

PHP: Wykres temperatury Polski z regresją liniową

Wykres można obejrzeć tutaj: https://wiki.ostrowski.net.pl/php_mysql/pol_temp.php

Ten artykuł przedstawia przykład programu w PHP, który pobiera dane z bazy danych MySQL, a następnie za pomocą biblioteki JavaScript **Chart.js** wyświetla wykres liniowy średnich temperatur w Polsce wraz z prostą linią trendu (regresją liniową).

Dane wejściowe

Program korzysta z bazy danych MySQL o nazwie `polandtemperature`, która zawiera tabelę `temp` z następującymi kolumnami:

- `ID` – identyfikator wiersza,
- `Date` – data pomiaru,
- `Temp` – wartość temperatury.

Połączenie z bazą danych

Połączenie z bazą danych realizowane jest za pomocą PDO:

```
$conn = new  
PDO("mysql:host=localhost;dbname=polandtemperature;charset=utf8mb4",  
"viewer", "viewer");
```

Jeśli połączenie się nie powiedzie, program zakończy działanie z komunikatem błędu.

Pobieranie i przetwarzanie danych

Program wykonuje zapytanie SQL:

```
SELECT * FROM temp ORDER BY ID;
```

Wyniki zapisywane są do tablicy PHP. Z kolumn `Date` i `Temp` wyodrębniane są osobne tablice:

```
$labels = array_column($Data, 'Date');  
$temps = array_column($Data, 'Temp');
```

Obliczanie regresji liniowej

W celu dodania trendu liniowego, wykonywane są obliczenia regresji liniowej metodą najmniejszych kwadratów:

```
$slope = ($n * $sum_xy - $sum_x * $sum_y) / ($n * $sum_x2 - $sum_x ** 2);  
$intercept = ($sum_y - $slope * $sum_x) / $n;
```

Następnie generowana jest druga tablica zawierająca dane dla linii trendu:

```
$trendLine = array_map(fn($x) => round($slope * $x + $intercept, 2),  
$x_vals);
```

Wyświetlanie wykresu za pomocą Chart.js

W HTML wyświetlany jest wykres z dwiema seriami danych:

- rzeczywiste dane temperatur (`Poland Average Temperature`) – czerwona linia,
- linia trendu (`Linear Trend Line`) – przerywana niebieska linia.

```
datasets: [  
  {  
    label: 'Poland Average Temperature',  
    data: [...],  
    borderColor: 'rgba(255, 99, 132, 1)'  
  },  
  {  
    label: 'Linear Trend Line',  
    data: [...],  
    borderColor: 'rgba(54, 162, 235, 1)',  
    borderDash: [5, 5]  
  }  
]
```

Biblioteka **Chart.js** generuje responsywny wykres, który można osadzić na stronie WWW.

Efekt końcowy

Użytkownik widzi liniowy wykres temperatur wraz z prostą, która pokazuje ogólny trend (np. ocieplanie się klimatu lub spadki temperatur). Trend ułatwia interpretację danych historycznych.

Podsumowanie

Ten program demonstruje:

- Jak pobrać dane z MySQL w PHP,
- Jak obliczyć regresję liniową,
- Jak wykorzystać Chart.js do wizualizacji danych i trendów.

Dzięki temu rozwiązaniu możemy łatwo tworzyć dynamiczne, interaktywne wykresy statystyczne w aplikacjach webowych.

Matematyka: regresja liniowa

W celu wyznaczenia prostoliniowego trendu danych (tzw. regresji liniowej) stosujemy model matematyczny:

$$y = a \cdot x + b$$

Gdzie:

- x – niezależna zmienna (czas, indeks daty),
- y – zmienna zależna (temperatura),
- a – współczynnik kierunkowy (nachylenie),
- b – wyraz wolny (punkt przecięcia z osią Y).

Aby obliczyć a i b , używamy poniższych wzorów:

$$a = \frac{n \sum x_i y_i - \sum x_i \sum y_i}{n \sum x_i^2 - (\sum x_i)^2}$$

$$b = \frac{\sum y_i - a \sum x_i}{n}$$

Gdzie:

- n – liczba punktów danych,
- $\sum x_i$ – suma wszystkich wartości x (np. indeksów dat),
- $\sum y_i$ – suma wszystkich temperatur,
- $\sum x_i y_i$ – suma iloczynów $x_i \cdot y_i$,
- $\sum x_i^2$ – suma kwadratów wartości x_i .

Następnie dla każdego punktu x_i obliczamy odpowiadające y_i na linii trendu:

$$\hat{y}_i = a \cdot x_i + b$$

Wartości $\hat{y}_i$ tworzą prostą linię trendu, którą dodajemy do wykresu obok rzeczywistych danych.

Dzięki temu możemy łatwo zauważyć, czy dane wykazują tendencję rosnącą, malejącą, czy są stabilne w czasie.

Kod

[pol_temp.php](#)

```
<?php
// Database connection settings
$serverName = "localhost";
$database = "polandtemperature";
$username = "";
$password = "";

// Connect using PDO for MySQL
try {
    $conn = new
PDO("mysql:host=$serverName;dbname=$database;charset=utf8mb4",
$username, $password);
    $conn->setAttribute(PDO::ATTR_ERRMODE, PDO::ERRMODE_EXCEPTION);
} catch (PDOException $e) {
    die("Connection failed: " . $e->getMessage());
}

// Fetch data
$data = [];
$stmt = $conn->query("SELECT * FROM temp ORDER BY ID;");
while ($row = $stmt->fetch(PDO::FETCH_ASSOC)) {
    $data[] = $row;
}

// Close DB connection
$conn = null;

// Extract columns
$labels = array_column($data, 'Date');
$temps = array_column($data, 'Temp');

// Convert dates to numeric values (e.g. index) for regression
$x_vals = range(0, count($temps) - 1);
$y_vals = $temps;

// Linear regression calculation ( $y = a * x + b$ )
$n = count($x_vals);
$sum_x = array_sum($x_vals);
$sum_y = array_sum($y_vals);
$sum_xy = array_sum(array_map(fn($x, $y) => $x * $y, $x_vals,
$y_vals));
$sum_x2 = array_sum(array_map(fn($x) => $x * $x, $x_vals));

$slope = ($n * $sum_xy - $sum_x * $sum_y) / ($n * $sum_x2 - $sum_x **
2);
$intercept = ($sum_y - $slope * $sum_x) / $n;

// Generate trend line data
$trendLine = array_map(fn($x) => round($slope * $x + $intercept, 2),
$x_vals);
?>
```

```
<!DOCTYPE html>
<html>
<head>
  <title>Temperature Chart with Trend Line</title>
  <script src="https://cdn.jsdelivr.net/npm/chart.js"></script>
</head>
<body>
  <h2>Temperature Trend: Poland Average</h2>
  <canvas id="tempChart" width="800" height="400"></canvas>

  <script>
    const ctx =
document.getElementById('tempChart').getContext('2d');
    const tempChart = new Chart(ctx, {
      type: 'line',
      data: {
        labels: <?= json_encode($labels) ?>,
        datasets: [
          {
            label: 'Poland Average Temperature',
            data: <?= json_encode($temps) ?>,
            borderColor: 'rgba(255, 99, 132, 1)',
            fill: false,
            tension: 0.1
          },
          {
            label: 'Linear Trend Line',
            data: <?= json_encode($trendLine) ?>,
            borderColor: 'rgba(54, 162, 235, 1)',
            borderDash: [5, 5],
            fill: false,
            pointRadius: 0,
            tension: 0
          }
        ]
      },
      options: {
        responsive: true,
        scales: {
          y: {
            beginAtZero: false,
            title: {
              display: true,
              text: 'Temperature (°C)'
            }
          },
          x: {
            title: {
              display: true,
              text: 'Date'
            }
          }
        }
      }
    });
  </script>
</body>
</html>
```

```
        }  
    }  
});  
</script>  
</body>  
</html>
```

From:

<http://wiki.ostrowski.net.pl/> - **Kacper's Wiki**

Permanent link:

http://wiki.ostrowski.net.pl/doku.php?id=narzedzia:php_global_warming&rev=1747515377

Last update: **2025/05/17 22:56**