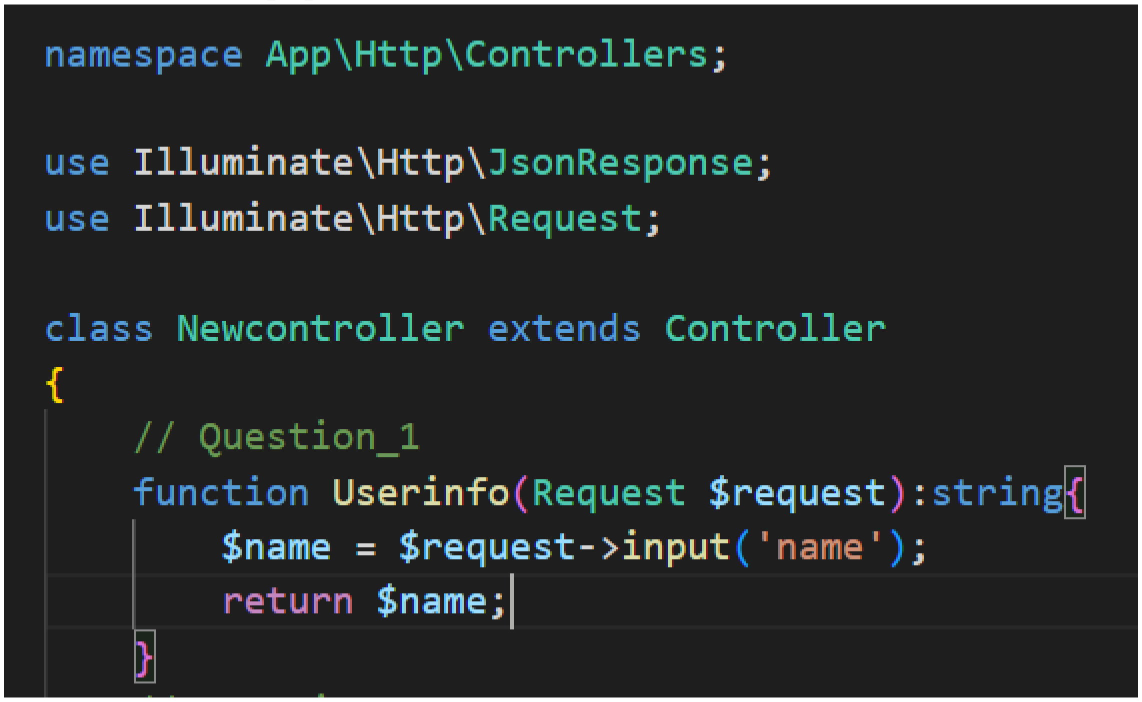
# Module 14 Assignment

1. You have a Laravel application with a form that submits user information using a POST request. Write the code to retrieve the 'name' input field value from the request and store it in a variable called $name.
2. In your Laravel application, you want to retrieve the value of the 'User-Agent' header from the current request. Write the code to accomplish this and store the value in a variable called $userAgent.
3. You are building an API endpoint in Laravel that accepts a GET request with a 'page' query parameter. Write the code to retrieve the value of the 'page' parameter from the current request and store it in a variable called $page. If the parameter is not present, set $page to null.
4. Create a JSON response in Laravel with the following data:
   ```
   {
           "message": "Success",
           "data": {
                   "name": "John Doe",
                   "age": 25
           }
   }
   ```
5. You are implementing a file upload feature in your Laravel application. Write the code to handle a file upload named 'avatar' in the current request and store the uploaded file in the 'public/uploads' directory. Use the original filename for the uploaded file.
6. Retrieve the value of the 'remember_token' cookie from the current request in Laravel and store it in a variable called $rememberToken. If the cookie is not present, set $rememberToken to null.
7. Create a route in Laravel that handles a POST request to the '/submit' URL. Inside the route closure, retrieve the 'email' input parameter from the request and store it in a variable called $email. Return a JSON response with the following data:
   ```
   {
           "success": true,
           "message": "Form submitted successfully."
   }
   ```

Answer:
1. NewController.php

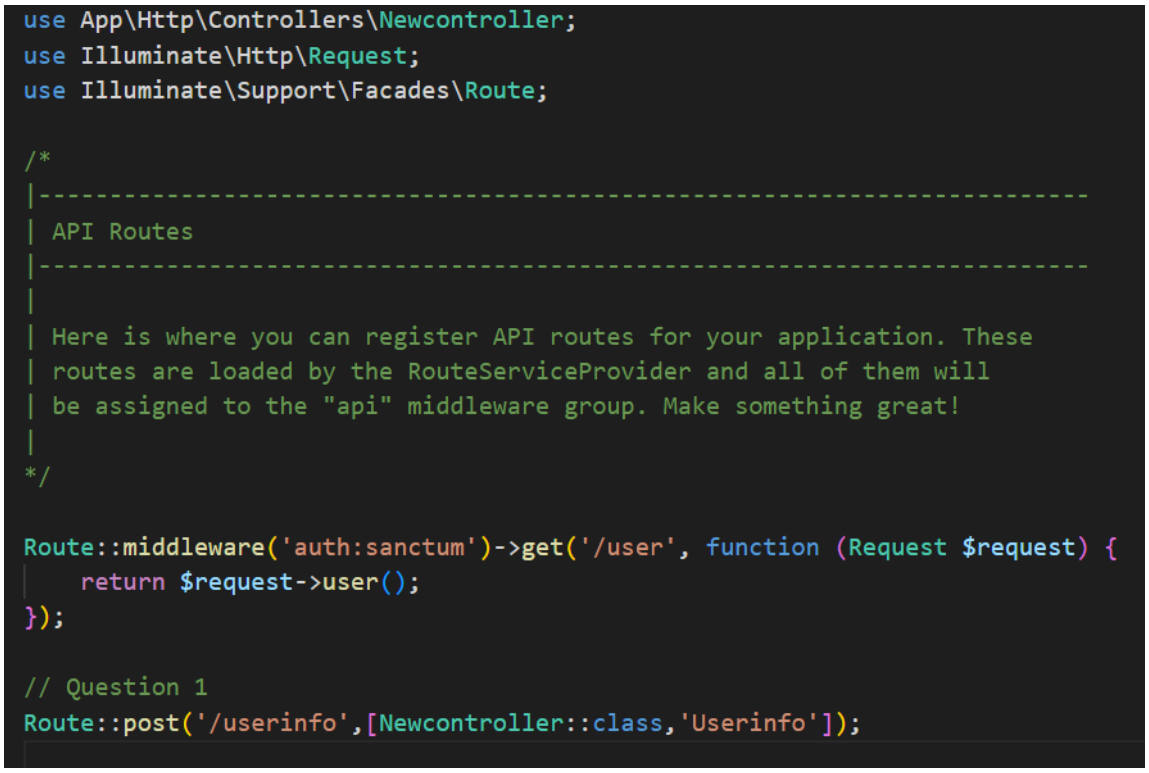


```php
namespace App\Http\Controllers;

use Illuminate\Http\JsonResponse;
use Illuminate\Http\Request;

class Newcontroller extends Controller
{
    // Question_1
    function Userinfo(Request $request):string{
        $name = $request->input('name');
        return $name;
    }
```

api.php:



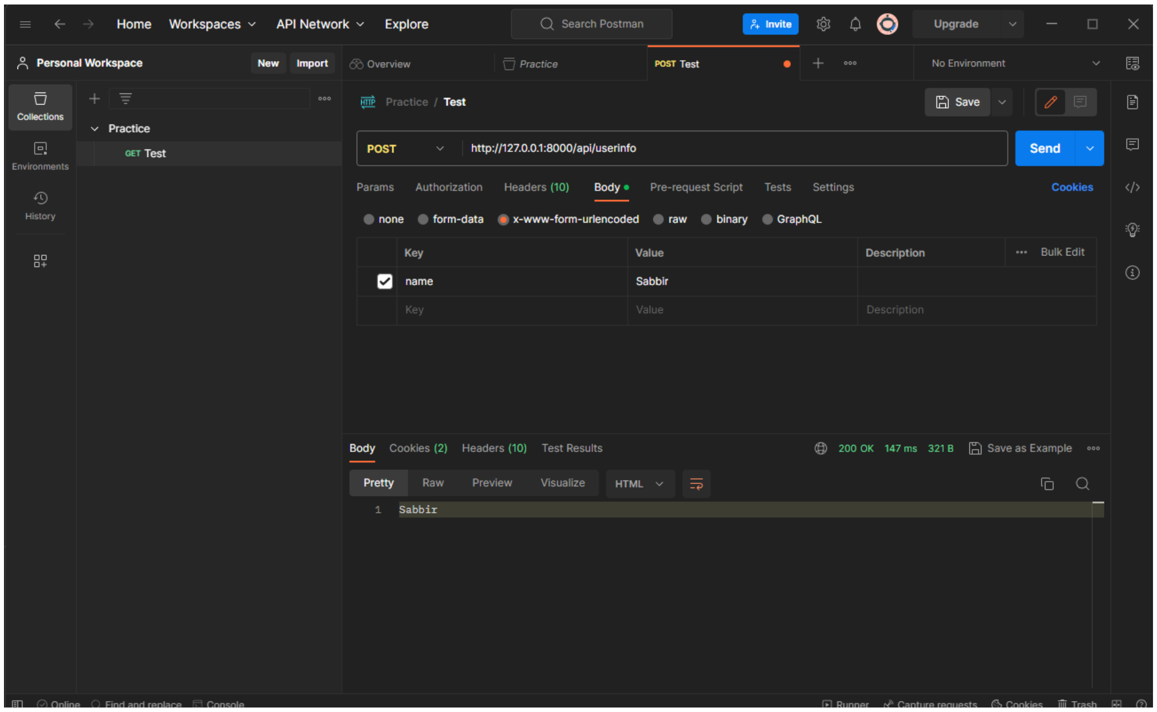
```php
use App\Http\Controllers\Newcontroller;
use Illuminate\Http\Request;
use Illuminate\Support\Facades\Route;

/*
|--------------------------------------------------------------------------
| API Routes
|--------------------------------------------------------------------------
|
| Here is where you can register API routes for your application. These
| routes are loaded by the RouteServiceProvider and all of them will
| be assigned to the "api" middleware group. Make something great!
|
*/

Route::middleware('auth:sanctum')->get('/user', function (Request $request) {
    return $request->user();
});

// Question 1
Route::post('/userinfo',[Newcontroller::class,'Userinfo']);
```

Postman:



Practice / Test

POST http://127.0.0.1:8000/api/userinfo

Params | Authorization | Headers (10) | Body ● | Pre-request Script | Tests | Settings

none | form-data | ● x-www-form-urlencoded | raw | binary | GraphQL

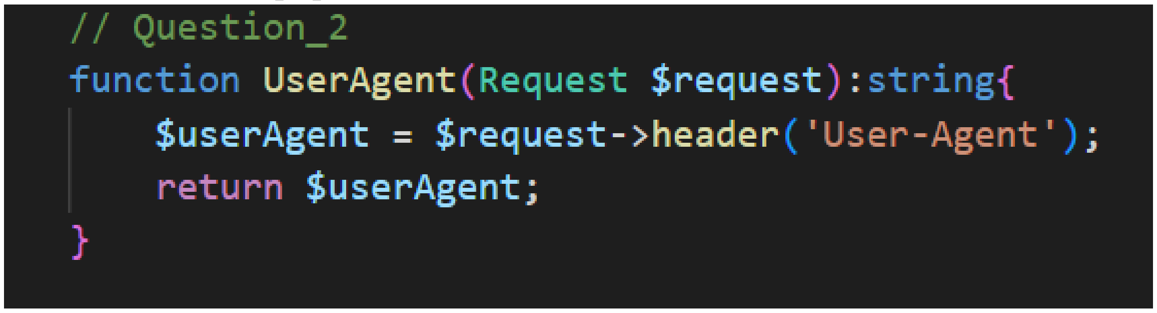
| | Key | Value | Description |
|---|---|---|---|
| ✓ | name | Sabbir | |
| | Key | Value | Description |

Body | Cookies (2) | Headers (10) | Test Results — 200 OK 147 ms 321 B — Save as Example

Pretty | Raw | Preview | Visualize | HTML

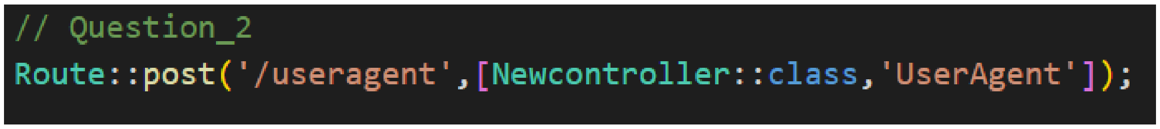
```
Sabbir
```

2. NewController.php



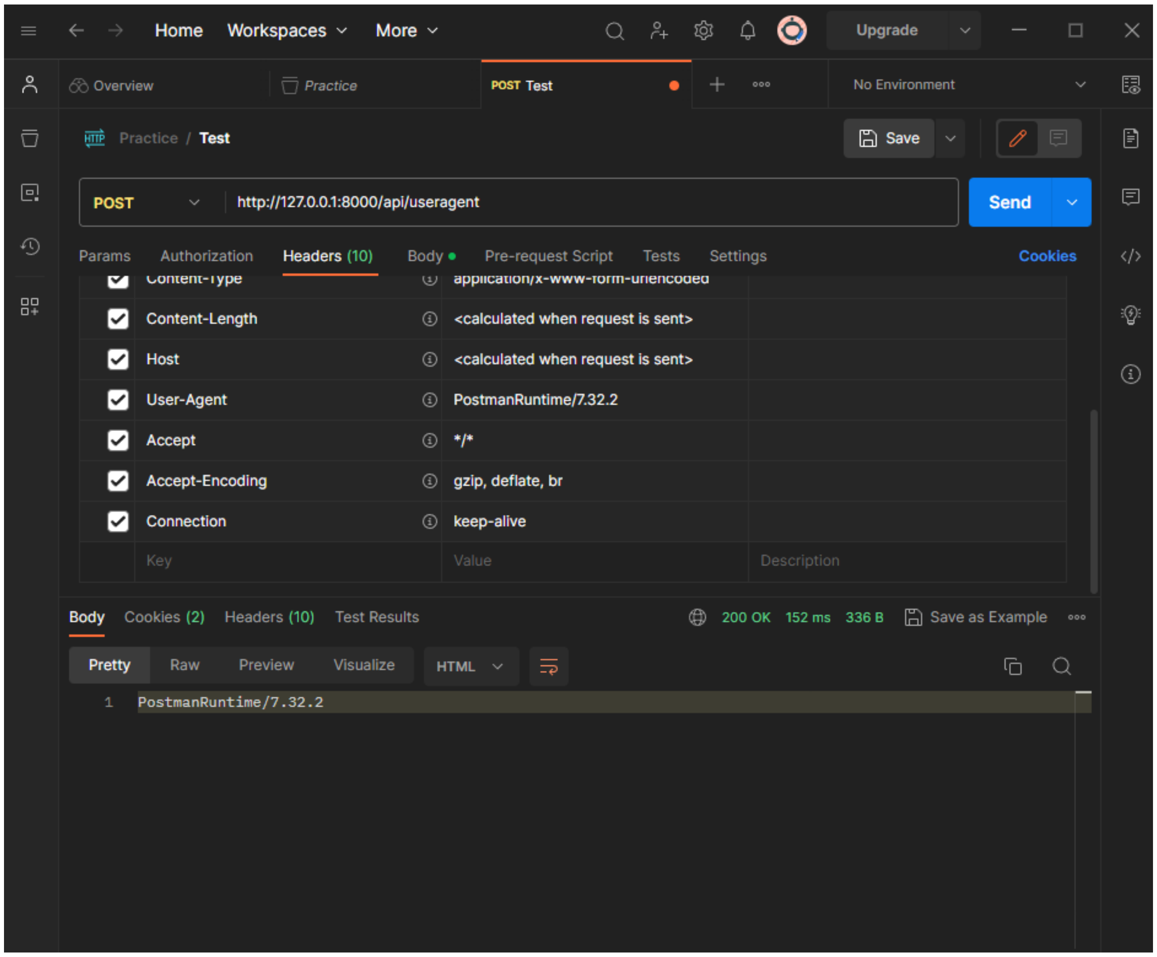
```php
// Question_2
function UserAgent(Request $request):string{
    $userAgent = $request->header('User-Agent');
    return $userAgent;
}
```

api.php:



```php
// Question_2
Route::post('/useragent',[Newcontroller::class,'UserAgent']);
```

Postman:



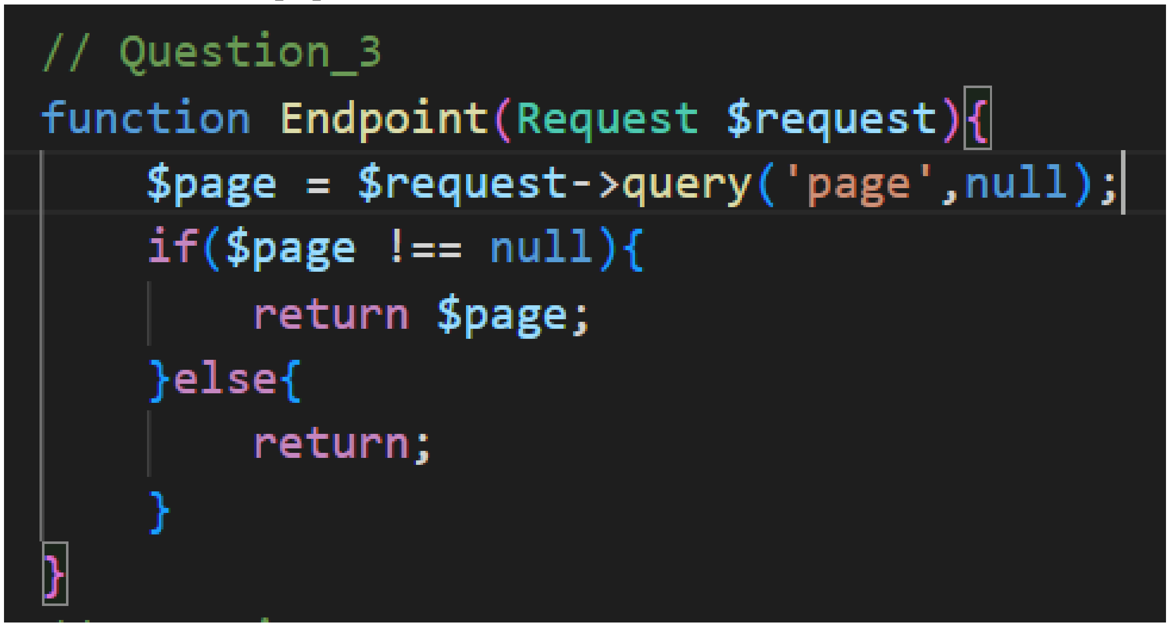
```
POST http://127.0.0.1:8000/api/useragent
```

| | Key | Value | Description |
|---|---|---|---|
| ✓ | Content-Type | application/x-www-form-urlencoded | |
| ✓ | Content-Length | <calculated when request is sent> | |
| ✓ | Host | <calculated when request is sent> | |
| ✓ | User-Agent | PostmanRuntime/7.32.2 | |
| ✓ | Accept | */* | |
| ✓ | Accept-Encoding | gzip, deflate, br | |
| ✓ | Connection | keep-alive | |

200 OK 152 ms 336 B

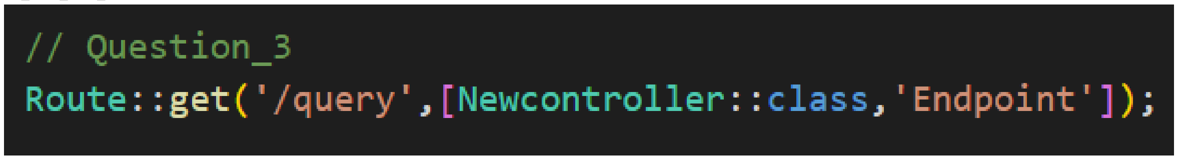
```
PostmanRuntime/7.32.2
```

3. NewController.php:



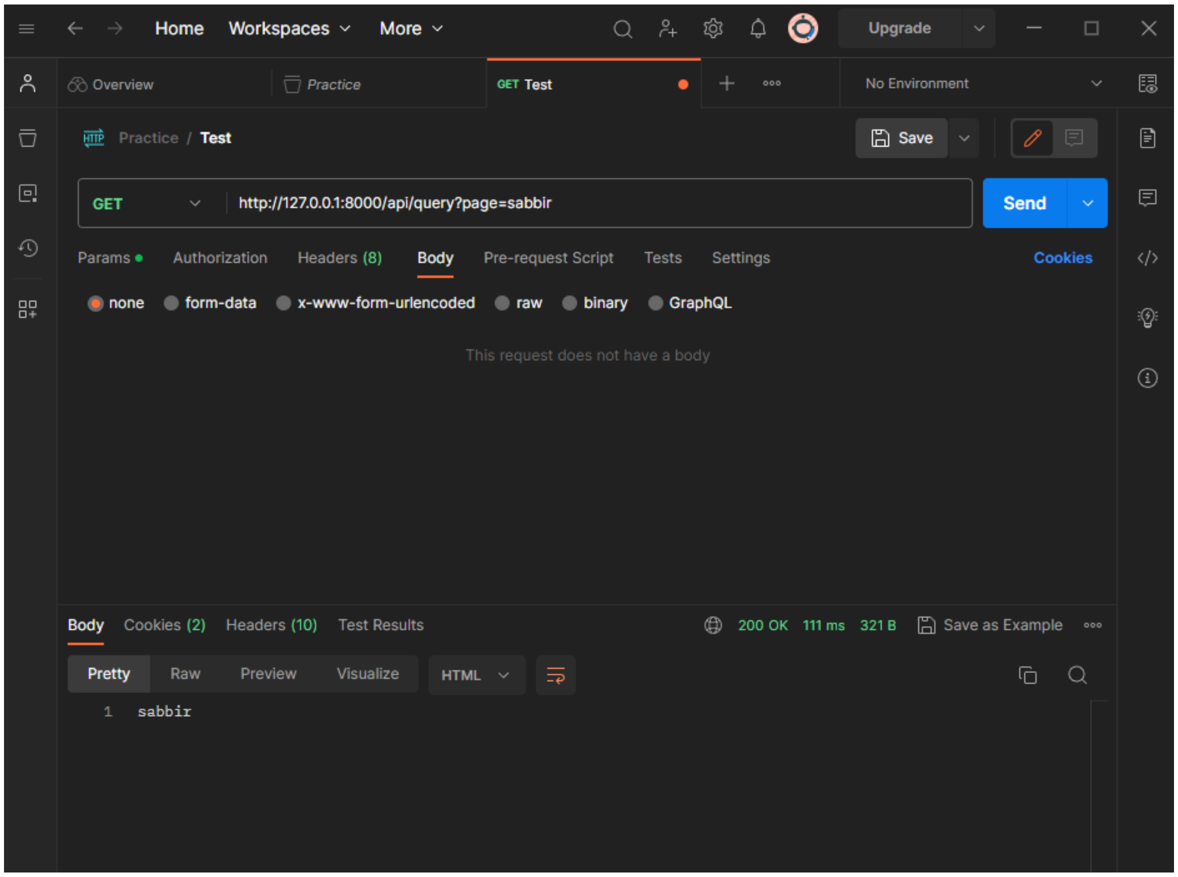
```php
// Question_3
function Endpoint(Request $request){
    $page = $request->query('page',null);
    if($page !== null){
        return $page;
    }else{
        return;
    }
}
```

api.php:



```
// Question_3
Route::get('/query',[Newcontroller::class,'Endpoint']);
```

Postman:



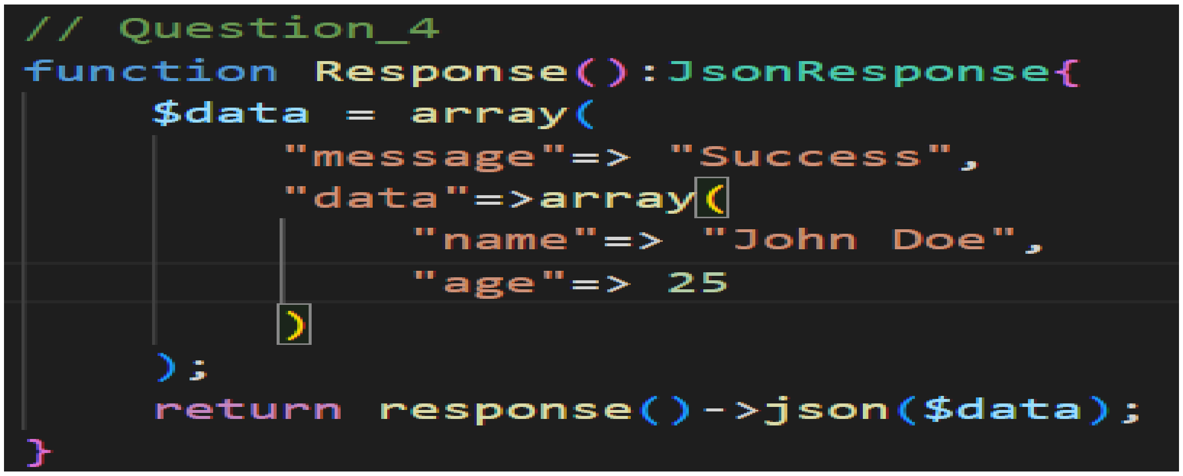
GET http://127.0.0.1:8000/api/query?page=sabbir

This request does not have a body

200 OK 111 ms 321 B

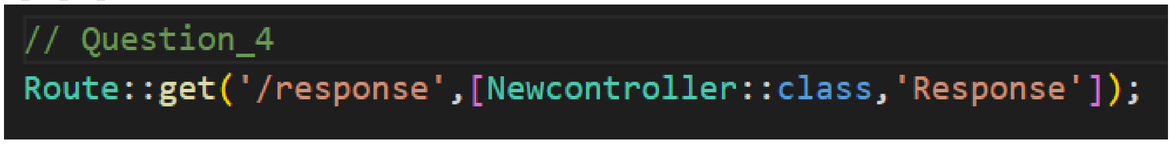
```
sabbir
```

4. NewController.php:



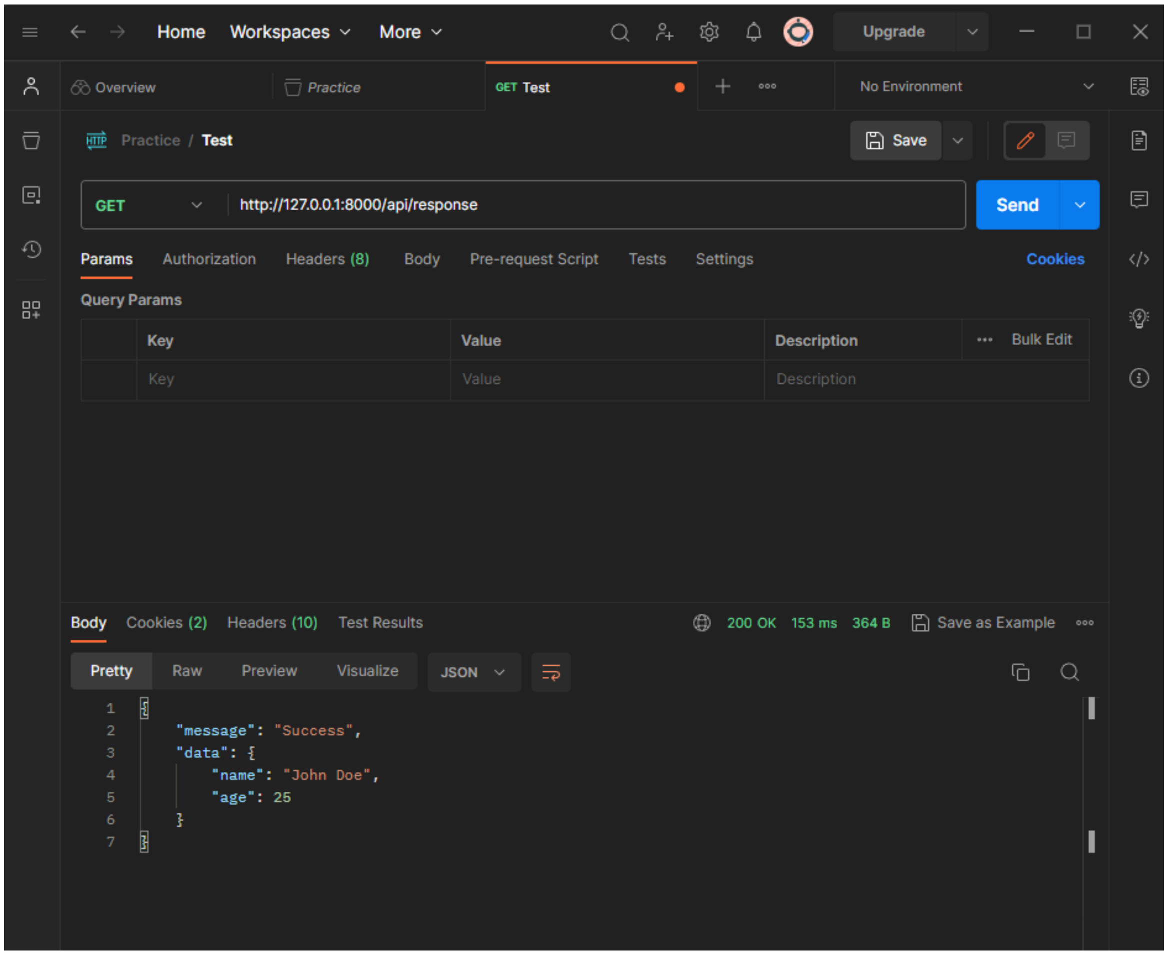
```
// Question_4
function Response():JsonResponse{
        $data = array(
                "message"=> "Success",
                "data"=>array(
                        "name"=> "John Doe",
                        "age"=> 25
                )
        );
        return response()->json($data);
}
```

api.php:



```php
// Question_4
Route::get('/response',[Newcontroller::class,'Response']);
```

Postman:

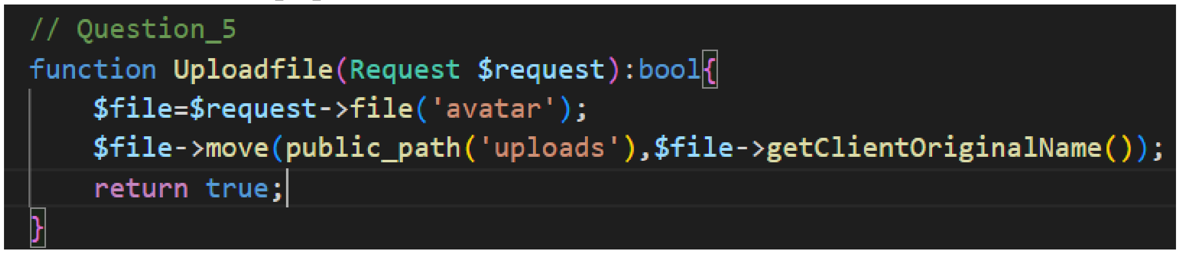


GET http://127.0.0.1:8000/api/response

200 OK 153 ms 364 B

```json
{
    "message": "Success",
    "data": {
        "name": "John Doe",
        "age": 25
    }
}
```

5. NewController.php:

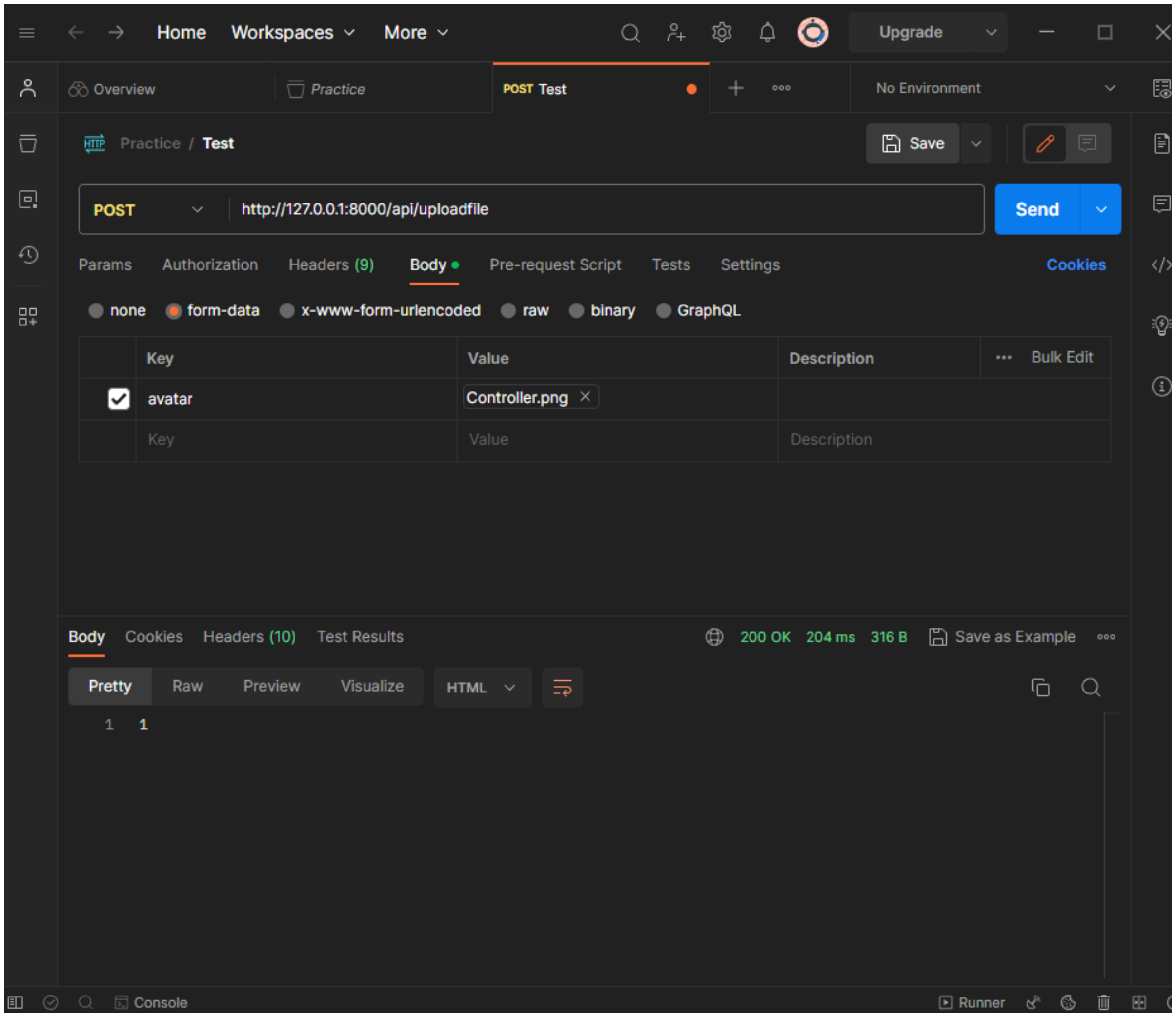


```php
// Question_5
function Uploadfile(Request $request):bool{
    $file=$request->file('avatar');
    $file->move(public_path('uploads'),$file->getClientOriginalName());
    return true;
}
```

api.php:

```php
// Question_5
Route::post('/uploadfile',[Newcontroller::class,'Uploadfile']);
```

Postman:



POST http://127.0.0.1:8000/api/uploadfile

Body: form-data

| Key | Value | Description |
|---|---|---|
| avatar | Controller.png | |

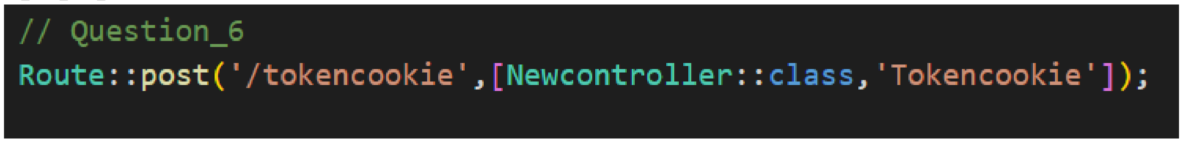
200 OK 204 ms 316 B

```
1
```

6. NewController.php:

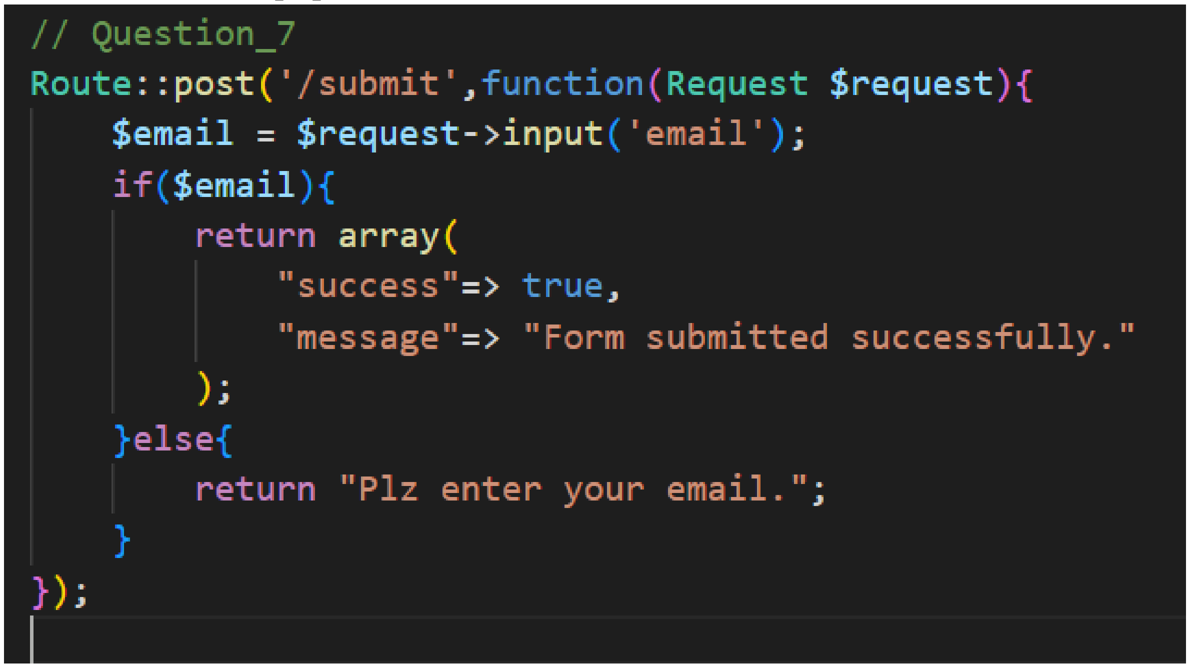
```php
// Question_6
function Tokencookie(Request $request){
    $rememberToken = $request->cookie('remember_token',null);
    return $rememberToken;
}
```

api.php:



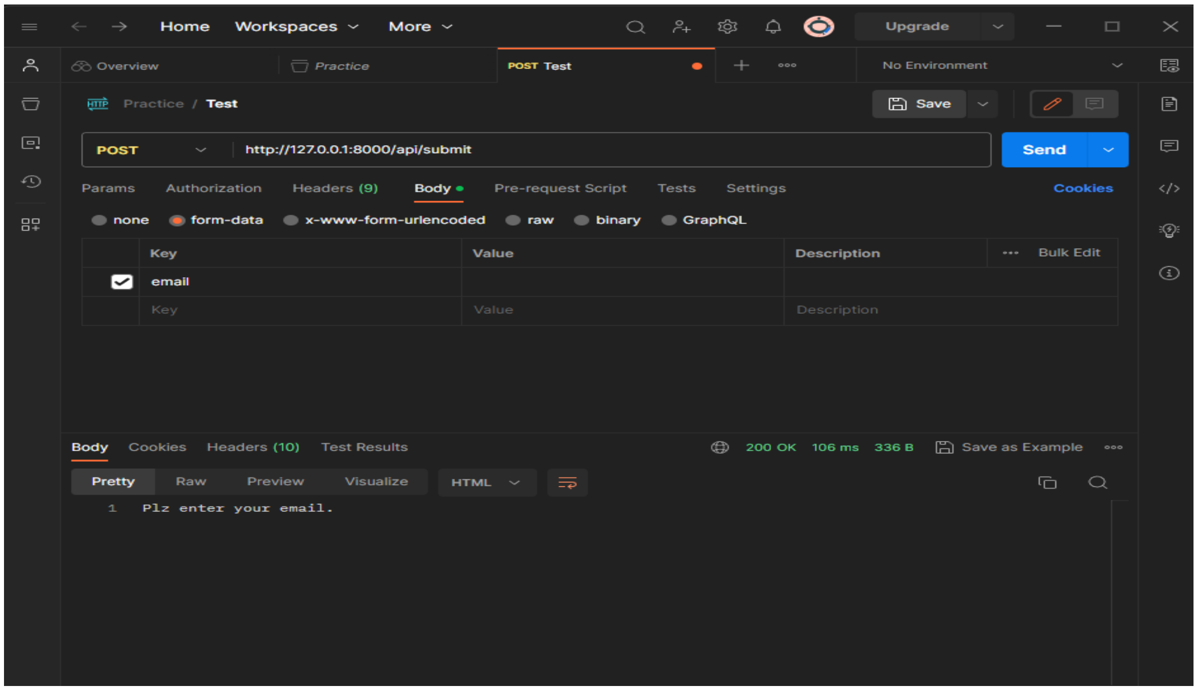
```php
// Question_6
Route::post('/tokencookie',[Newcontroller::class,'Tokencookie']);
```

7. NewController.php:



```php
// Question_7
Route::post('/submit',function(Request $request){
    $email = $request->input('email');
    if($email){
        return array(
            "success"=> true,
            "message"=> "Form submitted successfully."
        );
    }else{
        return "Plz enter your email.";
    }
});
```

Postman_1:



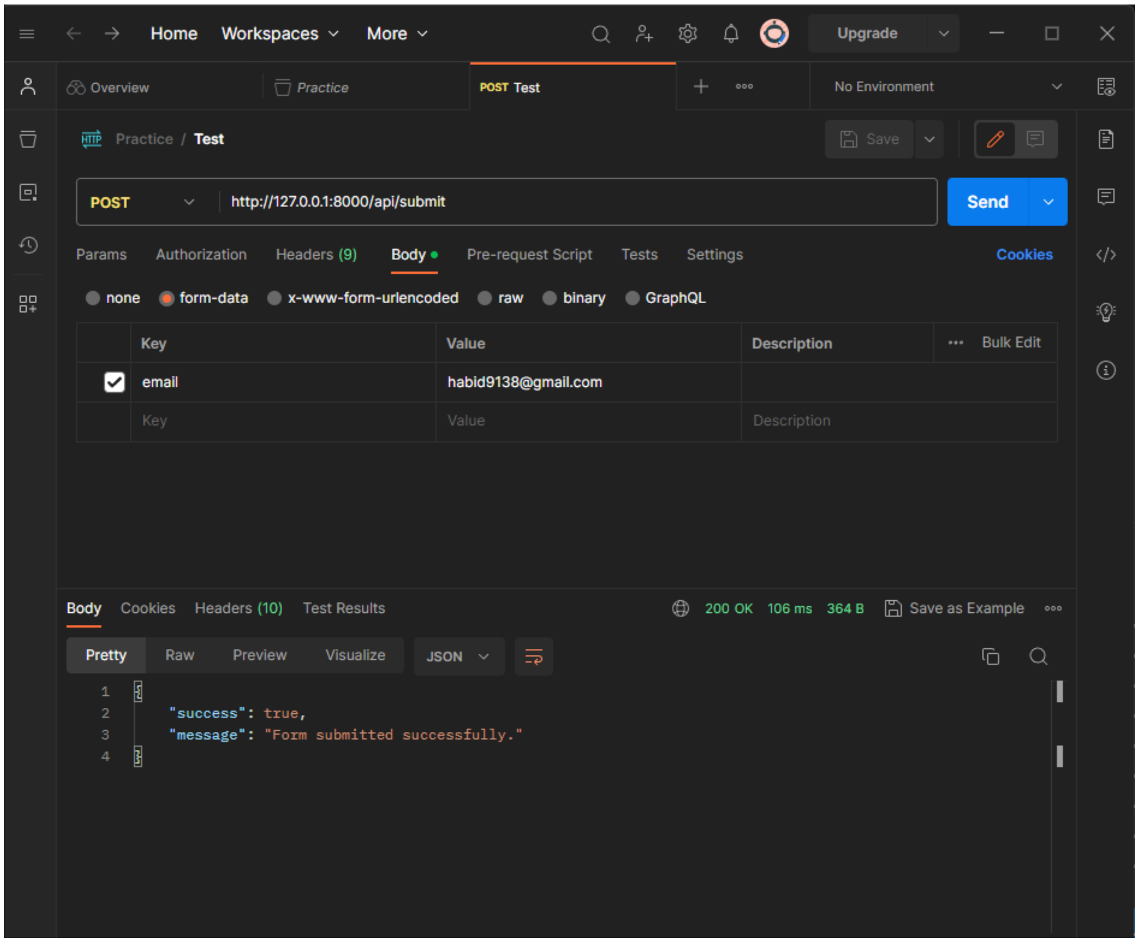
```
POST    http://127.0.0.1:8000/api/submit

Body: form-data
Key     Value    Description
email

200 OK   106 ms   336 B

1   Plz enter your email.
```

Postman_2:



POST http://127.0.0.1:8000/api/submit Send

| Key | Value | Description |
|---|---|---|
| ✓ email | habid9138@gmail.com | |
| Key | Value | Description |

200 OK 106 ms 364 B

```
{
    "success": true,
    "message": "Form submitted successfully."
}
```