|  |  |
| --- | --- |
| Rest Api Example Project | |
| Location | C:\wamp\www\rest\_api2 |
| Tutorials I looked at | <https://medium.com/@dharshithasrimal/php-rest-api-7441197312d7>  <https://wpwebinfotech.com/blog/how-to-build-simple-rest-api-in-php/>  <https://www.arubanetworks.com/techdocs/AOS-CX/10.08/HTML/rest_v1/Content/Chp_ts/res-api-res-cod-10.htm>  <https://phpdelusions.net/pdo_examples/select>  <https://stackoverflow.com/questions/8263371/how-can-prepared-statements-protect-from-sql-injection-attacks> |
| Github link | <https://github.com/dana-tal/employee_rest_api> |
| Local database | <http://localhost/phpmyadmin/sql.php?db=itemapi&table=employee&pos=0>  Database name: itemapi  Table : employee |
| Tested  with | Postman |
| Introduction | A simple, object oriented rest api that handles CRUD requests on employee database table. The database interface is using PDO , to prevent sql injections.  The api expects the input data to be in JSON format and returns JSON response as well.  The data provided by the api user is checked and tested. In case of invalid data the api throws exceptions that are caught and served back as an object containg explanation of the error, rest api code and status.  The project also contains examples of sending requests through code at index.php |

|  |  |  |
| --- | --- | --- |
| Endpoints | | |
| method | url | Description |
| GET | localhost/rest\_api2/api.php/employees | Retrieves all the employees |
| GET | localhost/rest\_api2/api.php/employees/4 | Retrives specific employee,  For example with id=4 |
| POST | url: localhost/rest\_api2/api.php/employees | Add a new employee |
| data:   {    "emp\_name": "Dana Tal",              "emp\_code": "E002",              "emp\_email": "dana@example.com",              "emp\_phone": "98765432",              "emp\_address": "gilboa 12, Hararit",              "emp\_designation": "Teacher",              "emp\_joining\_date": "2022-01-02"  } |
| PUT | url:  localhost/rest\_api2/api.php/employees/10 | Update a specific employee.  You can supply a partial list of fields. |
| data:  {  "emp\_email":"tomer@n12.co.il",      "emp\_address":"sesame street 500,Ramat Yishay",      "emp\_phone":"9999555333",      "emp\_joining\_date":"2024-06-01"  } |
| DELETE | localhost/rest\_api2/api.php/employees/4 | Delete a specific employee |

|  |  |
| --- | --- |
| Files | |
| api.php | The main gateway through which the request is handled. According to the http method and entrypoint, it dispatches the request to the proper handler. |
| curl\_helper.php | Contains a function : sendRequest by which one can send a rest api request programatically. |
| db\_access.php | Contains a class named DbAccess. The class offers public methods, each dedicated to a single entry point . The methods perform the actual required action in the database. For example, for the GET all entrypoint it offers the method getAllEmployees which reads and returns all the employees in the table. |
| dbconfig.php | The file contains the database credentials. |
| index.php | Examples of sending this rest api requests. |
| responses.php | functions for returning the api json responses. |
| rest.php | Contains a class Rest that offers methods , one for each entry point. The class methods are called directly from api.php |
| rest\_exception.php | Contains a class RestException that is derived from php regular Exception and it extends it with a status field and getStatus method. |
| validator.php | Contains a class Validator that offers validation methods such as validating an email, phone etc. |