    return $stmt;
}

```

Create “count()” method

Still in the product class (product.php file), add the following method. The total rows are needed to build the pagination array. It is included in the ‘paging’ computation.

By using this site, you agree to our [Privacy Policy](https://www.codeofaninja.com/privacy-policy) <<https://www.codeofaninja.com/privacy-policy>> .
and [Terms of Use](https://www.codeofaninja.com/terms-of-use) <<https://www.codeofaninja.com/terms-of-use>> .

OK

```
// used for paging products
public function count(){
    $query = "SELECT COUNT(*) as total_rows FROM " . $this->table;

    $stmt = $this->conn->prepare( $query );
    $stmt->execute();
    $row = $stmt->fetch(PDO::FETCH_ASSOC);

    return $row['total_rows'];
}
```

Get “paging” array

- Create shared folder.
- Open shared folder.
- Create `utilities.php` file.
- Open `utilities.php` file and place the following code.

By using this site, you agree to our [Privacy Policy](https://www.codeofaninja.com/privacy-policy) <<https://www.codeofaninja.com/privacy-policy>> and [Terms of Use](https://www.codeofaninja.com/terms-of-use) <<https://www.codeofaninja.com/terms-of-use>> .

OK

```

<?php
class Utilities{

    public function getPaging($page, $total_rows, $records_per_page)

        // paging array
        $paging_arr=array();

        // button for first page
        $paging_arr["first"] = $page>1 ? "{$page_url}page=1" : "";

        // count all products in the database to calculate total
        $total_pages = ceil($total_rows / $records_per_page);

        // range of links to show
        $range = 2;

        // display links to 'range of pages' around 'current page'
        $initial_num = $page - $range;
        $condition_limit_num = ($page + $range) + 1;

        $paging_arr['pages']=array();
        $page_count=0;

        for($x=$initial_num; $x<$condition_limit_num; $x++){
            // be sure '$x is greater than 0' AND 'less than or equal to $total_pages'
            if(($x > 0) && ($x <= $total_pages)){
                $paging_arr['pages'][$page_count]['page']=$x;
                $paging_arr['pages'][$page_count]['url']="{$page_url}page=$x";
                $paging_arr['pages'][$page_count]['current_page']=($x==$page);

                $page_count++;
            }
        }

        // button for last page
        $paging_arr["last"] = $page<$total_pages ? "{$page_url}page={$total_pages}" : "";

        // json format
        return $paging_arr;
}

```

By using this site, you agree to our [Privacy Policy](https://www.codeofaninja.com/privacy-policy) <<https://www.codeofaninja.com/privacy-policy>>
and [Terms of Use](https://www.codeofaninja.com/terms-of-use) <<https://www.codeofaninja.com/terms-of-use>> .

OK

Output

Open POSTMAN. Enter the following as the request URL.



Click the blue "Send" button.

- If there are products found, scroll down to see the paging node. It should look like this:

```

POST http://localhost/api/product/read_paging.php
Status: 200 OK Time: 18 ms Size: 1.33 KB
Send Save Download
Pretty Raw Preview JSON
200 OK
Standard response for successful HTTP requests.
The actual response will depend on the request method used. In a GET request, the response will contain an entity corresponding to the requested resource. In a POST request the response will contain an entity describing or containing the result of the action.

{
  "id": "28",
  "name": "Wallet Men",
  "description": "You can absolutely use this wallet men",
  "price": "799",
  "category_id": "6",
  "category_name": "Books"
},
{
  "paging": {
    "first": "",
    "pages": [
      {
        "page": 1,
        "url": "http://localhost/api/product/read_paging.php?page=1",
        "current_page": "yes"
      },
      {
        "page": 2,
        "url": "http://localhost/api/product/read_paging.php?page=2",
        "current_page": "no"
      },
      {
        "page": 3,
        "url": "http://localhost/api/product/read_paging.php?page=3",
        "current_page": "no"
      }
    ],
    "last": "http://localhost/api/product/read_paging.php?page=4"
  }
}

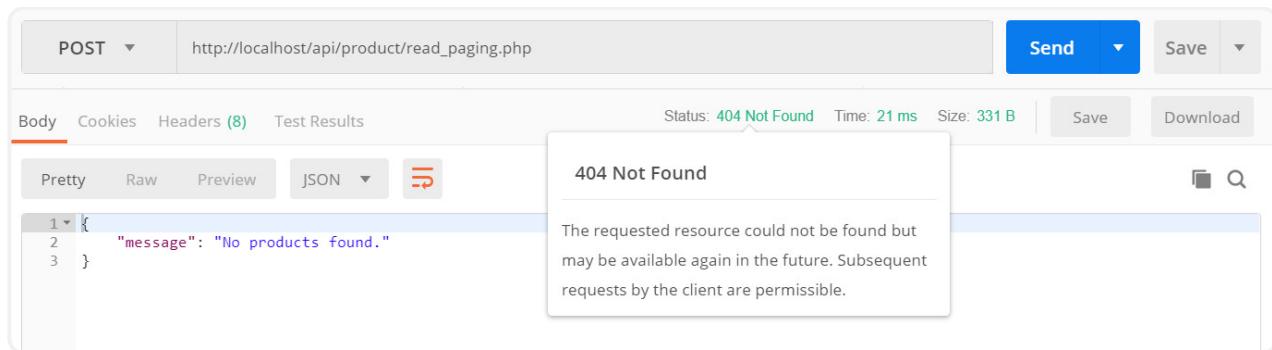
```

< <https://www.codeofaninja.com/wp-content/uploads/2017/02/read-paging-200.jpg> >

- If there are no products found, the output will look like this:

By using this site, you agree to our [Privacy Policy](https://www.codeofaninja.com/privacy-policy) < <https://www.codeofaninja.com/privacy-policy> >
and [Terms of Use](https://www.codeofaninja.com/terms-of-use) < <https://www.codeofaninja.com/terms-of-use> > .

OK



< <https://www.codeofaninja.com/wp-content/uploads/2017/02/read-paging-404.jpg>>

Read Categories

Create "category.php" file

- Open objects folder.
- Create new category.php file.
- Place the following code inside the category.php file.

By using this site, you agree to our [Privacy Policy](https://www.codeofaninja.com/privacy-policy) < https://www.codeofaninja.com/privacy-policy >
and [Terms of Use](https://www.codeofaninja.com/terms-of-use) < https://www.codeofaninja.com/terms-of-use > .

OK

```

<?php
class Category{

    // database connection and table name
    private $conn;
    private $table_name = "categories";

    // object properties
    public $id;
    public $name;
    public $description;
    public $created;

    public function __construct($db){
        $this->conn = $db;
    }

    // used by select drop-down list
    public function readAll(){
        //select all data
        $query = "SELECT
                    id, name, description
                FROM
                    " . $this->table_name . "
                ORDER BY
                    name";

        $stmt = $this->conn->prepare( $query );
        $stmt->execute();

        return $stmt;
    }
}
?>

```

Create "read.php" file

- Create new category folder.
- Open that folder and create new read.php file inside it.
- Open read.php file and place the following code.

By using this site, you agree to our [Privacy Policy](https://www.codeofaninja.com/privacy-policy) <<https://www.codeofaninja.com/privacy-policy>>
and [Terms of Use](https://www.codeofaninja.com/terms-of-use) <<https://www.codeofaninja.com/terms-of-use>> .

OK

```

<?php
// required header
header("Access-Control-Allow-Origin: *");
header("Content-Type: application/json; charset=UTF-8");

// include database and object files
include_once '../config/database.php';
include_once '../objects/category.php';

// instantiate database and category object
$database = new Database();
$db = $database->getConnection();

// initialize object
$category = new Category($db);

// query categories
$stmt = $category->read();
$num = $stmt->rowCount();

// check if more than 0 record found
if($num>0){

    // products array
    $categories_arr=array();
    $categories_arr["records"]=array();

    // retrieve our table contents
    // fetch() is faster than fetchAll()
    // http://stackoverflow.com/questions/2770630/pdofetchall-vs-
    while ($row = $stmt->fetch(PDO::FETCH_ASSOC)){
        // extract row
        // this will make $row[ 'name' ] to
        // just $name only
        extract($row);

        $category_item=array(
            "id" => $id,
            "name" => $name,
            "description" => html_entity_decode($description)
        );
}

```

By using this site, you agree to our [Privacy Policy](https://www.codeofaninja.com/privacy-policy) <<https://www.codeofaninja.com/privacy-policy>>
and [Terms of Use](https://www.codeofaninja.com/terms-of-use) <<https://www.codeofaninja.com/terms-of-use>> .

OK

```

    http_response_code(200);

    // show categories data in json format
    echo json_encode($categories_arr);
}

else{

    // set response code - 404 Not found
    http_response_code(404);

    // tell the user no categories found
    echo json_encode(
        array("message" => "No categories found.")
    );
}
?>

```

Add Category "read()" method

- Open objects folder.
- Open category.php file.
- The previous section's code will not work without the following code inside the category.php file.
- Add the following method inside the Category class.

```

// used by select drop-down list
public function read(){

    //select all data
    $query = "SELECT
              id, name, description
        FROM
              " . $this->table_name . "
        ORDER BY
              name";

    $stmt = $this->conn->prepare( $query );
    $stmt->execute();
}

```

By using this site, you agree to our [Privacy Policy](https://www.codeofaninja.com/privacy-policy) <<https://www.codeofaninja.com/privacy-policy>>
and [Terms of Use](https://www.codeofaninja.com/terms-of-use) <<https://www.codeofaninja.com/terms-of-use>> .

OK

Output

Open POSTMAN. Enter the following as the request URL.



Click the blue "Send" button.

- If there are categories found, it should look like this:

```

POST http://localhost/api/category/read.php
Status: 200 OK Time: 32 ms Size: 820 B
Send Save Download
Body Cookies Headers (8) Test Results
Pretty Raw Preview JSON
200 OK
Standard response for successful HTTP requests.
The actual response will depend on the request
method used. In a GET request, the response will
contain an entity corresponding to the requested
resource. In a POST request the response will
contain an entity describing or containing the
result of the action.

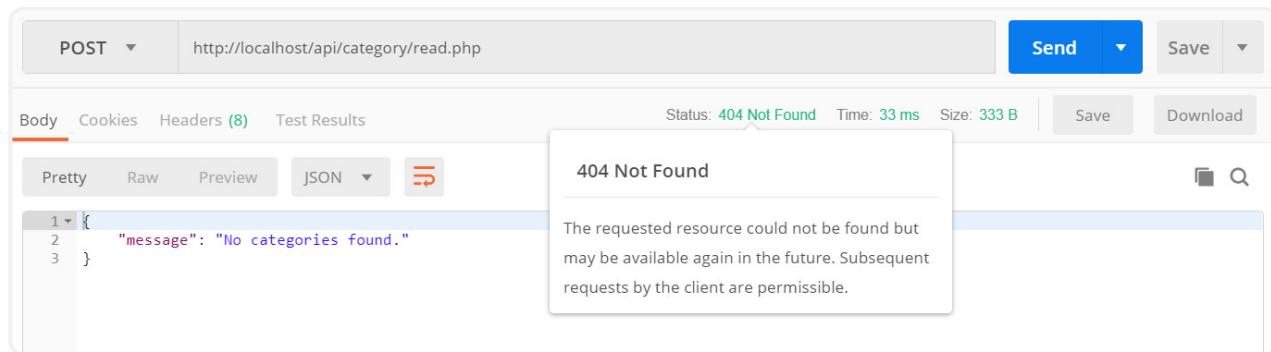
records: [
    {
        "id": "6",
        "name": "Books",
        "description": "Kindle books, audio books"
    },
    {
        "id": "2",
        "name": "Electronics",
        "description": "Gadgets, drones and more."
    },
    {
        "id": "1",
        "name": "Fashion",
        "description": "Category for anything related to fashion."
    },
    {
        "id": "20",
        "name": "Home",
        "description": "Category for home and garden products!"
    },
    {
        "id": "3",
        "name": "Motors",
        "description": "Motor sports and more"
    },
    {
        "id": "5",
        "name": "Sports",
        "description": "Category for sports and fitness products"
    }
]
  
```

< https://www.codeofaninja.com/wp-content/uploads/2017/02/read-categories-200.jpg>

- If there are no categories found, the output will look like this:

By using this site, you agree to our [Privacy Policy](https://www.codeofaninja.com/privacy-policy) < https://www.codeofaninja.com/privacy-policy>
and [Terms of Use](https://www.codeofaninja.com/terms-of-use) < https://www.codeofaninja.com/terms-of-use> .

OK



< <https://www.codeofaninja.com/wp-content/uploads/2017/02/read-categories-404.jpg> >

Download Source Codes

We highly recommend for you to follow and study our well-detailed, step-by-step tutorial above first. Nothing beats experience when it comes to learning.

But we believe you will learn faster if you'll see the final source code as well. We consider it as your additional guide.

Imagine the value or skill upgrade it can bring you. The additional income you can get from your work, projects or business. The precious time you save. Isn't that what you want?

Download LEVEL 1 source code

By using this site, you agree to our [Privacy Policy](https://www.codeofaninja.com/privacy-policy) < <https://www.codeofaninja.com/privacy-policy> > and [Terms of Use](https://www.codeofaninja.com/terms-of-use) < <https://www.codeofaninja.com/terms-of-use> > .

OK

FEATURES	LEVEL 1
Create product	YES
Read products	YES
Read one product	YES
Update product	YES
Delete product	YES
Search products	YES
Paginate products	YES
Read categories	YES
FREE email support for 3 months	YES
Source code updates via email	YES
\$30 – Download now < https://buy.paddle.com/product/663261>	

Download LEVEL 2 source code

Before downloading the LEVEL 2 source code, I highly recommend learning the LEVEL 1 source code first. Use our tutorial above. LEVEL 2 might be a bit challenging to understand if you don't fully understand LEVEL 1.

By using this site, you agree to our [Privacy Policy](https://www.codeofaninja.com/privacy-policy) < <https://www.codeofaninja.com/privacy-policy>>
and [Terms of Use](https://www.codeofaninja.com/terms-of-use) < <https://www.codeofaninja.com/terms-of-use>> .

OK

FEATURES	LEVEL 2
All features of LEVEL 1 source code	YES
Delete selected product	YES
Export product CSV	YES
Read products by category	YES
Search products with pagination	YES
Create category	YES
Read categories	YES
Read one category	YES
Update category	YES
Delete category	YES
Search categories	YES
Paginate categories	YES
Delete selected categories	YES
Export categories CSV	YES
Search categories with pagination	YES
FREE email support for 6 months	YES
Source code updates via email	YES

By using this site, you agree to our [Privacy Policy](https://www.codeofaninja.com/privacy-policy) < <https://www.codeofaninja.com/privacy-policy>>
 and [Terms of Use](https://www.codeofaninja.com/terms-of-use) < <https://www.codeofaninja.com/terms-of-use>> .

OK

You have two options:

- a. Take your skills to the next level by learning how to do REST API authentication (with user interface) on our [**REST API Authentication Example in PHP – JWT Tutorial**](https://www.codeofaninja.com/2018/09/rest-api-authentication-example-php-jwt-tutorial.html) <<https://www.codeofaninja.com/2018/09/rest-api-authentication-example-php-jwt-tutorial.html>> .
- b. The tutorial above focuses on the API side, without any user interface. We will learn how to create, read, update and delete database records (with user interface) on our [**AJAX CRUD Tutorial**](https://www.codeofaninja.com/2015/06/php-crud-with-ajax-and-oop.html) <<https://www.codeofaninja.com/2015/06/php-crud-with-ajax-and-oop.html>> .

If you want to receive new and updated high-quality tutorials, [**please subscribe**](#) <<https://www.codeofaninja.com/subscribe>> for free.

Related Tutorials

We listed all our high quality full-stack web development tutorials here: [**Click here**](#) <<https://codeofaninja.com/web-development-tutorials/>> .

Notes

Found An Issue?

If you found a problem with this code, please send us an email. Before you send an email, please read our [**our code of conduct**](#) <<https://www.codeofaninja.com/code-of-conduct>> . Our team's email address is [**support@codeofaninja.com**](mailto:support@codeofaninja.com)

Please be descriptive about your issue. Please provide the error messages,

By using this site, you agree to our [**Privacy Policy**](#) <<https://www.codeofaninja.com/privacy-policy>> and [**Terms of Use**](#) <<https://www.codeofaninja.com/terms-of-use>> .

OK

Receive valuable web development tutorials to your email. Subscribe now for FREE!

Name

Email

Subscribe Now

Thank You!

Please note that this post is not yet in its final form. We are going to update this post so it will be perfect in the future.

If you have a friend or know someone who needs this PHP REST API Tutorial, please share this page to them! I know you will help them a lot by doing it. Thanks!



By Mike Dalisay

I'm Mike Dalisay, a pro web developer since 2010. I love web development. Improving our tutorials and source codes makes me happy. Do you want to suggest an edit to our tutorial? Got something to ask about our source codes? You may use our comments section below or

By using this site, you agree to our **Privacy Policy** <<https://www.codeofaninja.com/privacy-policy>> and **Terms of Use** <<https://www.codeofaninja.com/terms-of-use>> .

OK

185 comments



Mirosław Całka

August 24, 2021 at 5:34 am

<https://uploads.disquscdn.com/images/0bd04a0468e05aa050da2e5a531b79a0710357f4d2f7e9a1f303044d2baf6d05.png> <

<https://uploads.disquscdn.com/images/0bd04a0468e05aa050da2e5a531b79a0710357f4d2f7e9a1f303044d2baf6d05.png> > Hi

While testing point 6.3 in Postman I see warnings "Trying to access array offset on value of type bool". Is this correct?



Mike Dalisay < https://www.codeofaninja.com >

August 31, 2021 at 1:41 am

Hi @mirosawcaka, I cannot reproduce the issue. Would you show us your code in objects/product.php file in line around 110?



0000magda0000

February 10, 2021 at 9:36 pm

I am new to PHP and didn't know that I had to initialize the vars in a constructor function. here is my initial question:

By using this site, you agree to our [Privacy Policy](https://www.codeofaninja.com/privacy-policy) < <https://www.codeofaninja.com/privacy-policy> > and [Terms of Use](https://www.codeofaninja.com/terms-of-use) < <https://www.codeofaninja.com/terms-of-use> > .

OK

" HI :) I built a rest API with this tutorial, I enjoyed it a lot. Thank you! Now I am trying to deploy the API on Heroku. But I am having trouble with the database.php configurations. Bc this code examples depends on the class database which is used to set up the connection to the Database and PHP doesn't want me to put my environment variables to access the Heroku DB within the class. Any hints on how I could solve that?



YoMama

February 1, 2021 at 7:26 pm

This is too complex.

I want to look at how to implement endpoints with PHP versus Express or Rails or Django.

Most of this is about database and retrieving data. It's a lot of stuff that skips an overview of essential.

I give it an F.



Mike Dalisay < <https://www.codeofaninja.com> >

November 21, 2018 at 1:00 pm

Hi @iqbalprashiasuhardi, would you show us a screenshot of your read.php code?

By using this site, you agree to our [Privacy Policy](https://www.codeofaninja.com/privacy-policy) < <https://www.codeofaninja.com/privacy-policy> >
and [Terms of Use](https://www.codeofaninja.com/terms-of-use) < <https://www.codeofaninja.com/terms-of-use> > .

OK

Hi @disqus_OT9R0KMI6s, did you try to run your code on localhost?



Mike Dalisay < https://www.codeofaninja.com >

November 21, 2018 at 11:38 am

Hi @disqus_WgiJsBfCzg, it will work with PHP 5.5+, the code is not tested with postgresql.



Mike Dalisay < https://www.codeofaninja.com >

November 21, 2018 at 10:36 am

Hi @disqus_LGtoUa2quM, would you send a screenshot of your update.php code around line 26-32? Make sure your /objects/product.php was properly set up as well.



Mike Dalisay < https://www.codeofaninja.com >

November 21, 2018 at 10:33 am

Hi @vincentjonathan, you're welcome! Yes, I believe it is safe from SQL Injection because we are using prepared statements. About authentication, I suggest you study our tutorial here: <https://www.codeofaninja.com/2018/09/rest-api-authentication-example-php-jwt-tutorial.html> <
<https://www.codeofaninja.com/2018/09/rest-api-authentication-example-php-jwt-tutorial.html>>

By using this site, you agree to our [Privacy Policy](https://www.codeofaninja.com/privacy-policy) < <https://www.codeofaninja.com/privacy-policy> >
and [Terms of Use](https://www.codeofaninja.com/terms-of-use) < <https://www.codeofaninja.com/terms-of-use> > .

OK

▼ ▼ **October 31, 2018 at 6:06 am**

Hi, everybody,

i'm trying this code but I don't know why I can't create a product. it return me 503 error code. Any idea?



Mike Dalisay < https://www.codeofaninja.com >

November 21, 2018 at 10:06 am

Hi @disqus_33q8IBEIJf, we are unable to replicate the issue. 503 means service is unavailable, make sure your server is running correctly.



Mike Dalisay < https://www.codeofaninja.com >

October 30, 2018 at 6:12 pm

I suggest you test both the options.



Mike Dalisay < https://www.codeofaninja.com >

October 30, 2018 at 6:11 pm

Hi @disqus_UHxBXCcyty, sorry, I did not understand your question. What exactly are you trying to accomplish?

By using this site, you agree to our [Privacy Policy](https://www.codeofaninja.com/privacy-policy) < https://www.codeofaninja.com/privacy-policy >
and [Terms of Use](https://www.codeofaninja.com/terms-of-use) < https://www.codeofaninja.com/terms-of-use > .

OK

Hi @disqus_qYLV79Vi6b, it looks like your product.php file is not in the correct location. Please take a look at the section 2.0 (file structure) above.



Mike Dalisay < https://www.codeofaninja.com >

October 16, 2018 at 8:13 am

Thanks for bringing this to my attention. That part is now fixed. I was testing the code for a negative result and I forgot to bring it back.



Mike Dalisay < https://www.codeofaninja.com >

October 16, 2018 at 8:11 am

Great! Thank you for sharing your solution. I'm sure this is helpful for other people who face the same issue.



Mike Dalisay < https://www.codeofaninja.com >

October 12, 2018 at 7:57 am

Hi @eugeniesoulange, I'm unable to replicate the issue. Would you show us your read.php code? Please wrap your code in pre and code tag, see an example here:

<https://help.disqus.com/commenting/what-html-tags-are-allowed-within-comments> < https://help.disqus.com/commenting/what-html-tags-are-allowed-within-comments >

By using this site, you agree to our [Privacy Policy](https://www.codeofaninja.com/privacy-policy) < https://www.codeofaninja.com/privacy-policy >
and [Terms of Use](https://www.codeofaninja.com/terms-of-use) < https://www.codeofaninja.com/terms-of-use > .

OK

Hi @khalid, thanks for the suggestion. We do not have one yet, but we will make it in the future. It will be posted on androidcode.ninja website.



DANN CHRISTIAN

September 19, 2018 at 10:28 am

Is there anyway to put token for authentication for this REST API tutorial



Mike Dalisay < https://www.codeofaninja.com >

October 12, 2018 at 7:22 am

Hi @dannchristian, yes, our tutorial here might help you:

https://www.codeofaninja.com/2018/09/rest-api-authentication-example-php-jwt-tutorial.html < https://www.codeofaninja.com/2018/09/rest-api-authentication-example-php-jwt-tutorial.html >



Mike Dalisay < https://www.codeofaninja.com >

September 11, 2018 at 7:09 am

I suggest you learn more about POST and GET here

https://stackoverflow.com/questions/715335/get-vs-post-in-ajax < https://stackoverflow.com/questions/715335/get-vs-post-in-ajax >

By using this site, you agree to our **Privacy Policy** < <https://www.codeofaninja.com/privacy-policy> >
and **Terms of Use** < <https://www.codeofaninja.com/terms-of-use> > .

OK

Thanks for your suggestion.

What's wrong in the following?

[<](http://localhost/php_mysql/php_api/product/searchdate.php?start_date=2014-06-01&end_date=2014-06-30)

[>](http://localhost/php_mysql/php_api/product/searchdate.php?start_date=2014-06-01&end_date=2014-06-30)

Please correct it.



Mike Dalisay < <https://www.codeofaninja.com>>

September 14, 2018 at 4:58 pm

I can't correct it because I cannot debug it for you. If you're using the URL with values, you must use the GET request to get it processed by PHP.



Mike Dalisay < <https://www.codeofaninja.com>>

September 10, 2018 at 8:52 am

Hi @dannchristian make sure the read() method exists inside your product class. Make sure you did not misplace the closing bracket of your product class.



Mike Dalisay < <https://www.codeofaninja.com>>

September 10, 2018 at 8:50 am

By using this site, you agree to our [Privacy Policy](https://www.codeofaninja.com/privacy-policy) < <https://www.codeofaninja.com/privacy-policy>>
and [Terms of Use](https://www.codeofaninja.com/terms-of-use) < <https://www.codeofaninja.com/terms-of-use>> .

OK



DANN CHRISTIAN

September 10, 2018 at 9:08 am

```
conn = $db;  
}  
  
}  
  
// read products  
function read() {  
  
// select all query  
$query = "SELECT  
c.name as category_name, p.id, p.name, p.description, p.price, p.category_id,  
p.created  
FROM  
" . $this->table_name . " p  
LEFT JOIN  
categories c  
ON p.category_id = c.id  
ORDER BY  
p.created DESC";  
  
// prepare query statement  
$stmt = $this->conn->prepare($query);  
  
// execute query  
$stmt->execute();
```

~~return \$stmt;~~

By using this site, you agree to our [Privacy Policy](https://www.codeofaninja.com/privacy-policy) <<https://www.codeofaninja.com/privacy-policy>>
and [Terms of Use](https://www.codeofaninja.com/terms-of-use) <<https://www.codeofaninja.com/terms-of-use>> .

OK

```
// search products
function search($keywords){

    // select all query
    $query = "SELECT
        c.name as category_name, p.id, p.name, p.description, p.price, p.category_id,
        p.created
    FROM
        " . $this->table_name . " p
    LEFT JOIN
        categories c
    ON p.category_id = c.id
    WHERE
        p.name LIKE ? OR p.description LIKE ? OR c.name LIKE ?
    ORDER BY
        p.created DESC";

    // prepare query statement
    $stmt = $this->conn->prepare($query);

    // sanitize
    $keywords=htmlspecialchars(strip_tags($keywords));
    $keywords = "%{$keywords}%";

    // bind
    $stmt->bindParam(1, $keywords);
    $stmt->bindParam(2, $keywords);
    $stmt->bindParam(3, $keywords);
```

By using this site, you agree to our [Privacy Policy](https://www.codeofaninja.com/privacy-policy) <<https://www.codeofaninja.com/privacy-policy>>
and [Terms of Use](https://www.codeofaninja.com/terms-of-use) <<https://www.codeofaninja.com/terms-of-use>> .

OK

```
return $stmt;  
}  
?>
```

the only script in my product.php



Mike Dalisay < https://www.codeofaninja.com >

September 11, 2018 at 7:00 am

Hi @dannchristian, the code looks wrong, please follow our tutorial carefully.



DANN CHRISTIAN

September 11, 2018 at 8:31 am

i Mike

How this become wrong it seems i copy paste it to the editor



Mike Dalisay < https://www.codeofaninja.com >

September 14, 2018 at 4:55 pm

Hi @dannchristian, please do not just copy and paste the codes from tutorial above. Please understand how it works so you can make it work.

By using this site, you agree to our [Privacy Policy < https://www.codeofaninja.com/privacy-policy >](https://www.codeofaninja.com/privacy-policy)
[and Terms of Use < https://www.codeofaninja.com/terms-of-use > .](https://www.codeofaninja.com/terms-of-use)

OK

[<](https://uploads.disquscdn.com/images/43de4f600bd88abdb77455f7bd6eef10c42099fdb92af4249a77ef27a7dd9072.png)
[>](https://uploads.disquscdn.com/images/43de4f600bd88abdb77455f7bd6eef10c42099fdb92af4249a77ef27a7dd9072.png)



[Mike Dalisay < https://www.codeofaninja.com >](https://www.codeofaninja.com)

[September 14, 2018 at 4:56 pm](#)

This code looks correct.



[Mike Dalisay < https://www.codeofaninja.com >](https://www.codeofaninja.com)

[September 10, 2018 at 8:34 am](#)

Hi @maziksolutionsmzk, you can try our Angular tutorial linked on the post above.



[Mike Dalisay < https://www.codeofaninja.com >](https://www.codeofaninja.com)

[September 10, 2018 at 8:14 am](#)

You can pass the value the same way we pass the data in the "create" operation on our tutorial above.



[Mike Dalisay < https://www.codeofaninja.com >](https://www.codeofaninja.com)

[August 25, 2018 at 6:39 pm](#)

By using this site, you agree to our [Privacy Policy < https://www.codeofaninja.com/privacy-policy >](https://www.codeofaninja.com/privacy-policy)
and [Terms of Use < https://www.codeofaninja.com/terms-of-use >](https://www.codeofaninja.com/terms-of-use).

OK

```
$start_date="2014-06-01";  
$end_date"2014-06-30";
```

I'm not sure why this is not working on your end. Which PHP and MySQL version are you using? Are you debugging locally?

Yes, the isset method is needed. The code on your searchdate.php is in comments, please remove the double slash.



Mike Dalisay < https://www.codeofaninja.com >

August 14, 2018 at 1:27 pm

Hi @disqus_TR7OR7SoPY, if it works on localhost, I think there's a problem with your database connection. Would you try to show an error message using this inside the "else" statement.

```
print_r($this->conn->errorInfo());
```



Mike Dalisay < https://www.codeofaninja.com >

August 14, 2018 at 1:16 pm

Would you paste your whole query? Also, make sure you got records with the category ID you specified.

By using this site, you agree to our [Privacy Policy < https://www.codeofaninja.com/privacy-policy >](https://www.codeofaninja.com/privacy-policy) and [Terms of Use < https://www.codeofaninja.com/terms-of-use >](https://www.codeofaninja.com/terms-of-use).

OK

how would i update read.php in the API to show only a certain category? e.g. only Electronics



Mike Dalisay < https://www.codeofaninja.com >

July 31, 2018 at 9:15 am

Hi @disqus_jOu97V5UXn, you need to add a "WHERE" clause on your query. It looks like: ...WHERE category_id=3 something like that.



Mike Dalisay < https://www.codeofaninja.com >

July 30, 2018 at 12:16 pm

Hi @disqus_rjV2B2Ktib. Unfortunately, our tutorial above is using PDO only. We do not have tutorial for MySQLi.



Rani Somu

July 16, 2018 at 9:58 pm

Hi,

This article is clearly explained each and every step. It is very useful. Thank you for this article.

I need your guidance to achieve the following requirements.

1. "SELECT * FROM" . \$this->table_name . " WHERE created between ? and ?;"
2. "SELECT * FROM" . \$this->table_name . " WHERE created >=? and created<=?;"

By using this site, you agree to our [Privacy Policy](https://www.codeofaninja.com/privacy-policy) < https://www.codeofaninja.com/privacy-policy >
and [Terms of Use](https://www.codeofaninja.com/terms-of-use) < https://www.codeofaninja.com/terms-of-use > .

OK



Mike Dalisay < https://www.codeofaninja.com >

July 30, 2018 at 12:14 pm

You're welcome @ranisomu! What exactly is the guidance you are seeking? Those queries look correct to me.

If you are asking how to bind values on those queries, I suggest you study the "search" feature on our tutorial here:

<https://www.codeofaninja.com/2014/06/php-object-oriented-crud-example-oop.html> < https://www.codeofaninja.com/2014/06/php-object-oriented-crud-example-oop.html >



Rani Somu

July 31, 2018 at 2:52 pm

Thank you for your reply Mr. Mike.

Yes, I need your guidance to bind the values for the above queries. I already tried with the format of search query, but it didn't return nothing. That's why I'm seeking your help. Can you please guide me.

[https://uploads.disquscdn.com/images/f9fa8c215aeaa3484ad724e34b4f86560756e3a6933244e3f6f7f58d87d6a816.jpg < https://uploads.disquscdn.com/images/f9fa8c215aeaa3484ad724e34b4f86560756e3a6933244e3f6f7f58d87d6a816.jpg >](https://uploads.disquscdn.com/images/f9fa8c215aeaa3484ad724e34b4f86560756e3a6933244e3f6f7f58d87d6a816.jpg)

[https://uploads.disquscdn.com/images/bd49ffed7c4bf68ed02a7c08e345af5beb727a0f4597f7675e7ef9eab1388221.jpg < https://uploads.disquscdn.com/images/bd49ffed7c4bf68ed02a7c08e345af5beb727a0f4597f7675e7ef9eab1388221.jpg >](https://uploads.disquscdn.com/images/bd49ffed7c4bf68ed02a7c08e345af5beb727a0f4597f7675e7ef9eab1388221.jpg)

By using this site, you agree to our [Privacy Policy](https://www.codeofaninja.com/privacy-policy) < https://www.codeofaninja.com/privacy-policy >
and [Terms of Use](https://www.codeofaninja.com/terms-of-use) < https://www.codeofaninja.com/terms-of-use > .

OK

<https://uploads.disquscdn.com/images/78276562d271d0bed90e1efd427071daa1bad0ff6738dda6e096bc2a24f2d52c.jpg> <

<https://uploads.disquscdn.com/images/78276562d271d0bed90e1efd427071daa1bad0ff6738dda6e096bc2a24f2d52c.jpg>>

<https://uploads.disquscdn.com/images/4f31f482105b2348d3816ac0fc99f9d79249ec1a35df5ad7070a90609bcab4bd.jpg> <

<https://uploads.disquscdn.com/images/4f31f482105b2348d3816ac0fc99f9d79249ec1a35df5ad7070a90609bcab4bd.jpg>>

<https://uploads.disquscdn.com/images/16e8b4e46f572fc2c9d019ce9b18b48d0269e9fdf1a1718f75ec22f799fe36b8.jpg> <

<https://uploads.disquscdn.com/images/16e8b4e46f572fc2c9d019ce9b18b48d0269e9fdf1a1718f75ec22f799fe36b8.jpg>>

<https://uploads.disquscdn.com/images/c29d1e0fe79d59f8ea660abb6e0b05549a719f6e9ab6aaeb2c2c72e917673cca.jpg> <

<https://uploads.disquscdn.com/images/c29d1e0fe79d59f8ea660abb6e0b05549a719f6e9ab6aaeb2c2c72e917673cca.jpg>>

<https://uploads.disquscdn.com/images/b3ccee05bc4bce7954ffd1d7e61f56c81f889b027deffb4004e1fe11072c3f8f.jpg> <

<https://uploads.disquscdn.com/images/b3ccee05bc4bce7954ffd1d7e61f56c81f889b027deffb4004e1fe11072c3f8f.jpg>>

<https://uploads.disquscdn.com/images/771bf2c823ae85c43d4e40593bad2134582a3ffbc4e9f36aa61377b3279f17a7.jpg> <

<https://uploads.disquscdn.com/images/771bf2c823ae85c43d4e40593bad2134582a3ffbc4e9f36aa61377b3279f17a7.jpg>>

Here are the screenshots for your reference.

By using this site, you agree to our Privacy Policy <<https://www.codeofaninja.com/privacy-policy>> and Terms of Use <<https://www.codeofaninja.com/terms-of-use>> .

OK



July 31, 2018 at 5:29 pm

Please see the code in section 10.7 here

<https://www.codeofaninja.com/2014/06/php-object-oriented-crud-example-oop.html> < <https://www.codeofaninja.com/2014/06/php-object-oriented-crud-example-oop.html>>



Rani Somu

July 31, 2018 at 7:00 pm

That section deals with LIKE and bind parameter surrounded by %{ }% as such as in mysql. It's working fine for LIKE.

In my case, I need to do it for BETWEEN and >=, <=. The step which is used for LIKE doesn't work for my requirement.

I'm not getting an idea, how to pass the bind parameter for BETWEEN. So only I had attached the screenshots.

Kindly please refer it and guide me to achieve this.



Mike Dalisay < https://www.codeofaninja.com >

August 14, 2018 at 1:19 pm

@ranisomu, sorry, I misunderstood your question. If you need to use the between/and clause, it should look like this:

By using this site, you agree to our [Privacy Policy](https://www.codeofaninja.com/privacy-policy) < <https://www.codeofaninja.com/privacy-policy>>
and [Terms of Use](https://www.codeofaninja.com/terms-of-use) < <https://www.codeofaninja.com/terms-of-use>> .

OK



Mike Dalisay < https://www.codeofaninja.com >

July 15, 2018 at 12:39 am

Hi @pradeep_shekhwat, how are you testing this API? Would you show your code in create.php file?



Mike Dalisay < https://www.codeofaninja.com >

July 15, 2018 at 12:33 am

Hi @Husam, would you show use the code of your update.php so we can see what was wrong? How are you testing this?



Mike Dalisay < https://www.codeofaninja.com >

July 15, 2018 at 12:25 am

Hi @disqus_ky7Wa1JuzZ, thanks for the kind words and suggestion! Postman is very useful, thanks for the tip about it and testing the API with Android as well.



Mike Dalisay < https://www.codeofaninja.com >

July 15, 2018 at 12:24 am

Hi @marceloclopes, thanks for the suggestion! We will definitely release a new tutorial with JSON Web Tokens soon. Please subscribe:

[< https://www.codeofaninja.com/subscribe >](https://www.codeofaninja.com/subscribe)

By using this site, you agree to our [Privacy Policy < https://www.codeofaninja.com/privacy-policy >](https://www.codeofaninja.com/privacy-policy) and [Terms of Use < https://www.codeofaninja.com/terms-of-use >](https://www.codeofaninja.com/terms-of-use).

OK



Mike Dalisay < https://www.codeofaninja.com >

July 15, 2018 at 12:20 am

Thank you for the detailed reply @tobiasparent! We'll definitely improve the tutorial above with your suggestions in mind. I'm looking forward to the link of your work. I'm glad you appreciate our resource even though it is not perfect. Thank you!



Mike Dalisay < https://www.codeofaninja.com >

July 15, 2018 at 12:01 am

Hi @amit, you can always use the green download button on section 12.0 above.



Mike Dalisay < https://www.codeofaninja.com >

July 14, 2018 at 11:57 pm

You're welcome @savarisagayadevans! Please subscribe for more awesomeness, see: [< https://www.codeofaninja.com/subscribe >](https://www.codeofaninja.com/subscribe)



Mike Dalisay < https://www.codeofaninja.com >

July 14, 2018 at 11:54 pm

Hi @dave_aeez, first, I'm not forcing anyone to use MySQL. A person like you

By using this site, you agree to our [< https://www.codeofaninja.com/privacy-policy >](https://www.codeofaninja.com/privacy-policy) and [< https://www.codeofaninja.com/terms-of-use >](https://www.codeofaninja.com/terms-of-use).

OK

I'm not forcing you to use it. In fact, I updated the tutorial with InnoDB as well.

I don't believe my question makes no sense. Some users have had to rely on MyISAM for its full text search capabilities, and in some cases, better performance in applications that didn't require transactional capabilities.



Mike Dalisay < https://www.codeofaninja.com >

June 7, 2018 at 5:08 pm

Hi @dave_geez, thanks for your comment. Would you tell us what's wrong with using MyISAM in the example above?



Mike Dalisay < https://www.codeofaninja.com >

June 7, 2018 at 4:58 pm

Hi @gabiluva, why do you need an ID to be returned from the create method?



Gabi Luva

June 7, 2018 at 6:39 pm

Normally I would return the hole json object (or an id) in a spa app or mobile app you need that for any other operations following, show details or link some other object to this new one or delete or I don't know whatever etc.

By using this site, you agree to our [Privacy Policy < https://www.codeofaninja.com/privacy-policy >](https://www.codeofaninja.com/privacy-policy) and [Terms of Use < https://www.codeofaninja.com/terms-of-use >](https://www.codeofaninja.com/terms-of-use).

OK

Unfortunately, we do not have this feature on our tutorial above. But you can play around with PDO's lastInsertId(), see

<https://stackoverflow.com/questions/10680943/pdo-get-the-last-id-inserted>
[< https://stackoverflow.com/questions/10680943/pdo-get-the-last-id-](https://stackoverflow.com/questions/10680943/pdo-get-the-last-id-inserted)
[inserted>](https://stackoverflow.com/questions/10680943/pdo-get-the-last-id-inserted)



keith

May 18, 2018 at 4:05 am

This doesn't work for a react fetch call – I've followed the code exactly, I'm receiving a Json response back when I visit in a browser but (I think) something is wrong with the header where the react code:

```
fetch('fetch('http://localhost/api/product/read_one.php?id=60')  
.then((response) => response.json())  
.then(json => this.setState({series:json}))
```

won't retrieve the data back

why is that?

thanks

K



Mike Dalisay < https://www.codeofaninja.com >

May 28, 2018 at 7:08 am

By using this site, you agree to our [Privacy Policy < https://www.codeofaninja.com/privacy-policy >](https://www.codeofaninja.com/privacy-policy)
and [Terms of Use < https://www.codeofaninja.com/terms-of-use >](https://www.codeofaninja.com/terms-of-use).

OK

[<
\[>\]\(https://www.codeofaninja.com/2016/07/react-crud-tutorial.html\)](https://www.codeofaninja.com/2016/07/react-crud-tutorial.html)



Vivek Gupta

May 15, 2018 at 2:40 am

not able to get output of search() function

i m getting this

Fatal error: Uncaught Error: Call to undefined method Product::search() in
E:xampp\htdocs\root\folder\api\product\search.php:21

line no. 21// query products

```
$stmt = $product->search($keywords);
```

```
$num = $stmt->rowCount();
```

plz help



[< https://www.codeofaninja.com >](https://www.codeofaninja.com)

May 28, 2018 at 7:02 am

Hi @disqus_VMblFhhH2J, please make sure you added the search() method inside your product class. See the search method in section 9.2 above.



[< https://www.codeofaninja.com >](https://www.codeofaninja.com)

May 10, 2018 at 3:06 pm

By using this site, you agree to our [< https://www.codeofaninja.com/privacy-policy >](https://www.codeofaninja.com/privacy-policy)
[< https://www.codeofaninja.com/terms-of-use >](https://www.codeofaninja.com/terms-of-use).

OK



Mike Dalisay < https://www.codeofaninja.com >

May 10, 2018 at 2:37 pm

Hi @meindertjorna, I'm glad it works now and thanks for sharing your solution! But do you mean you placed the read() function inside the Product class?



Mike Dalisay < https://www.codeofaninja.com >

May 10, 2018 at 2:26 pm

You're welcome @nbatioco! You're right, you can use the technology to connect lot of applications.



Mike Dalisay < https://www.codeofaninja.com >

May 10, 2018 at 2:10 pm

Hi @androidcoban, unfortunately, that's not part of our tutorial above. We will release a related login tutorial soon, please subscribe so we can let you know. Use this form: [< https://www.codeofaninja.com/subscribe >](https://www.codeofaninja.com/subscribe)



Mike Dalisay < https://www.codeofaninja.com >

May 10, 2018 at 1:38 pm

Hi @tobiasparent, thanks for the kind words and sorry for the late reply. It's been a

By using this site, you agree to our [Privacy Policy < https://www.codeofaninja.com/privacy-policy >](https://www.codeofaninja.com/privacy-policy) and [Terms of Use < https://www.codeofaninja.com/terms-of-use >](https://www.codeofaninja.com/terms-of-use).

OK

Let me know if I understood your concern correctly. Do you mean you need the code to run with different type of data sources and not rely with PDO? You have a point, we can do that with PHP because it can connect even to an Oracle database.

Unfortunately, I am not familiar with that situation and my knowledge is currently limited to using PDO. It works with all my projects and my clients are happy with it.

It sounds like you're a very knowledgeable person and I'd love to learn more from you! Can you point us to better references or resources so that we can improve our tutorial above?

valex



March 25, 2018 at 7:12 pm

Am trying to connect to the mysql using PDO but am getting could not find driver error. what do i do?

Mike Dalisay < <https://www.codeofaninja.com>>



March 26, 2018 at 10:38 am

Hi @Valex, it looks like your development environment is not properly set up. You can try to use XAMPP, here's a tutorial

[<https://www.codeofaninja.com/2013/06/how-to-run-a-php-script.html>](https://www.codeofaninja.com/2013/06/how-to-run-a-php-script.html)

If you're working on a remote server, you should contact your web hosting

By using this site, you agree to our **Privacy Policy** <<https://www.codeofaninja.com/privacy-policy>> and **Terms of Use** <<https://www.codeofaninja.com/terms-of-use>> .

OK



Tarandeep Singh

March 14, 2018 at 7:21 pm

Thanks Man!!

It helped me lot in learning and making api in php for my project. Thanks again



Mike Dalisay < https://www.codeofaninja.com >

March 16, 2018 at 10:30 am

You're welcome @disqus_McgLMZMyJN! Please share our site to your friends and subscribe here **https://www.codeofaninja.com/subscribe < https://www.codeofaninja.com/subscribe >**



Sudhir < https://www.facebook.com/comedymood >

March 8, 2018 at 6:49 pm

how to remove .php from url in core php?



Mike Dalisay < https://www.codeofaninja.com >

March 16, 2018 at 10:30 am

Hi @sudhir600, if you want the clean URLs, you can make use of .htaccess, here's a helpful tutorial **https://www.codeofaninja.com/2013/04/mod-rewrite-url.html < https://www.codeofaninja.com/2013/04/mod-rewrite-url.html >**

By using this site, you agree to our **Privacy Policy < https://www.codeofaninja.com/privacy-policy >** and **Terms of Use < https://www.codeofaninja.com/terms-of-use >**.

OK



March 6, 2018 at 12:42 am

Hi. I'm using this tutorial to help me build my own REST api. But within the search part of my api, I need to be able to search by two parameters. Is this an easy fix in this code? I'm really struggling to figure out how to implement it.



Mike Dalisay < <https://www.codeofaninja.com>>

March 16, 2018 at 10:21 am

Hi @disqus_fpCGRZwrkb, if you need a feature where you can pass two parameters, you can format the search like this:

search.php?s=searchkey&s2=searchkey2

and get the value of s2 like this

```
$keywords2=isset($_GET["s2"]) ? $_GET["s2"] : "";
```



maneesha padamata

February 28, 2018 at 1:49 pm

thanks alot :)

By using this site, you agree to our **Privacy Policy** <<https://www.codeofaninja.com/privacy-policy>>
and **Terms of Use** <<https://www.codeofaninja.com/terms-of-use>> .

OK

You're welcome @maneeshapadamata! Please share our site to your friends and subscribe for free here [< https://www.codeofaninja.com/subscribe >](https://www.codeofaninja.com/subscribe)



So Fabulous

February 23, 2018 at 10:45 pm

Me and my classmate are making this api on different laptops. His and mine weren't able to communicate for some reason. While testing the create method, the only thing that's being inputted in my database are brackets. Like so

[< https://uploads.disquscdn.com/images/79d4ca780abe3a158ef684be084c6951ca57be16d2a0779e5b437495d62476c3.png >](https://uploads.disquscdn.com/images/79d4ca780abe3a158ef684be084c6951ca57be16d2a0779e5b437495d62476c3.png)

I did mess with the code a lot, since I was using it for something else, but I wonder if you know why this occurs?



Mike Dalisay < https://www.codeofaninja.com >

February 28, 2018 at 12:11 pm

Hi @disqus_7hJqGh3eKk, thank you for using our tutorial! Unfortunately, there's no way for me to debug your customizations. I highly recommend following the exact tutorial above first. After that, you can do your own customizations on the code. It will be easier since you understand the basics.

By using this site, you agree to our [Privacy Policy < https://www.codeofaninja.com/privacy-policy >](https://www.codeofaninja.com/privacy-policy) and [Terms of Use < https://www.codeofaninja.com/terms-of-use >](https://www.codeofaninja.com/terms-of-use).

OK

I am facing this problem

Fatal error: Call to undefined method Product::read() in
E:\xampp\htdocs\API\select.php



Mike Dalisay < https://www.codeofaninja.com >

February 23, 2018 at 10:42 am

@waleedrehan, please read my response on your first comment.



waleed rehan

February 21, 2018 at 3:19 pm

Hello

I am facing this error.

Fatal error: Using \$this when not in object context in E:\xampp\htdocs\API\view.php
on line 6

Kindly give me solution for this.



Mike Dalisay < https://www.codeofaninja.com >

February 23, 2018 at 10:41 am

Hi @waleedrehan, we don't have view.php on our tutorial above. Please follow the

By using this site, you agree to our **Privacy Policy** < <https://www.codeofaninja.com/privacy-policy> >
and **Terms of Use** < <https://www.codeofaninja.com/terms-of-use> > .

OK



Rather not Say

February 3, 2018 at 2:13 pm

Thanks.



Mike Dalisay < https://www.codeofaninja.com >

February 10, 2018 at 3:51 pm

You're welcome @Rather not Say!



Gary Harris

January 31, 2018 at 10:22 pm

This is a great guide for developing a REST API! Would it be possible for you guys to put together a simple guide on how to secure the API with OAuth? Or, could you point me to a resource that describes securing an API with OAuth from beginning to end? Thanks!



Mike Dalisay < https://www.codeofaninja.com >

February 10, 2018 at 4:02 pm

Thanks for the kind words about our tutorial @Gary Harris! We'll post our REST API authentication tutorial soon. You can get a lot ideas how to build it here:

<https://stackoverflow.com/questions/7999295/rest-api-authentication> <

By using this site, you agree to our **Privacy Policy** < <https://www.codeofaninja.com/privacy-policy> >
and **Terms of Use** < <https://www.codeofaninja.com/terms-of-use> > .

OK



Rookie Dev

January 24, 2018 at 6:11 am

Curious as to why you end each file with the php closing tag? Isn't this generally considered poor practice? <http://php.net/manual/en/language.basic-syntax.phptags.php> < <https://php.net/manual/en/language.basic-syntax.phptags.php>>



Mike Dalisay < https://www.codeofaninja.com >

January 24, 2018 at 12:07 pm

Hi @Rookie Dev, thanks for visiting our site and sharing your thoughts! The closing PHP tag is not a "bad practice". The link you sent says it is something that is "preferable" if a file is a pure PHP code.

I've been coding PHP for many years and the closing PHP tag was never a problem. It is just a personal preference. If you think an accidental whitespace is a big deal for your app, then I suggest you don't use the closing PHP tag.



Rookie Dev

January 26, 2018 at 11:08 pm

Thanks for clarifying! I want to make sure I follow best practices as I learn so that I learn the "correct" way.

By using this site, you agree to our [Privacy Policy](https://www.codeofaninja.com/privacy-policy) < <https://www.codeofaninja.com/privacy-policy> >
and [Terms of Use](https://www.codeofaninja.com/terms-of-use) < <https://www.codeofaninja.com/terms-of-use> > .

OK

You're welcome @disqus_ymPF68U0LP! Thanks for your positive feedback! Please share our site to your friends and subscribe here

[<https://www.codeofaninja.com/subscribe>](https://www.codeofaninja.com/subscribe)



Meltito Vagallon

December 29, 2017 at 3:00 pm

hi I got an error how to i solve this **Fatal error:** Uncaught Error: Class 'Database' not found in C:xampphtdocsAPI_practiceproductread.php:9



[>](https://www.codeofaninja.com)

December 30, 2017 at 10:38 pm

Hi @meltitovagallon, make sure you followed section 3.0 properly.



HCB

December 10, 2017 at 11:09 pm

First, I'm very excited about starting this, because it looks comprehensive enough without getting too far "into the weeds" for someone who needs to work a day job (stuck in a legacy env.) and learn new technologies. Question about the file structure though: Are all the directories to be placed off the docroot, or are the subsequent directories off the 'api' directory? Like this:

By using this site, you agree to our **Privacy Policy** <[>](https://www.codeofaninja.com/privacy-policy) and **Terms of Use** <[>](https://www.codeofaninja.com/terms-of-use).

OK

└── database.php – file used for connecting to the database.

└── objects/

└── product.php – contains properties and methods for “product” database queries.

└── category.php – contains properties and methods for “category” database queries.

└── product/

└── create.php – file that will accept posted product data to be saved to database.

└── delete.php – file that will accept a product ID to delete a database record.

└── read.php – file that will output JSON data based from “products” database records.

└── read_paging.php – file that will output “products” JSON data with pagination.

└── read_one.php – file that will accept product ID to read a record from the database.

└── update.php – file that will accept a product ID to update a database record.

└── search.php – file that will accept keywords parameter to search “products” database.

└── category/

└── read.php – file that will output JSON data based from “categories”

By using this site, you agree to our [Privacy Policy](https://www.codeofaninja.com/privacy-policy) < <https://www.codeofaninja.com/privacy-policy> >
and [Terms of Use](https://www.codeofaninja.com/terms-of-use) < <https://www.codeofaninja.com/terms-of-use> > .

OK

```
|── shared/  
|   └── utilities.php – file that will return pagination array.
```



Mike Dalisay < <https://www.codeofaninja.com>>

December 14, 2017 at 8:11 pm

Hi @HCB, thanks for the kind words! Yes, all the files are inside the "api" directory. You can place your "api" directory in your document root.



Mohamed Mohamed

December 5, 2017 at 10:50 am

Hi. May I ask, how do you automatically delete a product as you won't need to refresh the page?



Mike Dalisay < <https://www.codeofaninja.com>>

December 8, 2017 at 8:28 pm

Hi @disqus_tGEtJxEHOZ, the JavaScript application sends a post request with the "product ID" to the "delete" API. Follow our example here:

**<https://www.codeofaninja.com/2015/06/php-crud-with-ajax-and-oop.html> <
<https://www.codeofaninja.com/2015/06/php-crud-with-ajax-and-oop.html>>**

By using this site, you agree to our [Privacy Policy](https://www.codeofaninja.com/privacy-policy) < <https://www.codeofaninja.com/privacy-policy>>
and [Terms of Use](https://www.codeofaninja.com/terms-of-use) < <https://www.codeofaninja.com/terms-of-use>> .

OK

Hi @zelkaeddy, thanks for the catch! Sorry for the late reply, I just read your comment. It went to the spam folder. Those type has been fixed now, thanks to you again!



Mike Dalisay < https://www.codeofaninja.com >

December 4, 2017 at 8:26 pm

Hi @disqus_jd9M9R6P27, thanks for sharing your fix, I appreciate it!



Mike Dalisay < https://www.codeofaninja.com >

December 4, 2017 at 8:25 pm

Hi @disqus_jd9M9R6P27, thanks for the kind words about our tutorial! I'm sorry I'm not familiar with sqlsrv. You can check if the \$num variable value is correct or greater than 0. Make sure you sample data is sufficient as well.



Adriano Giovannini

November 27, 2017 at 3:00 am

Hello,

first and foremost great tutorial (and great job) but i have a little question:

In your screen i see the json output formatted, and your browser is chrome. Do you get a formatting that way? I have run both on linux and windows with chrome

.....
By using this site, you agree to our [Privacy Policy < https://www.codeofaninja.com/privacy-policy >](https://www.codeofaninja.com/privacy-policy)
[and Terms of Use < https://www.codeofaninja.com/terms-of-use > .](https://www.codeofaninja.com/terms-of-use)

OK

```
{"id": "1", "name": "Fashion", "description": "Category for anything related to fashion."},  
 {"id": "3", "name": "Motor", "description": "Motor sports and more"},  
 {"id": "5", "name": "Movies", "description": "Movie products"},  
 {"id": "6", "name": "Sports", "description": "Drop into new winter gear."}]}]
```



Mike Dalisay < <https://www.codeofaninja.com>>

December 4, 2017 at 8:07 pm

Hi @androidmegasources, thanks for the kind words about our tutorial! As I've said above, I'm using a Chrome extension called JSONView to make the JSON data readable in the browser. See

[<https://chrome.google.com/webstore/detail/jsonview/chklaanhfefbnpoihckbnefhakgolnmc>](https://chrome.google.com/webstore/detail/jsonview/chklaanhfefbnpoihckbnefhakgolnmc)

[<https://chrome.google.com/webstore/detail/jsonview/chklaanhfefbnpoihckbnefhakgolnmc>](https://chrome.google.com/webstore/detail/jsonview/chklaanhfefbnpoihckbnefhakgolnmc)



Masresha Tsegaye A.K.A Mati

November 24, 2017 at 3:56 pm

Hey Mike,

Thanks for the tutorial. U just saved me from being dependable from the Back-End guys.

I have a question. I want to read only one product by sending the id and get a response "the product full info" like we send requests to delete a product. I don't want to pass the value on the URL.

By using this site, you agree to our **Privacy Policy** <<https://www.codeofaninja.com/privacy-policy>> and **Terms of Use** <<https://www.codeofaninja.com/terms-of-use>> .

OK

<https://uploads.disquscdn.com/images/8e947c978957cf13ca11585f0fc8b689bdd8811dcdad0c126b1c8cee2846ddd9.png>



Mike Dalisay < https://www.codeofaninja.com >

November 26, 2017 at 7:08 am

Hi @masreshatsegayeakamati , you can use your JavaScript application to do a POST request and then your PHP code must accept a POST request as well. You can access the "id" or any parameter using this method.



Masresha Tsegaye A.K.A Mati

November 24, 2017 at 3:38 pm

What a great tutorial. This is what i've been looking for days. Thanks!!!!



Mike Dalisay < https://www.codeofaninja.com >

November 26, 2017 at 7:02 am

You're welcome @masreshatsegayeakamati! I'm glad to save you some time. Thanks for the kind words about our tutorial!



Kalana Wijekoon

November 21, 2017 at 5:03 pm

By using this site, you agree to our [Privacy Policy < https://www.codeofaninja.com/privacy-policy >](https://www.codeofaninja.com/privacy-policy) and [Terms of Use < https://www.codeofaninja.com/terms-of-use >](https://www.codeofaninja.com/terms-of-use).

OK



Mike Dalisay < https://www.codeofaninja.com >

November 24, 2017 at 8:14 am

You're welcome @kalanawijekoon! Please share our site to your friends, or subscribe [< https://www.codeofaninja.com/subscribe < https://www.codeofaninja.com/subscribe >](https://www.codeofaninja.com/subscribe) Thank you!



Besart Emine Cana

October 29, 2017 at 11:16 pm

Great tutorial! Thanks

Have some questions. how can I paginate search results?

Regards

Besart



Mike Dalisay < https://www.codeofaninja.com >

November 2, 2017 at 4:09 am

You're welcome @besarteminecana! You can use the same code pattern we used in section "10.0 Paginate Products". Or you can download the LEVEL 2 source code above.



Besart Emine Cana

By using this site, you agree to our [< https://www.codeofaninja.com/privacy-policy >](https://www.codeofaninja.com/privacy-policy) and [< https://www.codeofaninja.com/terms-of-use >](https://www.codeofaninja.com/terms-of-use).

OK

I made it already, thanks!



paulo

October 29, 2017 at 6:13 am

Utilities class and getPaging method can be static, so you don't have to instanciate it



Mike Dalisay < https://www.codeofaninja.com >

November 2, 2017 at 4:02 am

Hi @paulo, it can be. But it's a good practice to make it reusable for other objects.



Amin R

October 10, 2017 at 12:24 am

Hi @@nijazhai thanks alot for this applicable tutorial.

I have a problem in receiving data In the product/create.php file, "php://input" is empty when I send http post request.

I checked \$data = json_decode(file_get_contents("php://input")) and also \$_POST through these line of codes :

```
var_dump($data);
```

```
var_dump($_POST);
```

By using this site, you agree to our [Privacy Policy < https://www.codeofaninja.com/privacy-policy >](https://www.codeofaninja.com/privacy-policy)
[and Terms of Use < https://www.codeofaninja.com/terms-of-use > .](https://www.codeofaninja.com/terms-of-use)

OK

controller php file (create.php) shows nothing data received.
can you help me to debug this problem ?



Mike Dalisay < <https://www.codeofaninja.com>>

October 12, 2017 at 2:48 pm

Hi @amin_r, you're welcome!

About your question, did you use our JavaScript code with your create.php file?
Our script above was only tested with our JavaScript code.

Try to follow one of our JavaScript tutorials, see

[<https://www.codeofaninja.com/javascript-programming-tutorials>](https://www.codeofaninja.com/javascript-programming-tutorials)



Max Ufer

December 13, 2017 at 3:42 am

I have the same problem, but the code is the same. Can you help me?



Mike Dalisay < <https://www.codeofaninja.com>>

December 14, 2017 at 8:09 pm

Hi @maxufer, did you try to follow one of our JavaScript tutorials?

By using this site, you agree to our **Privacy Policy** <<https://www.codeofaninja.com/privacy-policy>>
and **Terms of Use** <<https://www.codeofaninja.com/terms-of-use>> .

OK

UPDATE: We are now using POSTMAN (<https://www.getpostman.com/> < <https://www.getpostman.com/>>) to test the output of our code above. You don't have to use any JavaScript code to test it.

We provided instructions and screenshots on each "Output" section so that the testing will be easier to follow.



atomicnacation

January 24, 2018 at 6:30 pm

Hi, I use Postman for the requests. I was experiencing the same issue: product created, but empty.

But, if you configure headers to "Content-Type : application/json" it works. I introduced json raw data for test. Here the example:

[<](https://uploads.disquscdn.com/images/5fb53c5473d3a8240880bb03ed576a4d338f022357c094d0df9c7d4f96a39301.png)
[>](https://uploads.disquscdn.com/images/5fb53c5473d3a8240880bb03ed576a4d338f022357c094d0df9c7d4f96a39301.png)
[<](https://uploads.disquscdn.com/images/4ee674bb8f7db323a9940ad20b14072134912ab98b00a9ccafc134cfea5ece6.png)
[>](https://uploads.disquscdn.com/images/4ee674bb8f7db323a9940ad20b14072134912ab98b00a9ccafc134cfea5ece6.png)
[<](https://uploads.disquscdn.com/images/84991f56ffe19b73b721993071546339b5c2ce3e73633b5ad10fbb0fceabcb.png)
[>](https://uploads.disquscdn.com/images/84991f56ffe19b73b721993071546339b5c2ce3e73633b5ad10fbb0fceabcb.png)

By using this site, you agree to our [Privacy Policy](#) < <https://www.codeofaninja.com/privacy-policy>>
and [Terms of Use](#) < <https://www.codeofaninja.com/terms-of-use>> .

OK



Mike Dalisay < https://www.codeofaninja.com >

January 25, 2018 at 10:09 am

Hi @atomicnation, thanks for sharing this tip! This one is very useful for Postman users. :)



atomicnation

January 25, 2018 at 8:45 pm

Thanks to you for this brilliant tutorial. By the way, I wonder if you know some place where there is any explanation of the headers and when & how they should be used, referred to REST services. Thanks!



Tarandeep Singh

March 14, 2018 at 7:22 pm

use

\$data = (object) \$_POST;

instead of

\$data = json_decode(file_get_contents("php://input"));



Mike Dalisay < https://www.codeofaninja.com >

March 16, 2018 at 10:19 am

By using this site, you agree to our [Privacy Policy < https://www.codeofaninja.com/privacy-policy >](https://www.codeofaninja.com/privacy-policy)
[and Terms of Use < https://www.codeofaninja.com/terms-of-use > .](https://www.codeofaninja.com/terms-of-use)

OK



Mike Dalisay < <https://www.codeofaninja.com>>

October 16, 2018 at 8:15 am

UPDATE: We are now using POSTMAN to test the output of our code above. You don't have to use any JavaScript code to test it.



Gary Thompson

September 16, 2017 at 12:41 am

Based on this work I've created endpoints in codeigniter for products – just as a learning exercise. **[<https://bitbucket.org/sushiguru/api>](https://bitbucket.org/sushiguru/api)**



Mike Dalisay < <https://www.codeofaninja.com>>

October 3, 2017 at 6:08 pm

Thanks for sharing your work @sushiguru!



Techbyte Solutions

August 26, 2017 at 1:04 pm

header("Access-Control-Allow-Origin: *"); is not suitable. It allows anyone to allow access to our resources.

You must provide some access token or OAuth for secure connection. bdw Your

By using this site, you agree to our **Privacy Policy < <https://www.codeofaninja.com/privacy-policy>>** and **Terms of Use < <https://www.codeofaninja.com/terms-of-use>>**.

OK



Mike Dalisay < https://www.codeofaninja.com >

August 27, 2017 at 5:11 pm

Thanks for sharing your thoughts @techbytesolutions! But this tutorial is not for private APIs. As said on the tutorial, this is not yet in its final form as well. You can always extend it to your project's requirements.



Canh Phong

December 7, 2017 at 12:41 pm

thanks



Mike Dalisay < https://www.codeofaninja.com >

December 8, 2017 at 8:31 pm

You're welcome @canhphong! Please share our site to your friends!



Guido Leijten

August 22, 2017 at 8:41 pm

Hi! It appears you're missing an else on 8.2 Product delete() method after the if statement under execute query.

By using this site, you agree to our [Privacy Policy < https://www.codeofaninja.com/privacy-policy >](https://www.codeofaninja.com/privacy-policy) and [Terms of Use < https://www.codeofaninja.com/terms-of-use >](https://www.codeofaninja.com/terms-of-use).

OK

Hi @guidoleijten, thanks for your comment! We don't miss an 'else' on section 8.2, the code will work as is. If '\$stmt->execute()' is true, the method will return true. It won't execute the 'return false' under it.



Guido Leijten

August 23, 2017 at 4:11 pm

Fair enough! It just looked like it as all the other methods do have it set up with an if/else. I must say that this is an excellent tutorial, I am following a PHP course and this week we had to find out how to build our own API, I searched the net for information on it and I got a vague idea, but after following your tutorial it's all so clear and it's amazing how simple it can be! I've recommended it to everyone here and my teacher will link to it on his blog as well, because not only is it good for people who want to learn how to create a RESTful API, but it's also excellent to get a better grasp on OOP as I hear from a lot of people here that they don't really know when to use classes.

One other thing, I am using the Chrome extension Advanced REST Client and when trying to PUT/POST I had to utf8_encode the input before json_decode, it appears that others in the comment section did not experience this issue. Easily fixable, but it made me wonder why my code didn't work without it.

Anyway, again – great tutorial! Definitely something I'll keep recommending when I hear someone struggling trying to create their own API.



Mike Dalisay < <https://www.codeofaninja.com> >

By using this site, you agree to our [Privacy Policy](https://www.codeofaninja.com/privacy-policy) < <https://www.codeofaninja.com/privacy-policy> >
and [Terms of Use](https://www.codeofaninja.com/terms-of-use) < <https://www.codeofaninja.com/terms-of-use> > .

OK

You're awesome @guidoleijten! Thank you for your kind words and recommending our site to your friends. Thanks for the tip about utf8_encode as well!



Mike Dalisay < https://www.codeofaninja.com >

August 5, 2017 at 4:17 pm

Hi @disqus_8D8jRDxac2, unfortunately we don't have a tutorial for this yet.



Roberto

July 6, 2017 at 12:46 pm

Hi Mike.

I had made the first level but when I run the read.php I get this: {"message": "No products found."}.

I have the tables well done. I had made the change in the table name "products" to "productos" and I don't receive any warning. Thanks in advance for your comments



Mike Dalisay < https://www.codeofaninja.com >

July 11, 2017 at 7:18 am

Did you change your table name in product.php?



Robert < http://www.recuperodati299euro.it/>

By using this site, you agree to our [Privacy Policy](https://www.codeofaninja.com/privacy-policy) < https://www.codeofaninja.com/privacy-policy >
and [Terms of Use](https://www.codeofaninja.com/terms-of-use) < https://www.codeofaninja.com/terms-of-use > .

OK

Hi Mike

one kind "issue" with all of this related with the AJAX CRUD js php tutorial that uses this API

All of this cannot be used in production right? :-)

Since e.g. the API is without authentication and all of the js files are easy readable so they will unveil the API path and commands

Do you think to add some security? Or this tutorial is intended for the sole purpose of teaching methods? Anyway the security is part of the learn stage ;-)



Mike Dalisay < https://www.codeofaninja.com >

June 28, 2017 at 5:47 pm

Hi @Corsari , this tutorial is intended for the sole purpose of teaching methods. You are right about the security being part of the learn stage. But that's another topic and deserves another post. We'll try to release that kind of tutorial soon.



Mike Dalisay < https://www.codeofaninja.com >

June 26, 2017 at 6:51 pm

Hi @Corsari , your comment above is the one I'm talking about. I don't know why but I found it in the spam section of Disqus, so I thought you deleted it.

Anyway, I added section 1.1 to 1.4 on the post. Those are based on your

By using this site, you agree to our [Privacy Policy < https://www.codeofaninja.com/privacy-policy >](https://www.codeofaninja.com/privacy-policy)
[and Terms of Use < https://www.codeofaninja.com/terms-of-use >](https://www.codeofaninja.com/terms-of-use).

OK



Mike Dalisay < <https://www.codeofaninja.com>>

June 26, 2017 at 4:47 am

Hi @Corsari , thanks for visiting our site!

To answer your question, we used LEFT JOIN to pull the "category name", without it, we can only pull the "category ID" from the products table and so we have to do another query for the "category name".

In other words, using the LEFT JOIN is more efficient because we don't have to do multiple queries just to get the category name.

By the way, I love your first comment here. Thanks you for the kind words! I wish you can republish it because I want to answer all your questions from there as well.



Robert < <http://www.recuperodati299euro.it/>>

June 26, 2017 at 2:39 pm

Hi Mike

As you may have noticed, yesterday I was following your tutorials (REST API + following one, the "Pure AJAX CRUD")

So I have written some comments around the website

Can you point me to the comment you mean?

Thank you

By using this site, you agree to our **Privacy Policy** <<https://www.codeofaninja.com/privacy-policy>> and **Terms of Use** <<https://www.codeofaninja.com/terms-of-use>> .

OK

small fixes to avoid confusion in newbies like me :-)

10.1 Create "read_paging.php" file, no indications on where to create it (we know, in the products folder, but is worth to indicate)

10.2 states "create config folder", but it has been already created for the database.php file

10.3 Create "readPaging()" method, no indications on where to create it (we know, in the product.php file, but is worth to indicate)

10.4 Create "count()" method, same as 10.3

11.3 Add Category "read()" method – Open "objects" folder. Create new "category.php" file. But this file already exists



Mike Dalisay < https://www.codeofaninja.com >

June 26, 2017 at 5:04 am

Hi @Corsari , thanks for your detailed corrections! I've made the changes on the post above.



Robert < http://www.recuperodati299euro.it />

June 27, 2017 at 1:03 am

Good ! I'm glad you have fixed them.



Robert < http://www.recuperodati299euro.it />

June 25, 2017 at 9:47 pm

By using this site, you agree to our [Privacy Policy](https://www.codeofaninja.com/privacy-policy) < https://www.codeofaninja.com/privacy-policy >
and [Terms of Use](https://www.codeofaninja.com/terms-of-use) < https://www.codeofaninja.com/terms-of-use > .

OK

[<](https://uploads.disquscdn.com/images/099775e74f8ca92fb2197d63b3d5581c34ac029893ea291be5a4ed47db03960a.png)
[>](https://uploads.disquscdn.com/images/099775e74f8ca92fb2197d63b3d5581c34ac029893ea291be5a4ed47db03960a.png>)



[Mike Dalisay < https://www.codeofaninja.com >](https://www.codeofaninja.com)

[June 26, 2017 at 4:49 am](#)

Thanks for sharing @Corsari! This will help Firefox users. On the screenshot, I saw that you bookmarked our site, thank you for that as well!



[Robert < http://www.recuperodati299euro.it/ >](http://www.recuperodati299euro.it/)

[June 26, 2017 at 2:31 pm](#)

:-)

I'd be surprised that someone that would land on Code Of A Ninja website would not bookmark it...



Petarr Nonkovic

[June 29, 2017 at 2:44 am](#)

Or just pass second argument to json_encode(\$products_arr, JSON_PRETTY_PRINT) , no extension needed. Or whatever you like, CHEERS ALL!
Great tutorial by the way!!

By using this site, you agree to our [Privacy Policy](#) < <https://www.codeofaninja.com/privacy-policy> >
and [Terms of Use](#) < <https://www.codeofaninja.com/terms-of-use> > .

OK



June 29, 2017 at 6:45 pm

I didn't know this, thanks for this tip @petarrnonkovic ! Glad you liked our tutorial!



hanifeoglu

March 27, 2018 at 8:36 pm

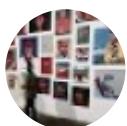
@petarrnonkovic thank for trick i m more like just your words mistake
“json_enconde” -> “json_encode” result here : echo json_encode(\$products_arr,
JSON_PRETTY_PRINT);



Mike Dalisay < https://www.codeofaninja.com >

March 28, 2018 at 11:29 am

Thank you guys for the tips! This helps other people using this site. :)



Nii Quartey

June 12, 2017 at 6:11 pm

how do i access this api via url? I am new to using API's and REST especially. Or how do i access it on other platforms. I need help please. It's a very good tutorial and i liked the step by step process

By using this site, you agree to our [Privacy Policy < https://www.codeofaninja.com/privacy-policy >](https://www.codeofaninja.com/privacy-policy)
[and Terms of Use < https://www.codeofaninja.com/terms-of-use > .](https://www.codeofaninja.com/terms-of-use)

OK

Hi @niiquartey , if you want your API to be accessible in other platforms, you should upload it on your web server and access it via your domain or temporary URL.



Anshul Mahipal

June 12, 2017 at 8:49 am

Helle Mike, great tutorial

But I stuck on single think that I want to post data using postman it create a record with empty field so please help me



Mike Dalisay < https://www.codeofaninja.com >

June 14, 2017 at 9:49 am

Hi @anshulmahipal , please read my notes after section 5.2 code.



Anshul Mahipal

June 14, 2017 at 10:23 am

after 5.2 you put these lines

Please complete one of our JavaScript programming tutorials. This same concept applies to our "update" and "delete" code.

But I am Stuck on what javascript I Use because i don't use HTML forms

By using this site, you agree to our **Privacy Policy** < <https://www.codeofaninja.com/privacy-policy> > and **Terms of Use** < <https://www.codeofaninja.com/terms-of-use> > .

OK

If you can't learn one of our JavaScript tutorials, our tutorial above is not right for you. You have to learn somewhere else.



Anshul Mahipal

June 8, 2017 at 3:42 pm

Great Tutorial Mike. But there is issue I just want to use that code with postman talking about create() it create the product but with empty field will you please tell me reason.



Mike Dalisay < <https://www.codeofaninja.com>>

June 14, 2017 at 9:51 am

You can only use our JavaScript code to test this API. Please read my notes after section 5.2 code above.



welly boang

June 5, 2017 at 3:12 pm

Hi Mike, great tutorial...

how execute POST method ?
such as create () method.. thanks

By using this site, you agree to our **Privacy Policy < <https://www.codeofaninja.com/privacy-policy>>**
and Terms of Use < <https://www.codeofaninja.com/terms-of-use>> .

OK

Hi @wellyboang , as I said at the end of section 5.2, I highly recommend completing this whole tutorial first. But if you want to test the code above, you have to use our JavaScript code.

The reason is our JavaScript code is designed to work with this REST API. Please complete one of our JavaScript programming tutorials. See

[<https://www.codeofaninja.com/javascript-programming-tutorials>](https://www.codeofaninja.com/javascript-programming-tutorials)

This same concept applies to our “update” and “delete” code.



Laurence Hudson Montgomery

May 11, 2017 at 7:35 am

Great tutorial, love the use of ‘output’ checks. I stumbled a bit at 6.3 because of hard to read details in the provided graphic not found in the text instructions:

“If you develop on localhost and will run the read.php file using this URL:

[<http://localhost/api/product/read_one.php>](http://localhost/api/product/read_one.php)

[>](http://localhost/api/product/read_one.php)

You will see an output like this...”

You might want to indicate in your text that an id value has to be passed :

“If you develop on localhost and will run the read.php file using this URL:

[<http://localhost/api/product/read_one.php?id=60>](http://localhost/api/product/read_one.php?id=60)

[>](http://localhost/api/product/read_one.php?id=60)

You will see an output like this...”

By using this site, you agree to our **Privacy Policy** <<https://www.codeofaninja.com/privacy-policy>>
and **Terms of Use** <<https://www.codeofaninja.com/terms-of-use>> .

OK



May 12, 2017 at 6:26 pm

Hi @laurencehudsonmontgomery , thanks for letting me know about the graphic being hard to read at your end.

I just replaced it with a bigger image. We added more description as well. Let us know if it is more readable now.



Nonsono Statoio

April 29, 2017 at 10:37 pm

Hi Mike, very interesting tut!

My console give me a problem here:

Parse error: syntax error, unexpected 'public' (T_PUBLIC) in
C:xampphtdocsAngular-restconfigcore.php on line **19**

maybe is because the public function must be declared inside a class?



Mike Dalisay < https://www.codeofaninja.com >

May 3, 2017 at 12:19 pm

Hi @nonsono_statoio , yes, all functions must be inside a class.



Nikhil Khandave

By using this site, you agree to our **Privacy Policy** < <https://www.codeofaninja.com/privacy-policy> >
and **Terms of Use** < <https://www.codeofaninja.com/terms-of-use> > .

OK

Hello Mike, great tutorial, learned so much from your tutorials, i hav just started PHP and angular js, i am getting error from this line

```
$product->id = isset($_GET['id']) ? $_GET['id'] : die("no record found");
```

it displays "no record found",

could there be something wrong, becuase when i hard code the product->id = 1,

it shows the result. and

one more thing how \$_GET['id'] works ?

Thanks in advace



Mike Dalisay < <https://www.codeofaninja.com>>

April 26, 2017 at 5:43 pm

Hi @nikhilkhandave , glad to hear you're learning so much from our tutorials!

About your question, \$_GET['id'] will work if you pass "id" as parameter in your URL. For example:

http://localhost/product/read_one.php?id=10 <

http://localhost/product/read_one.php?id=10



Nikhil Khandave

April 26, 2017 at 8:06 pm

By using this site, you agree to our **Privacy Policy** < <https://www.codeofaninja.com/privacy-policy>>
and **Terms of Use** < <https://www.codeofaninja.com/terms-of-use>> .

OK



Masresha Tsegaye A.K.A Mati

November 24, 2017 at 3:44 pm

Hey Mike, What if i want to read only one product only by passing this "id":"the id value". Like when we delete a product we pass the id and its value. I dont want to pass the question mark sign and write it on my URL.

<https://uploads.disquscdn.com/images/8e947c978957cf13ca11585f0fc8b689bdd8811dcdad0c126b1c8cee2846ddd9.png> <

<https://uploads.disquscdn.com/images/8e947c978957cf13ca11585f0fc8b689bdd8811dcdad0c126b1c8cee2846ddd9.png> >



Mike Dalisay < https://www.codeofaninja.com >

November 26, 2017 at 7:05 am

Hi @masreshatsegayeakamati, you can use your JavaScript application to do a POST request and then your PHP code must accept a POST request as well. You can access the "id" or any parameter this way.



Mauro Rojas

April 20, 2017 at 7:10 pm

thanks, excelent tutorial!



Mike Dalisay < https://www.codeofaninja.com >

By using this site, you agree to our [Privacy Policy](https://www.codeofaninja.com/privacy-policy) < https://www.codeofaninja.com/privacy-policy >
and [Terms of Use](https://www.codeofaninja.com/terms-of-use) < https://www.codeofaninja.com/terms-of-use > .

OK

Please share our site to one of your friends or subscribe for more excellent tutorials, use: [<https://www.codeofaninja.com/subscribe>](https://www.codeofaninja.com/subscribe)



Zuda Madhara

March 24, 2017 at 2:52 am

Hi Mike, Great stuff. However, output for read_one.php populates only the id as shown below:

```
{  
id: "2",  
name: null,  
description: null,  
price: null,  
category_id: null,  
category_name: null  
}
```

Could there be something wrong?

====Ignore me ... it was due to bad data in my database!!=====



Mike Dalisay <https://www.codeofaninja.com>

March 24, 2017 at 9:06 am

Hello @zudamadhara , I see, thanks for your comment and sharing your solution!

By using this site, you agree to our [Privacy Policy <https://www.codeofaninja.com/privacy-policy>](https://www.codeofaninja.com/privacy-policy)
and [Terms of Use <https://www.codeofaninja.com/terms-of-use>](https://www.codeofaninja.com/terms-of-use) .

OK



March 23, 2017 at 6:58 pm

Hi, great tutorial. I am getting Fatal error: Call to undefined method Product::count() in read_paging.php – any ideas? Thanks



Mike Dalisay < https://www.codeofaninja.com >

March 24, 2017 at 9:05 am

Hello @vitozgonik , this error can be solved by adding the "count()" method in section 10.4 above.



Vito Zgonik

March 24, 2017 at 6:01 pm

Thanks so much, I must have missed this out by accident :)



Ashish Maharjan

March 23, 2017 at 5:36 pm

Hi, Mike. Thank you so much a very easy and well formatted tutorial. I think there's a function "count()" missing in "Product" class which is calling from "read_paging.php" class. Could you please add that function? Thanks.

By using this site, you agree to our [Privacy Policy < https://www.codeofaninja.com/privacy-policy >](https://www.codeofaninja.com/privacy-policy)
[and Terms of Use < https://www.codeofaninja.com/terms-of-use > .](https://www.codeofaninja.com/terms-of-use)

OK

Hi @ashish_maharjan , thanks for the kind words!

I just added the "count()" method in section 10.4 above.



cebuano

March 22, 2017 at 9:26 pm

Hello Mike. Thank you for this wonderful Tutorial. But where could you find the core.php that you mentioned or 'include' in read.php? It's not part of the code above under config folder.



Mike Dalisay < https://www.codeofaninja.com >

March 23, 2017 at 10:40 am

Hello @cebuano , you're welcome!

Thanks for bringing this to my attention. I just updated the post and added the core.php code in section 10.2 above.



Saurabh Upadhyay

October 16, 2017 at 4:35 pm

<http://localhost/api/product/read.php <>
<http://localhost/api/product/read.php>>
undefined

By using this site, you agree to our [Privacy Policy < https://www.codeofaninja.com/privacy-policy >](https://www.codeofaninja.com/privacy-policy)
and [Terms of Use < https://www.codeofaninja.com/terms-of-use > .](https://www.codeofaninja.com/terms-of-use)

OK



Mike Dalisay < https://www.codeofaninja.com >

October 20, 2017 at 9:33 pm

Hi @disqus_jHq6n81WbA, it looks like you missed a closing bracket. Would you show us the full error message?



Abdelaal Atif

February 25, 2018 at 4:10 pm

got same issue kinda

SyntaxError: JSON.parse: unexpected character at line 1 column 1 of the JSON data
Solved it thanks for the extra-ordinary tuts!



Mike Dalisay < https://www.codeofaninja.com >

February 28, 2018 at 12:16 pm

Hi @abdelaalatif, I'm unable to replicate the issue. Would you tell us how did you solved it?

This site uses Akismet to reduce spam. **Learn how your comment data is processed < https://akismet.com/privacy/>**.

By using this site, you agree to our [Privacy Policy < https://www.codeofaninja.com/privacy-policy >](https://www.codeofaninja.com/privacy-policy)
[and Terms of Use < https://www.codeofaninja.com/terms-of-use >](https://www.codeofaninja.com/terms-of-use).

OK



HTTPS://WWW.CODEOFANINJA.COM/

Proudly powered by **WordPress < https://wordpress.org/ >**.

By using this site, you agree to our **Privacy Policy < https://www.codeofninja.com/privacy-policy >** and **Terms of Use < https://www.codeofninja.com/terms-of-use >**.

OK