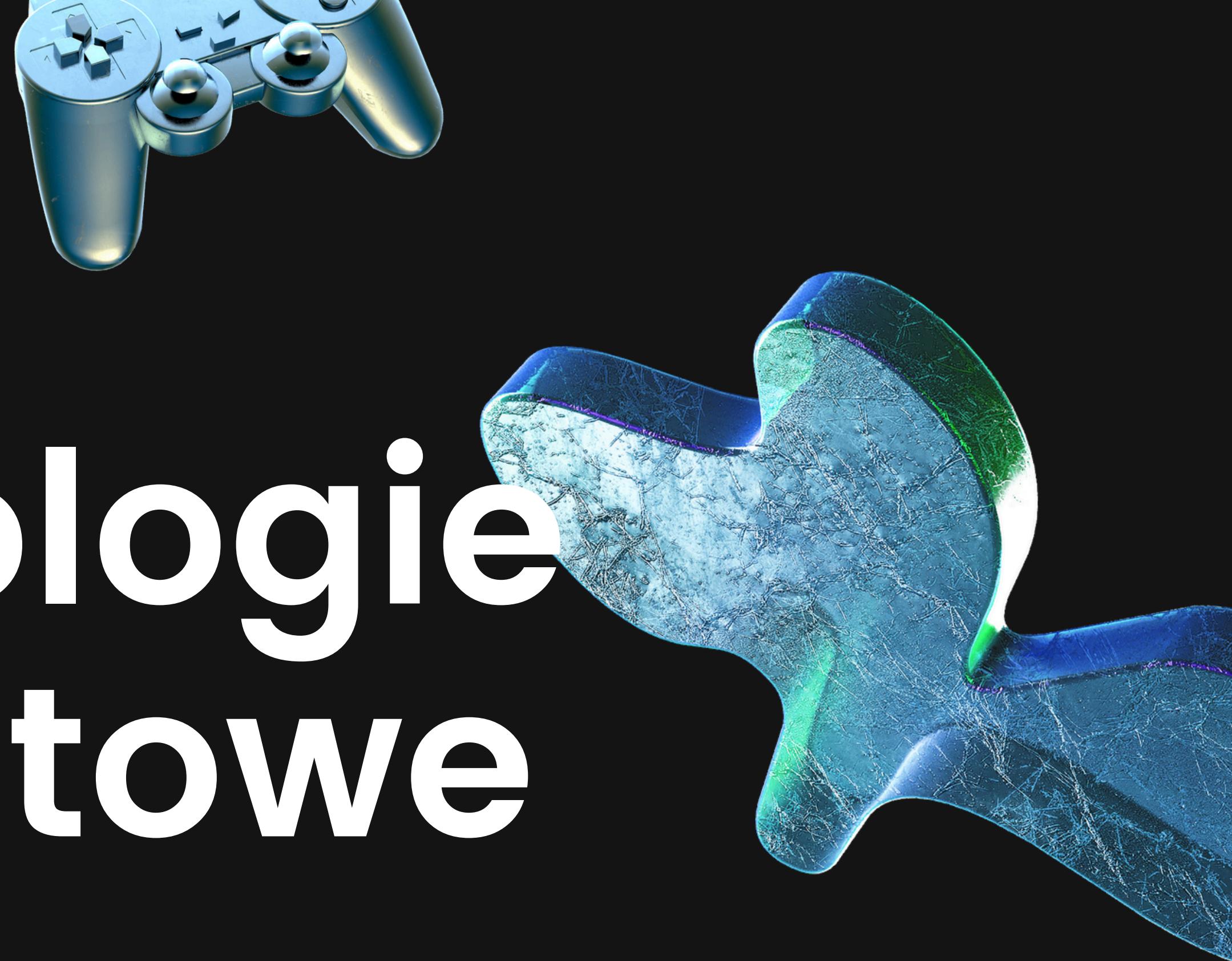
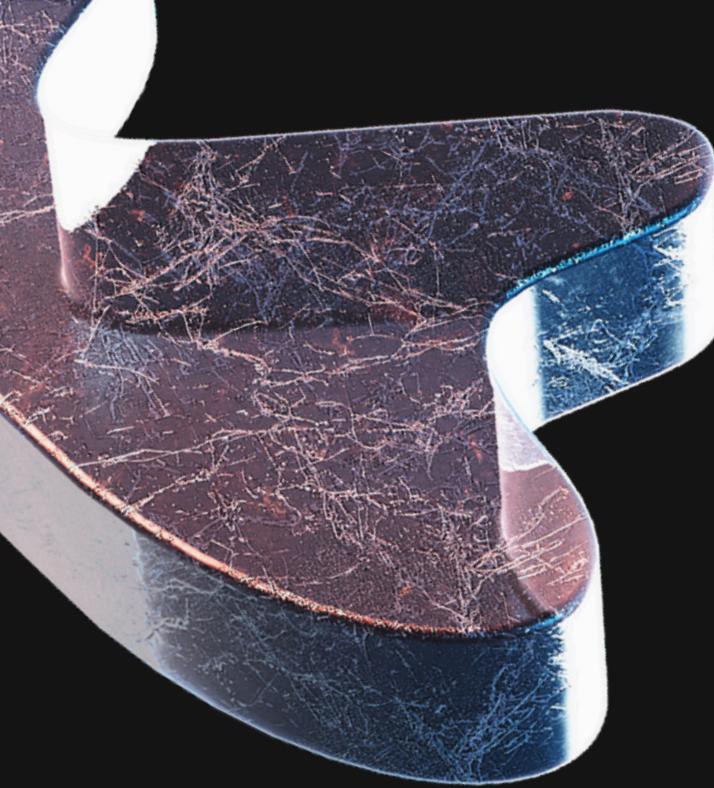


# Technologie internetowe

Antoni Marcinek i Kacper Kureń





## Opis projektu

Projekt “Car Search” to aplikacja webowa do zarządzania informacjami o samochodach w bazie danych. Pozwala na odczytywanie, dodawanie i usuwanie rekordów za pomocą przyjaznego interfejsu użytkownika.





## Specyfikacja

Wyświetlanie  
danych pojazdów  
z bazy danych.

Dodawanie nowych  
samochodów.

Usuwanie  
istniejących  
rekordów.



## Technologie

HTML

CSS

JavaScript

SQL

PHP





## Instrukcja do XAMPP

Pobierz i zainstaluj XAMPP ze strony  
<https://www.apachefriends.org/pl/index.html>  
zgodnie z instrukcjami na stronie.

Po zainstalowaniu uruchom XAMPP i włącz  
moduły Apache oraz MySQL.

Przejdź do panelu administracyjnego MySQL  
poprzez przeglądarkę, wpisując  
<http://localhost/phpmyadmin/>. Stwórz nową  
bazę danych o nazwie **“car\_database”**, utworz w  
niej tabelę **“cars”**, a następnie uzupełnij tę tabelę  
90 samochodami przygotowanym skryptem.

Umieść pliki projektu w folderze htdocs w  
folderze, gdzie zainstalowano XAMPP.

Nazwa	Data modyfikacji	Typ
dashboard	03.04.2024 12:00	
img	03.04.2024 12:00	
technologie-internetowe-projekt-main	14.06.2024 18:12	
webalizer	03.04.2024 12:00	
css	14.06.2024 18:12	Folder plików
js	14.06.2024 18:12	Folder plików
php	14.06.2024 18:12	Folder plików
sql	14.06.2024 18:12	Folder plików
car_image	14.06.2024 18:12	Plik PNG
index	14.06.2024 18:15	PHP Source File



## Kod HTML

```
<!DOCTYPE html>
<html lang="pl">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>Car Search</title>
    <link rel="stylesheet" href="css/styles.css">
    <!-- Link to Font Awesome for icons -->
    <link rel="stylesheet" href="https://cdnjs.cloudflare.com/ajax/libs/font-awesome/5.15.4/css/all.min.css">
    <script defer src="js/script.js"></script>
</head>
<body>
    <!-- Main container for the content -->
    <div class="container">
        <h1>Car Search</h1>
        <!-- Div element where fetched data will be displayed -->
        <div id="data"></div>
        <h2>Add new car:</h2>
        <!-- Form for adding a new car record -->
        <form id="car_form">
            <!-- Input fields for car details -->
            <input type="text" id="vin" placeholder="Vin number">
            <input type="text" id="brand" placeholder="Brand">
            <input type="text" id="model" placeholder="Model">
            <input type="text" id="year" placeholder="Year">
            <input type="text" id="mileage" placeholder="Mileage">
            <input type="text" id="engine" placeholder="Engine">
            <!-- Select field for fuel type -->
            <select id="fuel_type">
                <option value="">Fuel type</option>
                <option value="Petrol">Petrol</option>
                <option value="Diesel">Diesel</option>
                <option value="Electric">Electric</option>
            </select>
            <input type="number" id="owner_count" placeholder="Owner count">
            <!-- Select field for damage state -->
            <select id="damage_state">
                <option value="">Damage state</option>
                <option value="Very good">Very good</option>
                <option value="Good">Good</option>
            </select>
            <input type="text" id="photo_path" placeholder="Link to photo">
            <input type="text" id="price" placeholder="Price">
            <input type="text" id="phone_number" placeholder="Phone number">
            <!-- Submit button for the form -->
            <button type="submit">Submit</button>
        </form>
    </div>
</body>
</html>
```



## styles.css

```
body {
    font-family: Arial, sans-serif;
    background-color: #f5f5f5;
    margin: 0;
    padding: 0;
}

.container {
    max-width: 800px;
    margin: 20px auto;
    padding: 20px;
    background-color: #fff;
    border-radius: 8px;
    box-shadow: 0 2px 4px rgba(0, 0, 0, 0.1);
    width: 100%
}

.card {
    background-color: white;
    border-radius: 10px;
    box-shadow: 0 2px 5px rgba(0,0,0,0.15);
    display: flex;
    justify-content: space-evenly;
    align-items: center;
    padding: 20px;
    margin: 10px auto;
    width: 95%;
    transition: transform 0.3s ease;
}

.card img {
    width: 200px;
    height: auto;
    border-radius: 10px;
    margin-right: 20px;
    transition: transform 0.3s ease;
}

.card img:hover {
    transform: scale(1.5);
}

.card-content {
    display: flex;
    flex-direction: column;
    justify-content: center;
    width: 40%;
}

.card-content h3 {
    margin: 0;
    font-size: 24px;
}

.card-content p {
    margin: 5px 0;
    font-size: 16px;
}

.card-content .details {
    display: flex;
    flex-wrap: wrap;
}

.card-content .details p {
    width: 50%;
}

.buttons {
    display: flex;
    flex-direction: column;
    justify-content: center;
}

.buttons .phone-button {
    font-size: 16px;
    padding: 0px;
    background-color: #007bff;
    color: white;
    cursor: pointer;
    transition: background-color 0.3s ease;
}

.buttons button:hover {
    background-color: #0056b3;
}

/* Text that appears on hover */
.buttons .phone-button::after {
    content: attr(data-hover-text);
    position: absolute;
    top: 50%;
    left: 50%;
    transform: translate(-50%, -50%);
    opacity: 0;
    color: white;
    font-size: 14px;
    transition: opacity 0.3s ease;
}

/* Show the text on hover */
.buttons .phone-button:hover::after {
    opacity: 1;
}

.buttons .phone-button:hover p {
    visibility: hidden;
    opacity: 0;
    transition: opacity 0.3s ease;
}

h1,
h2 {
    text-align: center;
    color: #333;
}

#data {
    margin-top: 20px;
}

#data div {
    margin-top: 5px;
}

form {
    margin-top: 20px;
    display: flex;
    flex-direction: column;
}

select,
input {
    padding: 10px;
    margin-bottom: 10px;
    border: 1px solid #ccc;
    border-radius: 4px;
}

button {
    padding: 10px;
    border: none;
    background-color: #007bff;
    color: white;
    border-radius: 4px;
    cursor: pointer;
    animation: all 0.2s ease-in;
}

button:hover {
    background-color: #0056b3;
    position: relative;
    top: 1px;
}
```



## Kod JS

```
document.addEventListener("DOMContentLoaded", function () {
    // Fetching data from the database
    function display_data() {
        // Fetch data from the server using the fetch API
        fetch('php/read.php')
            .then(response => response.json()) // Parse the response as JSON
            .then(data => {
                let dataElement = document.getElementById('data') // Get the element with ID 'data'
                dataElement.innerHTML = "" // Clear any existing content in the 'data' element

                if (data.length > 0) {
                    // Iterate over each item in the data array
                    data.forEach(item => {
                        // Create a new div element
                        let div = document.createElement('div')
                        div.className = 'card' // Set the class of the div to 'card'

                        // Set the inner HTML of the div to include the item details
                        div.innerHTML =
                            `![${item.brand} ${item.model}](${item.photo_path})

### ${item.brand} ${item.model}



${item.fuel_type}



${item.mileage} km



${item.engine}



${item.year}



${item.damage_state}



### PLN


                    // Append the new div to the 'data' element
                    dataElement.appendChild(div)
                }
            })
        } else {
            // If there is no data, display a message indicating no data
            dataElement.innerHTML = "No data to display."
        }
    }
}).catch(error => console.error('Error:', error)) // Log any errors to the console
}
```



## read.php

```
<?php
// Connection to DB
$connection = new mysqli('localhost', 'root', '', 'car_database');

// Connection verification
if ($connection->connect_error) {
    die("Connection error: " . $connection->connect_error);
}

// SQL query
$sql = "SELECT * FROM cars";

// Query execution
$result = $connection->query($sql);

// Converting the query result to an array
$rows = array();
if ($result->num_rows > 0) {
    while($row = $result->fetch_assoc()) {
        $rows[] = $row;
    }
}

// Returning data in JSON format
echo json_encode($rows);

// Closing the connection to the database
$connection->close();
?>
```



## Kod JS

```
// Adding a new record
document.getElementById('car_form').addEventListener('submit', function (event) {
    event.preventDefault()
    // Get form values
    vin = document.getElementById('vin').value
    brand = document.getElementById('brand').value
    model = document.getElementById('model').value
    year = document.getElementById('year').value
    mileage = document.getElementById('mileage').value
    engine = document.getElementById('engine').value
    fuel_type = document.getElementById('fuel_type').value
    owner_count = document.getElementById('owner_count').value
    damage_state = document.getElementById('damage_state').value
    photo_path = document.getElementById('photo_path').value
    price = document.getElementById('price').value
    phone_number = document.getElementById('phone_number').value

    // Ensure no field is empty
    //The trim() method in JavaScript is used to remove whitespace from both ends of a string.
    //This includes spaces, tabs, and other whitespace characters.
    if (vin.trim() !== '' && brand.trim() !== '' && model.trim() !== '' &&
        year.trim() !== '' && mileage.trim() !== '' && engine.trim() !== '' &&
        owner_count.trim() !== '' && photo_path.trim() !== '' && price.trim() !== '' && phone_number.trim() !== '') {
        fetch('php/add.php', {
            method: 'POST',
            headers: {
                'Content-Type': 'application/x-www-form-urlencoded',
            },
            body: `vin=${vin}&brand=${brand}&model=${model}&year=${year}
&mileage=${mileage}&engine=${engine}&fuel_type=${fuel_type}
&owner_count=${owner_count}&damage_state=${damage_state}
&photo_path=${photo_path}&price=${price}&phone_number=${phone_number}`,
        })
        .then(response => response.json())
        .then(data => {
            console.log(data.message)
            display_data()
            // Clear form fields after successful submission
            document.getElementById('vin').value = ''
            document.getElementById('brand').value = ''
            document.getElementById('model').value = ''
            document.getElementById('year').value = ''
            document.getElementById('mileage').value = ''
            document.getElementById('engine').value = ''
            document.getElementById('fuel_type').value = ''
            document.getElementById('owner_count').value = ''
            document.getElementById('damage_state').value = ''
            document.getElementById('photo_path').value = ''
            document.getElementById('price').value = ''
            document.getElementById('phone_number').value = ''

        })
        .catch(error => console.error('Error:', error))
    } else {
        alert('No field can be empty!')
    }
})
```



## add.php

```
<?php
// Connection to the database
$connection = new mysqli('localhost', 'root', '', 'car_database');

// Verify the connection
if ($connection->connect_error) {
    die("Connection error: " . $connection->connect_error);
}

// Verify the data
if (isset($_POST['vin']) && isset($_POST['brand']) && isset($_POST['model']) &&
    isset($_POST['year']) && isset($_POST['mileage']) && isset($_POST['engine']) &&
    isset($_POST['fuel_type']) && isset($_POST['owner_count']) && isset($_POST['damage_state']) &&
    isset($_POST['photo_path']) && isset($_POST['price']) && isset($_POST['phone_number'])) {
    $vin = $_POST['vin'];
    $brand = $_POST['brand'];
    $model = $_POST['model'];
    $year = $_POST['year'];
    $mileage = $_POST['mileage'];
    $engine = $_POST['engine'];
    $fuel_type = $_POST['fuel_type'];
    $owner_count = $_POST['owner_count'];
    $damage_state = $_POST['damage_state'];
    $photo_path = $_POST['photo_path'];
    $price = $_POST['price'];
    $phone_number = $_POST['phone_number'];

    // SQL insert query
    $sql = "INSERT INTO cars (vin, brand, model, year, mileage, engine, fuel_type, owner_count, damage_state, photo_path, price, phone_number)
VALUES ('$vin', '$brand', '$model', '$year', '$mileage', '$engine', '$fuel_type', '$owner_count', '$damage_state', '$photo_path', '$price', '$phone_number')";

    // Execute the query
    if ($connection->query($sql) === TRUE) {
        $response = array('status' => 'success', 'message' => 'New record has been added.');
        echo json_encode($response);
    } else {
        $response = array('status' => 'error', 'message' => 'Error adding new record: ' . $connection->error);
        echo json_encode($response);
    }
} else {
    $response = array('status' => 'error', 'message' => 'Invalid request.');
    echo json_encode($response);
}

// Close the connection to the database
$connection->close();
?>
```



## Kod JS

```
// Function to delete data
function delete_data(id) {
  fetch(`php/delete.php?id=${id}`, {
    method: 'DELETE',
  })
    .then(response => response.json())
    .then(data => {
      console.log(data.message)
      location.reload() // Reload the page after deletion
    })
    .catch(error => console.error('Error:', error))
}
```



## delete.php

```
<?php
// Connection to the database
$connection = new mysqli('localhost', 'root', '', 'car_database');

// Connection verification
if ($connection->connect_error) {
  die("Connection error: " . $connection->connect_error);
}

// Verification if an ID to delete was passed
if (isset($_GET['id'])) {
  $id = $_GET['id'];

  // SQL query to delete a record
  $sql = "DELETE FROM cars WHERE id=$id";

  // Executing the query
  if ($connection->query($sql) === TRUE) {
    $response = array('status' => 'success', 'message' => 'Record has been deleted.');
    echo json_encode($response);
  } else {
    $response = array('status' => 'error', 'message' => 'Error deleting record: ' . $connection->error);
    echo json_encode($response);
  }
} else {
  $response = array('status' => 'error', 'message' => 'Invalid request.');
  echo json_encode($response);
}

// Closing the connection to the database
$connection->close();
?>
```



## Kod SQL

```
--  
-- Database: `car_database`  
--  
CREATE Database if NOT EXISTS car_database;  
--  
-- Table structure for the table `cars`  
--  
CREATE TABLE `cars` (  
    `id` int(11) NOT NULL,  
    `vin` varchar(17) NOT NULL,  
    `brand` varchar(50) NOT NULL,  
    `model` varchar(50) NOT NULL,  
    `year` int(11) NOT NULL,  
    `mileage` varchar(11) NOT NULL,  
    `engine` varchar(50) DEFAULT NULL,  
    `fuel_type` varchar(50) DEFAULT NULL,  
    `owner_count` int(11) DEFAULT NULL,  
    `damage_state` text DEFAULT NULL,  
    `photo_path` varchar(255) DEFAULT NULL,  
    `price` varchar(25) DEFAULT NULL,  
    `phone_number` varchar(20) DEFAULT NULL  
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4_general_ci;  
  
--  
-- Indeks for the table `cars`  
--  
ALTER TABLE `cars`  
    ADD PRIMARY KEY (`id`),  
    ADD UNIQUE KEY `vin` (`vin`);  
  
--  
-- AUTO_INCREMENT for table `cars`  
--  
ALTER TABLE `cars`  
    MODIFY `id` int(11) NOT NULL AUTO_INCREMENT, AUTO_INCREMENT=91;
```



```

INSERT INTO `cars` (`id`, `vin`, `brand`, `model`, `year`, `mileage`, `engine`, `fuel_type`, `owner_count`, `damage_state`, `photo_path`, `price`, `phone_number`) VALUES
(1, 'WAUZZZ8P1AA000001', 'Audi', 'A3', 2010, '321 000', '1.6L I4', 'Petrol', 2, 'good', 'https://encrypted-tbn0.gstatic.com/images?q=tbn:ANd9GctMvUlr7IrUGHQJapduwaXVz5BvpItg6VoaQQc7nPeFyTr70sAnYYI4ebTvyMoib9s7M&usqp=CAU', '35 000.00', '789-123-456'),
(2, 'WVZZZ1KZA000002', 'Volkswagen', 'Golf', 2010, '120 000', '1.4L TSI', 'Petrol', 1, 'good', 'https://a.allegroimg.com/s1024/0ced74/154b06d4d299af19b6a9b270e2e', '25 000.00', '890-234-567'),
(3, 'WBA3A31000K000003', 'BMW', '3 Series', 2012, '680 000', '2.0L I4', 'Diesel', 1, 'good', 'https://a.allegroimg.com/s1024/0c8a59/51e7b0624c23bbe1833fb4c35b06', '23 000.00', '901-345-678'),
(4, 'VF1KZ17864000004', 'Renault', 'Clio', 2015, '120 000', '1.2L I4', 'Petrol', 1, 'very good', 'https://interauto.krakow.pl/wp-content/uploads/2023/03/Renault-Clio-1-1824x78.jpg', '30 000.00', '812-456-789'),
(5, 'VF7SBRFB72000005', 'Citroën', 'C4', 2014, '400 000', '1.6L I4', 'Diesel', 2, 'very good', 'https://upload.wikimedia.org/wikipedia/commons/d/dd/2015\_Citro%C3%A8n\_C4\_Picasso\_%2887\_MY15%29\_Exclusive\_e-THP\_wagon\_%282015-06-15%29\_01.jpg', '45 000.00', '123-567-890'),
(6, 'W0L0XCE7597000006', 'Opel', 'Astra', 2009, '330 000', '1.6L I4', 'Petrol', 3, 'good', 'https://images.autouncle.com/pl/car\_images/4464f798/b34d-4838-86f0-0064c22f160&opel-astra-2009-rok-1-6-ben.jpg', '33 000.00', '234-678-901'),
(7, 'WAUZZZ8P2BA000007', 'Audi', 'A3', 2011, '155 000', '2.0L TDI', 'Diesel', 1, 'good', 'https://www.njuskalo.hr/image-w920x690/auti/audi-a3-1-6-tdi-tronic-sportback-slika-124455884.jpg', '55 000.00', '345-789-913'),
(8, 'WWAZZZ8P1C000008', 'Volkswagen', 'Golf', 2012, '263 000', '1.6L TDI', 'Diesel', 2, 'good', 'https://www.cnet.com/a/img/resize/9a7532b0126949961c2f72f9bf6e12b8ed4f914f/281/112-18273f4c-bb76-11e2-8a8e-0291187978f3/35165474\_OVR.jpg?auto=webp&fit=crop&height=675&width=1200', '63 000.00', '456-890'),
(9, 'WBA3A31000K000009', 'BMW', '3 Series', 2013, '531 500', '2.0L I4', 'Diesel', 1, 'very good', 'https://www.autocentrum.pl/dxp5bmEuSzkrVxReag5GLVlPOEIKfk-GVkhQiiUCDQRxtANR0LLB8aWkhkR1fHrgVXVYUVD5AU10ceIH0kcFTQ8oAkkrFuclDghGST5DBU8LdkRjPkMBTwEsGBEiCVsDQy8FA3oN', '51 500.00', '367-911-234'),
(10, 'VF1KZ178640000010', 'Renault', 'Clio', 2016, '223 000', '1.5L dci', 'Diesel', 1, 'good', 'https://www.njuskalo.hr/image-w920x690/auti/renault-clio-1-5-dci-2016-godina-navigacija-slika-105965988.jpg', '23 000.00', '478-112-345'),
(11, 'VF7SBRFB72000011', 'Citroën', 'C4', 2015, '164 000', '1.6L I4', 'Diesel', 2, 'very good', 'https://landinginteligente.com/fotos/CatalogoUsados/deconcesionarias--2023-6-15--13-15-15/8729cf1f-7355-4679-b9b1-e03fcf250db7.jpg', '64 000.00', '634-778-910'),
(12, 'W0L0XCE7597000012', 'Opel', 'Astra', 2010, '185 000', '1.4L I4', 'Petrol', 2, 'very good', 'https://image.ceneostatic.pl/data/products/150148361/p-opel-astra-2011-r-1-6-pb-115-km-bezwypadkowy.jpg', '85 000.00', '945-889-012'),
(13, 'WAUZZZ8P1B000013', 'Audi', 'A3', 2012, '335 000', '1.8L TFSI', 'Petrol', 1, 'very good', 'https://a.allegroimg.com/original/12e502/ba575572485aa@c85fc392a5f5f', '35 000.00', '156-990-123'),
(14, 'WVZZZ1KD0000014', 'Volkswagen', 'Golf', 2013, '115 000', '2.0L TDI', 'Diesel', 1, 'very good', 'https://a.allegroimg.com/s1024/0c5fad/0a97105248aab207d28bd2f25376', '15 000.00', '367-001-234'),
(15, 'WBA3A31000K000015', 'BMW', '3 Series', 2014, '135 000', '2.0L I4', 'Diesel', 1, 'good', 'https://s2.auto.drom.ru/photo/w4N-0VTOndX9KZByV0PP4NVQff7va9iu6Me3yN4Fdkkgy973pkBZkGrpujScsMRb6b92KgmP6XF-KLYDB4-fpNuNYjTs.jpg', '35 000.00', '478-112-345'),
(16, 'VF1KZ178640000016', 'Renault', 'Clio', 2017, '242 000', '1.0L Tce', 'Petrol', 1, 'good', 'https://img.chceauto.pl/renault-clio-kombi-3415-31155\_v2.jpg', '42 000.00', '534-788-918'),
(17, 'VF7SBRFB72000017', 'Citroën', 'C4', 2016, '341 000', '1.2L PureTech', 'Petrol', 2, 'very good', 'https://ireland.apollo.olcdn.com/v1/files/hhl2879piq51m-PL/image;s=4032x3024', '41 000.00', '345-799-012'),
(18, 'W0L0XCE7597000018', 'Opel', 'Astra', 2011, '185 000', '1.7L CDTI', 'Diesel', 2, 'very good', 'https://image.ceneostatic.pl/data/products/150148361/p-opel-astra-2011-r-1-6-pb-115-km-bezwypadkowy.jpg', '85 000.00', '356-800-123'),
(19, 'WAUZZZ8P1CA000019', 'Audi', 'A3', 2013, '123 000', '2.0L TDI', 'Diesel', 1, 'good', 'https://www.3auto.fr/composants/uploads/2023/04/audi-a3-sline-stronic-cockpit-avant.jpg', '23 000.00', '1434-678-910'),
(20, 'WVZZZ1KF0000020', 'Volkswagen', 'Golf', 2015, '93 000', '1.6L TDI', 'Diesel', 1, 'good', 'https://uzywane.renault.pl/CarExportImageManager.axd?ImageId=83868\_0', '93 000.00', '475-789-012'),
(21, 'WBA3A31000K000021', 'BMW', '3 Series', 2015, '134 000', '2.0L I4', 'Diesel', 1, 'very good', 'https://www.autocentrum.pl/dxp5bmEuSzkrVxReag5GLVlPOEIKfk-GVkhQiiUCDQRxtANR0LLB8aWkhkR1fHrgVXVYUVD5AU10ceIH0kcFTQ8oAkkrFuclDghGST5DBU8LdkRjPkMBTwEsGBEiCVsDQy8FA3oN', '34 000.00', '456-890-123'),
(22, 'VF1KZ178640000022', 'Renault', 'Clio', 2018, '56 000', '1.0L Tce', 'Petrol', 1, 'good', 'https://a.allegroimg.com/original/111532/6caf085341a9886a8853aba7d11/Renault-Clio-Renault-Clio-Salon-Polska-2018-1', '56 000.00', '237-981-234'),
(23, 'VF7SBRFB72000023', 'Citroën', 'C4', 2017, '65 000', '1.6L I4', 'Diesel', 2, 'good', 'https://upload.wikimedia.org/wikipedia/commons/f/fc/Citroen\_C4\_BlueHDi\_120\_Feel\_2017\_%2835385622361X29.jpg', '65 000.00', '588-012-345'),
(24, 'W0L0XCE7597000024', 'Opel', 'Astra', 2012, '123 000', '1.4L I4', 'Petrol', 2, 'good', 'https://www.auto-data.net/images/f107/Opel-Astra-J-facelift-2012.jpg', '23 000.00', '743-890-123'),
(25, 'WAUZZZ8P1DA000025', 'Audi', 'A3', 2014, '80 700', '1.6L TDI', 'Diesel', 1, 'good', 'https://i.ytimg.com/vi/OutX0lpbzRs/maxresdefault.jpg', '45 700.00', '232-789-012'),
(26, 'WVZZZ1L9D0000026', 'Audi', 'Q5', 2019, '152 890', '2.0L TFSI', 'Petrol', 1, 'good', 'https://a.allegroimg.com/s1024/0cbc8/8444b5dc4f38872d1fb13bead66', '52 890.00', '444-890-123'),
(27, 'WDCGG6B9F6G0000027', 'Mercedes-Benz', 'GLK', 2018, '290 400', '2.1L I4', 'Diesel', 1, 'good', 'https://ireland.apollo.olcdn.com/v1/files/eyJmbiI61jpoYTBuOWJid3FsjdEt1RTPTU9UT1BMiiwidyi6W3siZm4i0iJ3ZzRnbFnkxZi1PVE9NT1RPUEwiLCjzIjoiMTyAsLTewIiwiYSI6IjAiFV19.GV7zqEukHWFqda5XBpFoaeGzHRTPB', '200 000.00', '456-890-123'),
(28, 'WVZZZ5NLV0000028', 'Volkswagen', 'Tiguan', 2020, '64 000', '1.5L TSI', 'Petrol', 1, 'good', 'https://toptun.pl/userdata/public/gfx/461043/VOLKSWAGEN-TIGUAN-II-AD1-R-LINE-2020--PO-FACELIFTINGU-CHLAPACZE.jpg', '64 000.00', '123-012-345'),
(29, 'WBAWZT10200000004', 'BMW', 'X3', 2019, '90 000', '2.0L I4', 'Diesel', 1, 'good', 'https://bmw-uzywane.com.pl/assets/photo/upload/cars/33155/vehicle\_85ac0-crop-1200-800.jpg', '90 000.00', '312-678-910'),
(30, 'WA1VAAF79KD000005', 'Audi', 'Q7', 2019, '92 000', '3.0L V6', 'Diesel', 1, 'good', 'https://getrentacar.com/storage/cache/images/960-640-100-fit-191403.jpeg', '92 000.00', '345-789-012'),
(31, 'SALY2A8N4KA000006', 'Land Rover', 'Discovery', 2020, '192 000', '2.0L I4', 'Diesel', 1, 'very good', 'https://www.cnet.com/a/img/resize/c238ee418e4a57b7a02a5d05f4d64b8b96fe05c0/hub/2020/04/24/0e8ec652-19af-4f49-930c-8f0208ec94fb/2020-discovery-30.jpg?auto=webp&width=768', '102 000.00', '743-890-123'),
(32, 'SKYKTD2A7FG000007', 'Kia', 'Sorento', 2019, '135 000', '2.2L I4', 'Diesel', 1, 'very good', 'https://smartcdn.gprod.postmedia.digital/driving/wp-content/uploads/2019/02/2019-kia-sorento-sx-17.jpg', '105 000.00', '900-981-234'),
(33, 'WDC0G4KB2KF000008', 'Mercedes-Benz', 'GLC', 2019, '102 000', '2.0L I4', 'Diesel', 1, 'very good', 'https://ocdn.eu/pulscms-transforms/1/V7tk9pTUR8y8zYmJn1DliYjvDzTjIyMzYxDeNy5qCeS1QazKhCcDNBxyT8c0EsM0Cnt4AaAeWaQ', '132 000.00', '091-012-345'),
(34, 'WVZZZ5NLV0000009', 'Volkswagen', 'Tiguan', 2021, '19 000', '2.0L TSI', 'Petrol', 1, 'very good', 'https://avatars.mds.yandex.net/get-autor-uos/2158063/60725884e9c8d95e7d1c7f73e83189/456x342', '819 000.00', '921-678-910'),
(35, 'SUWZT2C54K9D000010', 'BMW', 'X5', 2019, '13 000', '3.0L I6', 'Diesel', 1, 'good', 'https://bmw-uzywane.com.pl/assets/photo/upload/cars/34763/vehicle\_84909-scale-1200-0.jpg', '103 000.00', '532-789-012'),
(36, 'SALW2R2V6KA000011', 'Land Rover', 'Range Rover Sport', 2019, '19 000', '3.0L V6', 'Diesel', 1, 'good', 'https://lh3.googleusercontent.com/proxy=0pkkJug2sDFMcYJrrd7s585ExTRFcB9zDvNgtdakBjfcu39udzNhignknEmb7jtilaeA47jdLfhnh3N2kRraTJAHHMyfNLFQKQdK8nhd5\_gIkYHmW5cmFC4', '193 000.00', '456-111-123'),
(37, 'JTMYFREV9JD000012', 'Toyota', 'RAV4', 2019, '15 000', '2.5L I4', 'Petrol', 1
```

# Działanie aplikacji:

The screenshot shows a web browser window with a tab titled "Car Search". The main content is a "Car Search" application displaying three car listings in a grid format.

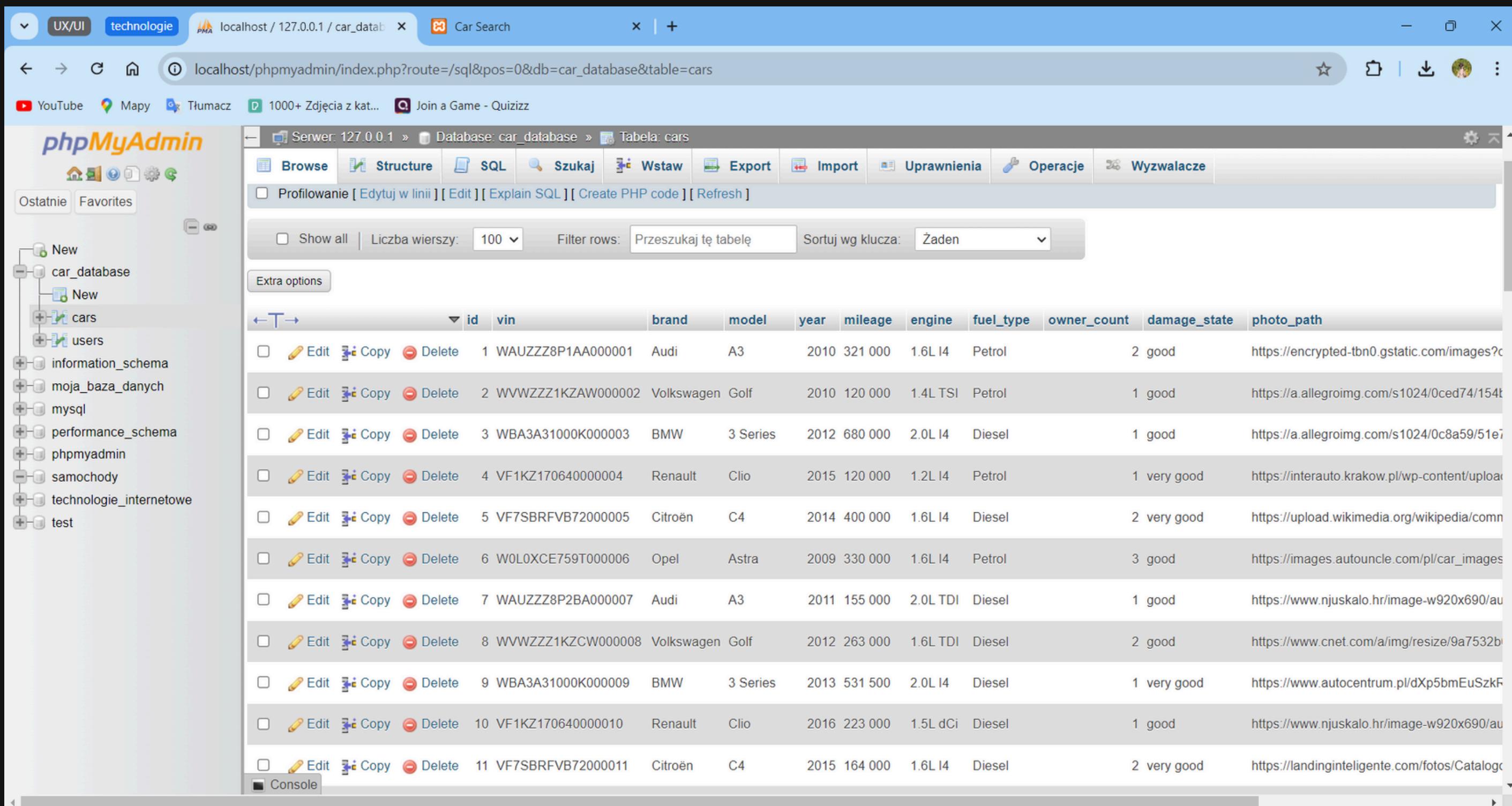
**Audi A3**  
35 000.00 PLN  
Petrol | 321 000 km | 2010 | good  
789-123-456

**Volkswagen Golf**  
25 000.00 PLN  
Petrol | 120 000 km | 2010 | good

**BMW 3 Series**  
23 000.00 PLN  
Diesel | 680 000 km | 2012 | good

A red arrow points from the top left towards the Audi A3 listing.

# Działanie aplikacji:



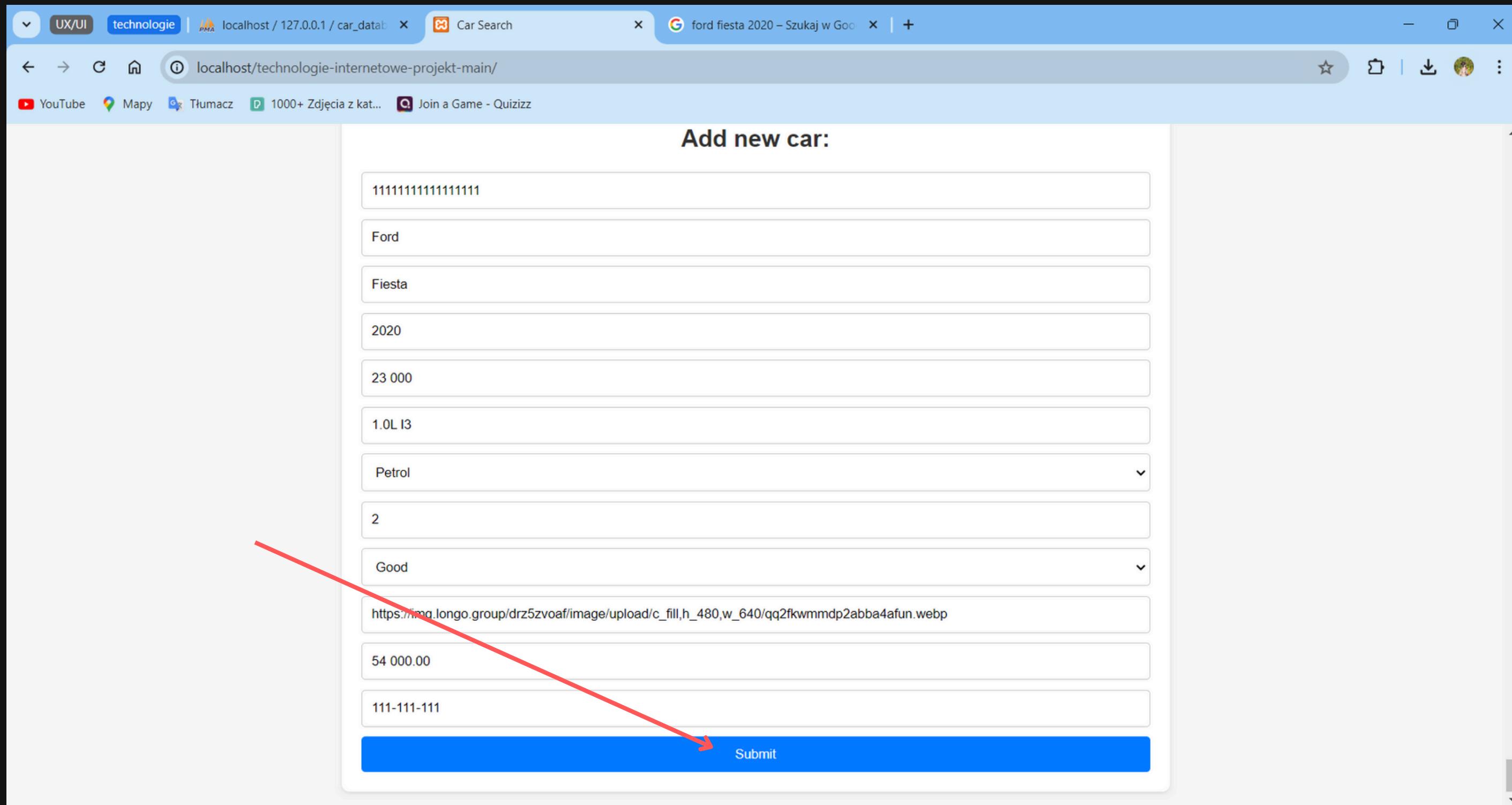
# Działanie aplikacji:

The screenshot shows a web browser window with a light blue header bar. The address bar displays the URL `localhost/technologie-internetowe-projekt-main/`. Below the address bar, there are several browser tabs and icons. The main content area of the browser shows a form titled "Add new car:" with various input fields and dropdown menus. The form fields include:

- Vin number
- Brand
- Model
- Year
- Mileage
- Engine
- Fuel type
- Owner count
- Damage state
- Link to photo
- Price
- Phone number

At the bottom of the form is a large blue "Submit" button.

# Działanie aplikacji:



# Działanie aplikacji:

The screenshot shows the phpMyAdmin interface for a database named 'car\_database'. The left sidebar lists various databases and tables, with a red arrow pointing from the top-left towards the delete icon of the last row in the 'cars' table. The 'cars' table has columns: id, vin, brand, model, year, mileage, engine, fuel\_type, owner\_count, damage\_state, and photo\_path. The data includes entries for Tesla Model 3, Ford Bronco, Toyota Highlander, Chevrolet Malibu, Chevrolet Silverado, Ford Edge, Jeep Cherokee, Ford Explorer, Nissan Frontier, and a Ford Fiesta.

	id	vin	brand	model	year	mileage	engine	fuel_type	owner_count	damage_state	photo_path
	82	5YJ3E1EA8MF600007	Tesla	Model 3	2021	13 000	electric	Electric	1	very good	<a href="https://imageio.forbes.com/specials-images/...">https://imageio.forbes.com/specials-images/...</a>
	83	1FMEE5DH9ML700008	Ford	Bronco	2022	23 000	2.3L I4	Petrol	1	very good	<a href="https://ocdn.eu/pulscms-transforms/1/gfcktk...">https://ocdn.eu/pulscms-transforms/1/gfcktk...</a>
	84	5TDYZRAH5MS800009	Toyota	Highlander	2021	12 000	3.5L V6	Petrol	1	very good	<a href="https://pewneauto.pl/media-thumbnails/5c/5c...">https://pewneauto.pl/media-thumbnails/5c/5c...</a>
	85	1G1ZE5ST0MF900010	Chevrolet	Malibu	2021	89 000	1.5L I4	Petrol	1	good	<a href="https://vehicle-images.dealerinspire.com/6d...">https://vehicle-images.dealerinspire.com/6d...</a>
	86	3GCUDHEL9MG100011	Chevrolet	Silverado	2021	3 000	6.2L V8	Petrol	1	good	<a href="https://images.arkonoffroad.com/web-compr...">https://images.arkonoffroad.com/web-compr...</a>
	87	2FMPK4K96MBA00012	Ford	Edge	2021	3 000	2.0L I4	Petrol	1	good	<a href="https://autogaleria.pl/img/1080/607/fit/conter...">https://autogaleria.pl/img/1080/607/fit/conter...</a>
	88	1C4PJMDX0KD100013	Jeep	Cherokee	2021	99 999	3.2L V6	Petrol	1	very good	<a href="https://di-uploads-pod11.dealerinspire.com/s...">https://di-uploads-pod11.dealerinspire.com/s...</a>
	89	1FM5K8D89LGA00014	Ford	Explorer	2021	2 300	2.3L I4	Petrol	1	good	<a href="https://ireland.apollo.olcdn.com/v1/files/eyJ...">https://ireland.apollo.olcdn.com/v1/files/eyJ...</a>
	90	1N6ED1EK1NN200015	Nissan	Frontier	2022	1 000	3.8L V6	Petrol	1	good	<a href="https://www.cnet.com/a/img/resize/3f48cdb5...">https://www.cnet.com/a/img/resize/3f48cdb5...</a>
	91	11111111111111111111	Ford	Fiesta	2020	23 000	1.0L I3	Petrol	2	Good	<a href="https://img.longo.group/drz5zvoaf/image/upl...">https://img.longo.group/drz5zvoaf/image/upl...</a>

# Działanie aplikacji:

The screenshot shows a web browser displaying a car search application. The page has a header with tabs for 'UX/UI', 'technologie', and 'localhost / 127.0.0.1 / car\_database'. The main content area shows three car listings in cards:

- Ford Explorer**: Price 322 300.00 PLN. Features: Petrol, 2.3L I4, 2 300 km, 2021, good. Buttons: Call, Delete.
- Nissan Frontier**: Price 350 000.00 PLN. Features: Petrol, 3.8L V6, 1 000 km, 2022, good. Buttons: Call, Delete.
- Ford Fiesta**: Price 54 000.00 PLN. Features: Petrol, 1.0L I3, 23 000 km, 2020, Good. Buttons: Call, Delete. A red arrow points from the Nissan Frontier card to the Ford Fiesta card's delete button.

At the bottom, there is a section for adding a new car with a text input field labeled 'Vin number'.

# Działanie aplikacji:

The screenshot shows the phpMyAdmin interface for a database named 'car\_database'. The 'cars' table is selected, displaying 10 rows of data. The columns are: id, vin, brand, model, year, mileage, engine, fuel\_type, owner\_count, damage\_state, and photo\_path. A red arrow points to the 'Delete' link for the last row (id 90).

	id	vin	brand	model	year	mileage	engine	fuel_type	owner_count	damage_state	photo_path
	82	5YJ3E1EA8MF600007	Tesla	Model 3	2021	13 000	electric	Electric	1	very good	<a href="https://imageio.forbes.com/specials-images/...">https://imageio.forbes.com/specials-images/...</a>
	83	1FMEE5DH9ML700008	Ford	Bronco	2022	23 000	2.3L I4	Petrol	1	very good	<a href="https://ocdn.eu/pulscms-transforms/1/gfcktk...">https://ocdn.eu/pulscms-transforms/1/gfcktk...</a>
	84	5TDYZRAH5MS800009	Toyota	Highlander	2021	12 000	3.5L V6	Petrol	1	very good	<a href="https://pewneauto.pl/media/thumbnails/5c/50...">https://pewneauto.pl/media/thumbnails/5c/50...</a>
	85	1G1ZE5ST0MF900010	Chevrolet	Malibu	2021	89 000	1.5L I4	Petrol	1	good	<a href="https://vehicle-images.dealerinspire.com/6d...">https://vehicle-images.dealerinspire.com/6d...</a>
	86	3GCUDHEL9MG100011	Chevrolet	Silverado	2021	3 000	6.2L V8	Petrol	1	good	<a href="https://images.arkonoffroad.com/web-compr...">https://images.arkonoffroad.com/web-compr...</a>
	87	2FMPK4K96MBA00012	Ford	Edge	2021	3 000	2.0L I4	Petrol	1	good	<a href="https://autogaleria.pl/img/1080/607/fit/conter...">https://autogaleria.pl/img/1080/607/fit/conter...</a>
	88	1C4PJMDX0KD100013	Jeep	Cherokee	2021	99 999	3.2L V6	Petrol	1	very good	<a href="https://di-uploads-pod11.dealerinspire.com/s...">https://di-uploads-pod11.dealerinspire.com/s...</a>
	89	1FM5K8D89LGA00014	Ford	Explorer	2021	2 300	2.3L I4	Petrol	1	good	<a href="https://ireland.apollo.olcdn.com/v1/files/ey...">https://ireland.apollo.olcdn.com/v1/files/ey...</a>
	90	1N6ED1EK1NN200015	Nissan	Frontier	2022	1 000	3.8L V6	Petrol	1	good	<a href="https://www.cnet.com/a/img/resize/3f48cdb5...">https://www.cnet.com/a/img/resize/3f48cdb5...</a>

**DZIĘKUJEMY**

**ZA UWAGĘ**