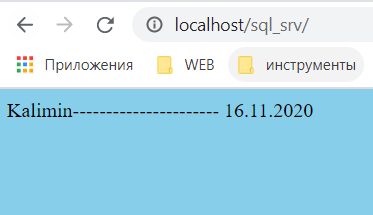
**Лабораторна робота 16.**

**Реалізація back-end REST-API для відображення вмісту однієї відомості.**

**Завдання для індивідуального виконання.**

Для БД свого варіанту організувати виведення на екран даних із таблиці, що має у своєму складі мінімум один зовнішній ключ.



index.php

<?php

    require\_once 'autorun.php';

    $query = new DBquery(3);

    $query->readFromDb();

?>

<!DOCTYPE html>

<html lang="en">

<head>

    <meta charset="UTF-8">

    <meta name="viewport" content="width=device-width, initial-scale=1.0">

    <title>Document</title>

</head>

<body style="background-color: skyblue;">

    <?php foreach($query->getRows() as $row) :?>

    <?php echo $row->getSurename() . "----------------------" ;?>

    <?php echo $row->getDateOrder() . "<br>" ;?>

    <?php endforeach; ?>

</body>

</html>

myquery.php

<?php

class Myquery {

    private $surename;

    private $date\_order;

    function \_\_construct(string $surename = '', DateTime $date\_order = null){

        $this->surename = $surename;

        $this->date\_order = $date\_order;

    }

    function getSurename(){

        return $this->surename;

    }

    function getDateOrder(){

        return $this->date\_order->format("d.m.Y");

    }

}

class DBquery {

    private $cod;

    private $rows = [];

    function \_\_construct(int $cod){

        $this->cod = $cod;

    }

    function readFromDb() {

        $db = new ClientsDB();

        $sql = "select c.surename, o.date\_order

                from orders o

                inner join clients c on o.client\_id = c.id where o.id = ". $this->cod;

        if ($exam = $db->makeQuery($sql)){

            foreach($exam as $row) {

                $this->rows[] = new Myquery($row['surename'], $row['date\_order']);

            }

        }

        return true;

    }

    function getRows(){

        return $this->rows;

    }

}

db.php

<?php

    class ClientsDB {

        private $errText = "";

        private $conn;

        function getErrText(){

            return $this->errText;

        }

        function connect(){

            try{

                $srvName = "DESKTOP-TPBKNFG";

                $connOptions = [

                    "Database"=>"Kalimin\_308\_12",

                    "UID"=>"kalimin",

                    "PWD"=>"123",

                    "CharacterSet"=>"UTF-8"

                ];

                $this->conn = sqlsrv\_connect($srvName, $connOptions);

                if($this->conn == false) {

                    $this->errText = FormatErrors(sqlsrv\_errors());

                    return false;

                }

                return true;

            }

            catch(Exception $e){

                $this->errText = "Unknown error";

                return false;

            }

        }

        function disconnect(){

            sqlsrv\_close($this->conn);

        }

        function makeQuery($sqlText, $flg\_disconnect = true){

            $this->errText = "";

            $result = [];

            try{

                if($this->conn || $this->connect()){

                    $sql\_stmt = sqlsrv\_query($this->conn, $sqlText);

                    if (!$sql\_stmt) {

                        $this->errText = sqlsrv\_errors();

                        return false;

                    }

                    while($row = sqlsrv\_fetch\_array($sql\_stmt, SQLSRV\_FETCH\_ASSOC)){

                        $result[] = $row;

                    }

                    sqlsrv\_free\_stmt($sql\_stmt);

                    return $result;

                }

        }

        catch(Exception $e) {

            $this->errText = "Unknown error!";

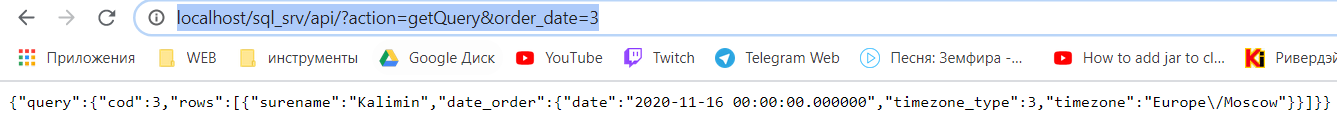
            return false;

        }

    }

}

Реалізувати back-end REST-API сервіс, що повертає ті ж самі дані у форматі JSON за GET запитом на визначений URL.



api/index.php

<?php

include \_\_DIR\_\_ . '/../autorun.php';

if($\_GET['action'] == 'getQuery') {

    $query = new DBquery((int)$\_GET['order\_date']);

    $query->readFromDB();

    $data = ['query' => $query];

} else {

    $data = ['error' => 'bad request'];

}

header("Access-Control-Allow-Origin: \*");

header("Content-type:application/json");

header("Accept-Language: \*");

echo json\_encode($data);

Имплементация интерфейса JsonSerializable класом DBquery

class DBquery implements JsonSerializable {

    public function jsonSerialize(){

        return [

            'cod' => $this->cod,

            'rows' => $this->rows

        ];

    }

Имплементация интерфейса JsonSerializable класом Myquery

class Myquery implements JsonSerializable{

    public function jsonSerialize(){

        return [

            'surename' => $this->surename,

            'date\_order' => $this->date\_order

        ];

    }