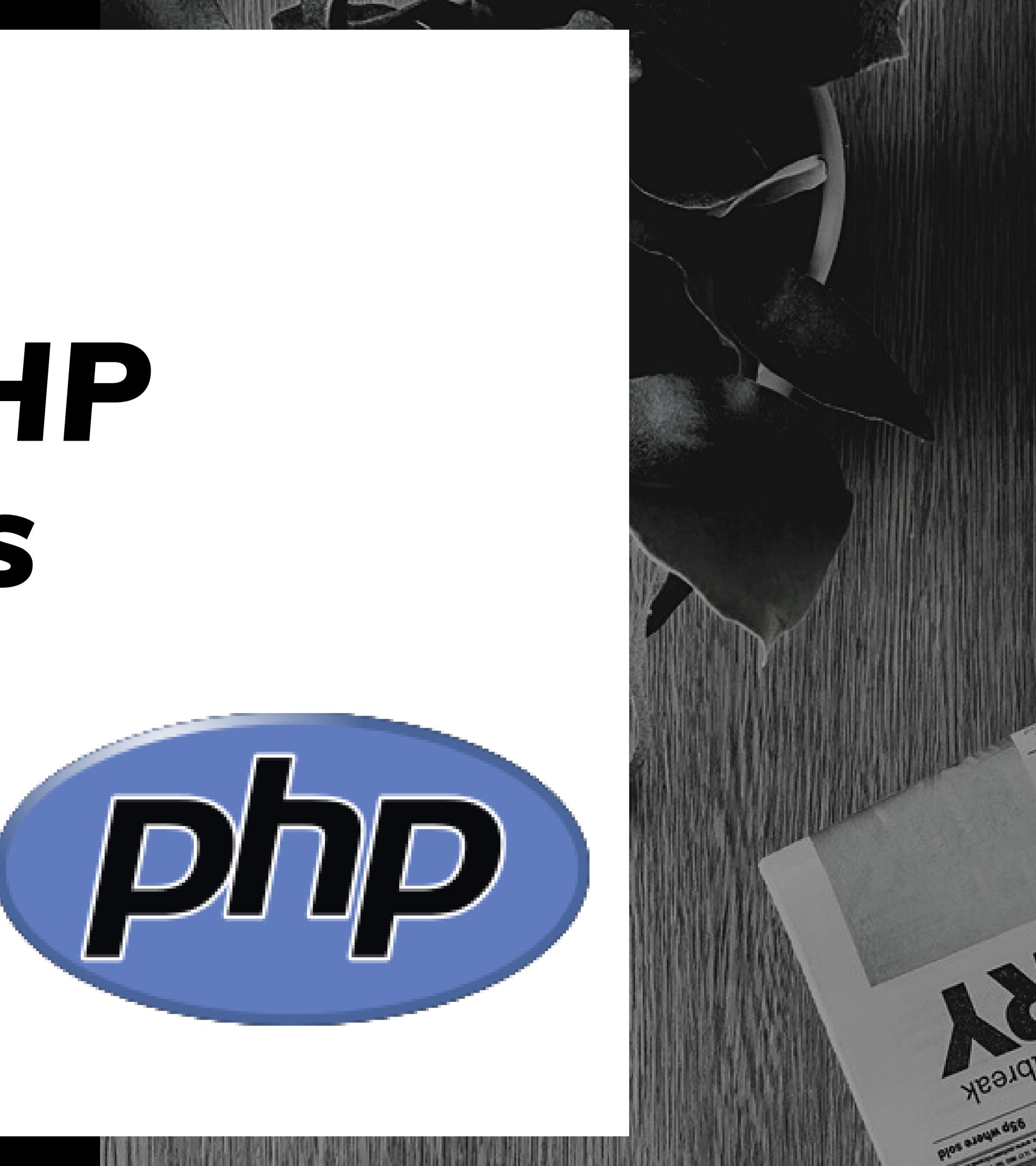


Szczepan Dworak
Grzegorz Bąk

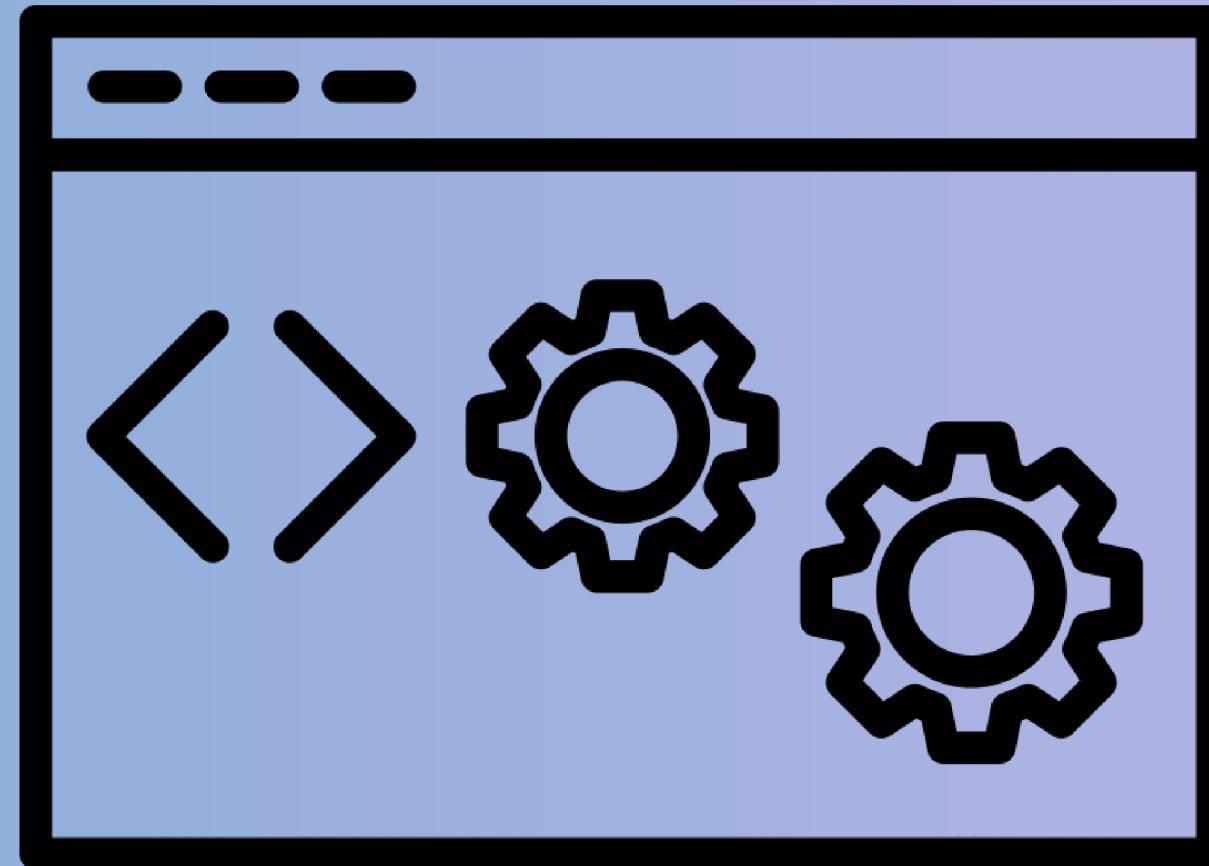
Backend: PHP oraz Node.js





Backend: Czym tak właściwie jest i historczne początki PHP

Korzenie współczesnego wyglądu aplikacji webowych.



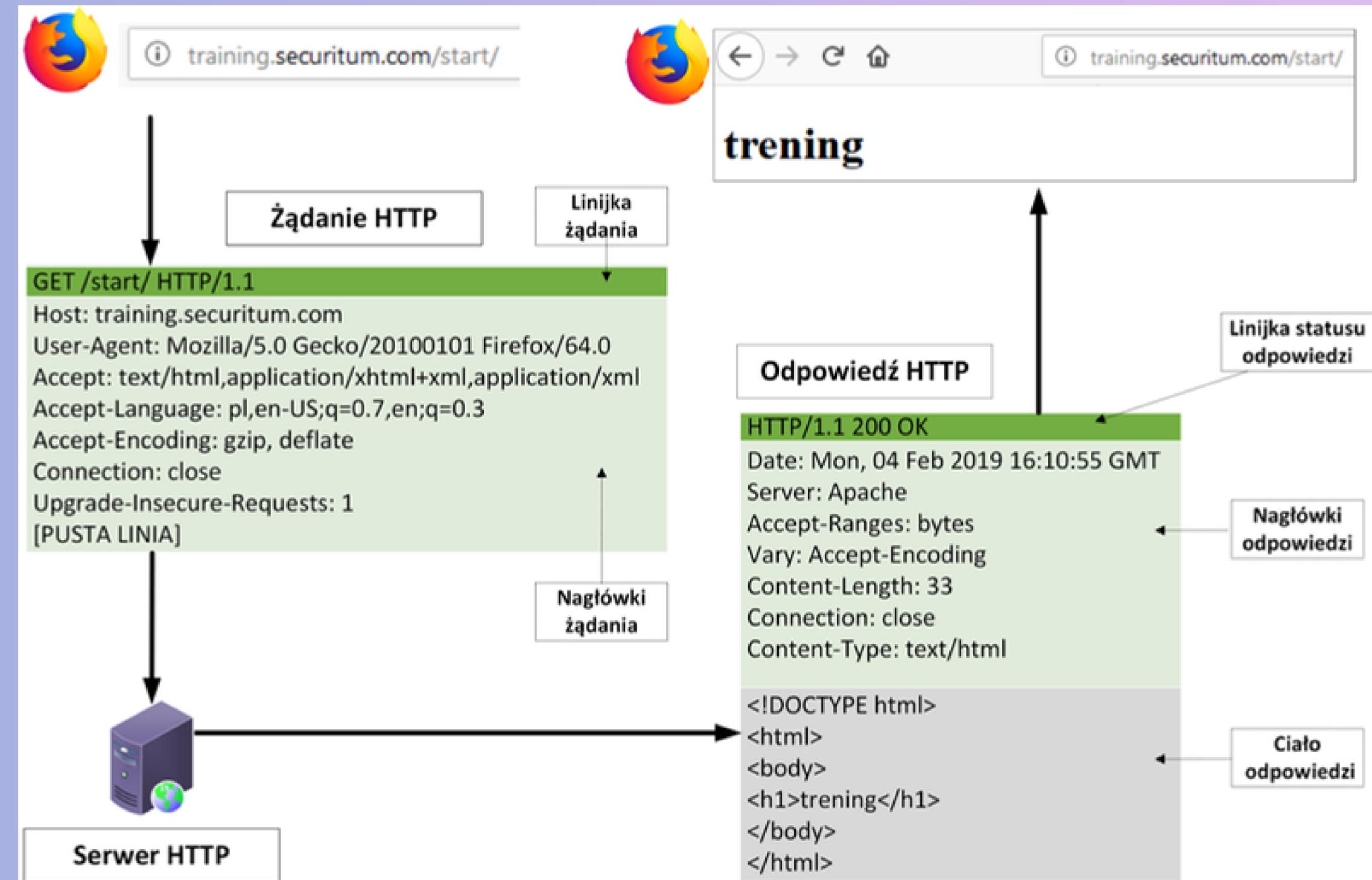
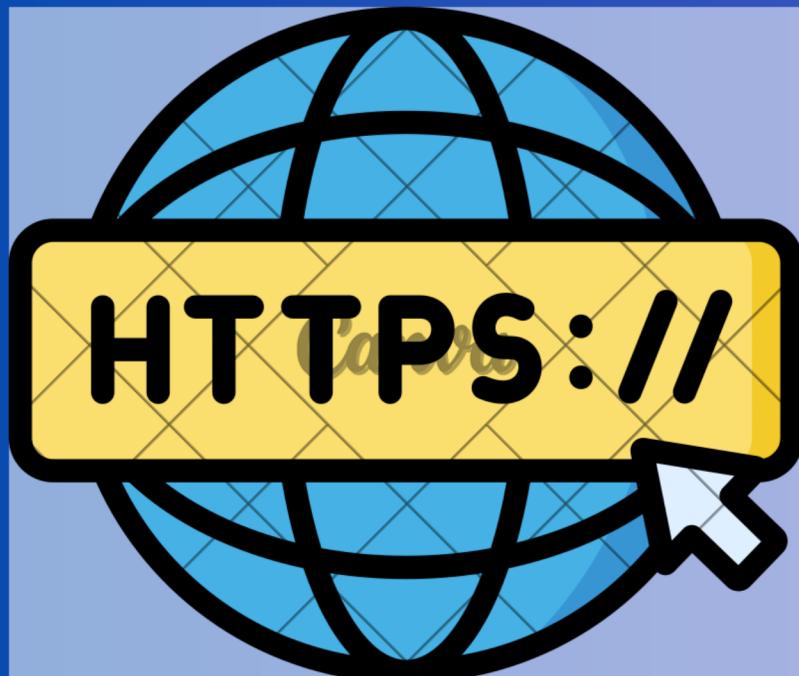
BACKEND

BACK-ENDEM NAZYWAMY TO, CO ZNAJDUJE SIĘ NA SERWERZE, A DO CZEGO UŻYTKOWNIK NIE MA BEZPOŚREDNIEGO DOSTĘPU. BACK-END UKRYTY JEST ZA FRONT-ENDEM I ZARZĄDZA CAŁYM SYSTEMEM DANEJ APLIKACJI LUB STRONY WWW.

JAKO GŁÓWNE ZADANIE BACKENDU MOŻNA UZNAĆ DOSTARCZANIE TREŚCI DLA UŻYTKOWNIKA. Z POZIOMU BAKCENDU UZYSKUJEMY DOSTĘP DO BAZ DANYCH ORAZ INNYCH USŁUG.

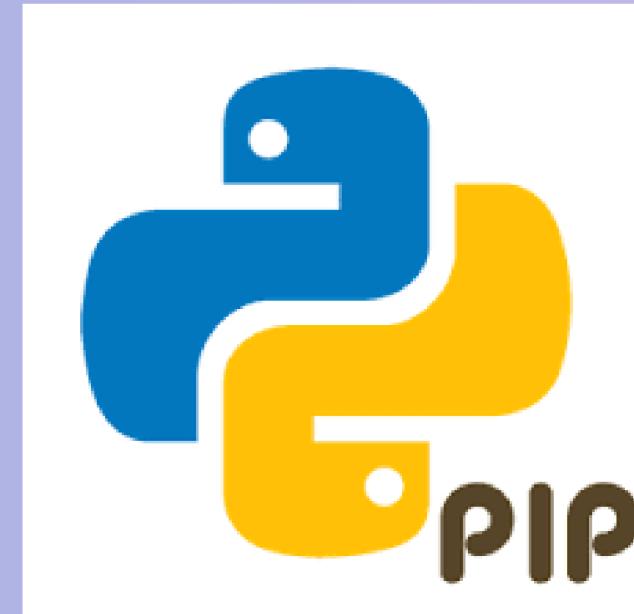
PRZESYŁ ŻĄDAŃ O DANE TREŚCI I ODPOWIEDZI Z DANYMI ODBYWA SIĘ PRZY POMOCY PROTOKOŁU HTTP/HTTPS W TZW. REQUEST-RESPONSE CYCLE.

CIEKAWOSTA: ABY WYSWIETLIĆ STRONĘ AMAZONA PRZEGŁĄDARKA WYKONUJE OKOŁO 300 ŻĄDAŃ HTTP.





django



Apache
MavenTM





- **1994:** POCZĄTEK JAKO "PERSONAL HOME PAGE TOOLS" PRZEZ RASMUSA LERDORFA.
- **1995-1998:** EWOLUCJA DO PHP/FI 2.0 I PHP 3 - "PHP: HYPERTEXT PREPROCESSOR".
- **2000:** PHP 4 Z ZEND ENGINE - POPRAWA WYDAJNOŚCI.
- **2004:** PHP 5 - WSPARCIE OOP, NOWY ZEND ENGINE II.
- **2015:** PHP 7 - SKOK WYDAJNOŚCI, NOWE FUNKCJE.
- **DZISIAJ:** SZEROKO UŻYWANY, CIĄGŁA EWOLUCJA, BOGATY EKOSYSTEM.

```
<!--include /text/header.html-->

<!--getenv HTTP_USER_AGENT-->
<!--if substr $exec_result Mozilla-->
    Hey, you are using Netscape!<p>
<!--endif-->

<!--sql database select * from table where user='\$username'-->
<!--ifless $numentries 1-->
    Sorry, that record does not exist<p>
<!--endif exit-->
    Welcome <!--$user-->!<p>
    You have <!--$index:0--> credits left in your account.<p>

<!--include /text/footer.html-->
```



PHP Przegląd

STATYSTYKI



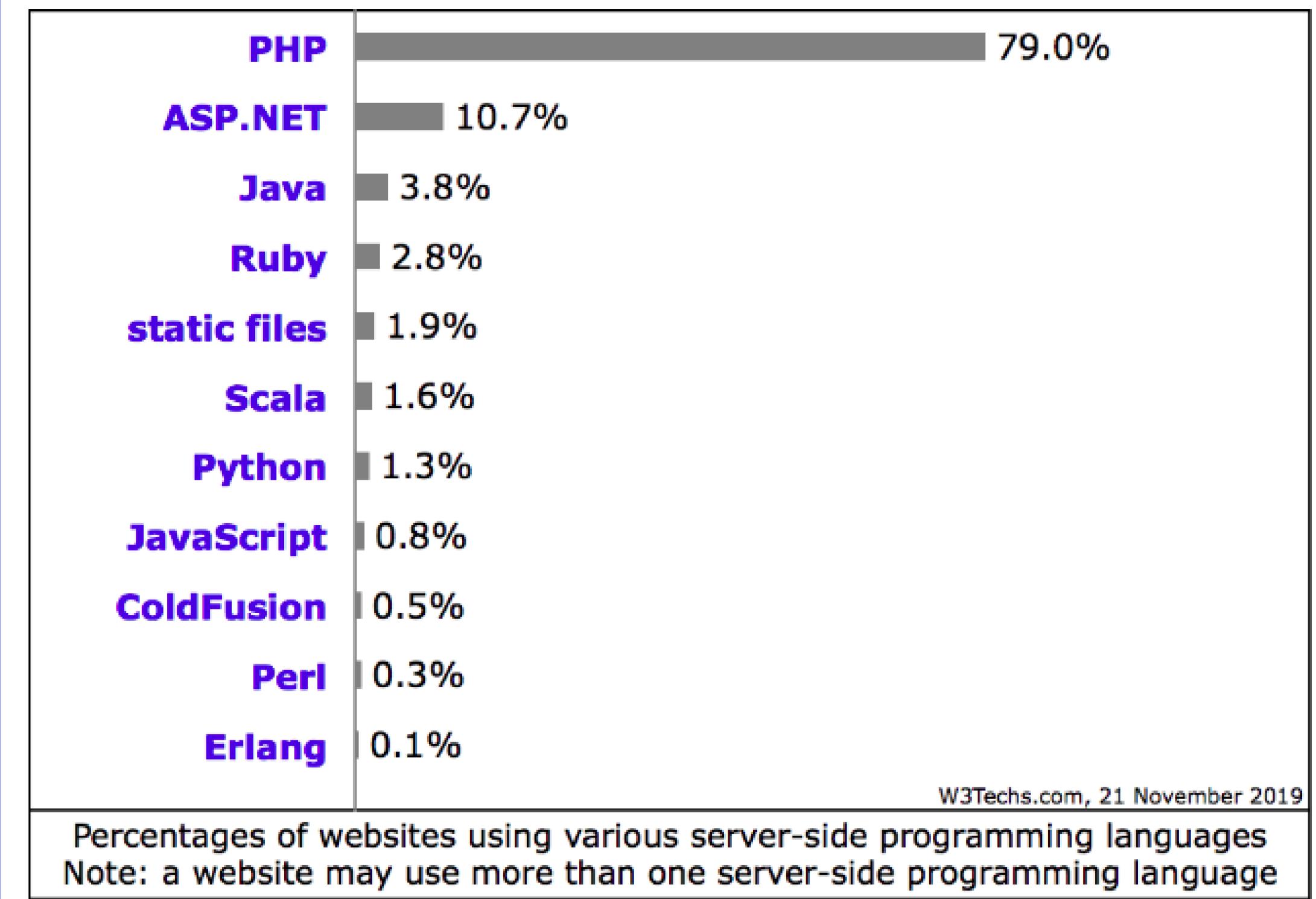
ZASTOSOWANIA



MOCNE I SŁABE
STRONY



Dobrze się
trzyma
staruszek





- **ELASTYCZNOŚĆ I INTEGRACJA:**
UMOŻLIWIĘ ŁATWĄ INTEGRACJĘ Z
RÓŻNYMI BAZAMI DANYCH I SYSTEMAMI.
- **CMS:** PODSTAWA DLA WORDPRESS,
DRUPAL, JOOMLA - UŁATWIA
ZARZĄDZANIE TREŚCIĄ.
- **E-COMMERCE:** MAGENTO, WOOCOMMERCE
- NAPĘDZA SKLEPY INTERNETOWE.
- **APLIKACJE WEBOWE:** ROZWÓJ
ZŁOŻONYCH SYSTEMÓW Z LARAVEL,
SYMFONY.
- **SPOŁECZNOŚCI ONLINE:** FORA TAKIE JAK
PHPBB.

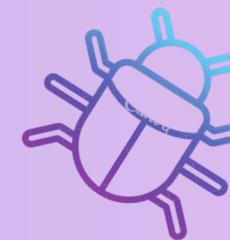


WPROWADZENIE DO SKŁADNI PHP

PORÓWNANIE JĘZYKA PHP Z INNYMI JĘZYKAMI: PLUSY I MINUSY



- **ZMIENNE:** PHP UŻYWA \$, DYNAMICZNE TYPOWANIE VS. STATYCZNE/LEKKO DYNAMICZNE TYPOWANIE W JAVA/JAVASCRIPT/PYTHON.
- **KONKATENACJA:** PHP (.), JAVASCRIPT/PYTHON (+).
- **DEFINICJA FUNKCJI:** PHP FUNCTION, PYTHON DEF Z SELF, JAVASCRIPT SKRÓCONA SKŁADNIA W KLASACH.
- **OBSŁUGA BŁĘDÓW:** PHP, PYTHON, JAVASCRIPT UŻYWAJĄ TRY I CATCH/EXCEPT, ALE RÓŻNIĄ SIĘ PODEJŚCIEM DO OBSŁUGI



PHP

```
<!DOCTYPE html>
<html>
<body>

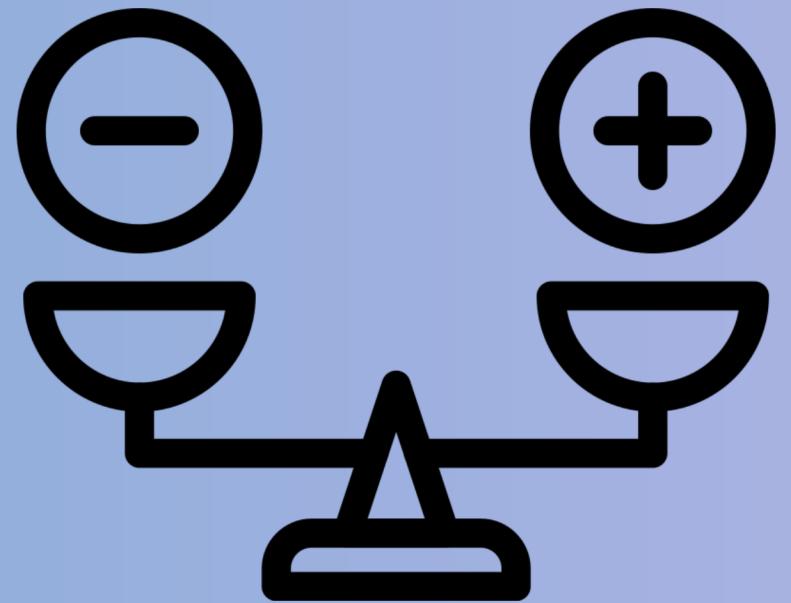
<?php
$color = "red";
echo "My car is " . $color . "<br>";
echo "My house is " . $COLOR . "<br>";
echo "My boat is " . $coL0R . "<br>";
?>

</body>
</html>
```

JS

```
// How to create variables:
var x;
let y;

// How to use variables:
x = 5;
y = 6;
let z = x + y;
```



ZALETY:

- *POPULARNOŚĆ I DOSTĘPNOŚĆ*
- *WSPARCIE DLA WIELU BAZ DANYCH*
- *PROSTOTA*
- *DYNAMICZNOŚĆ*



WADY:

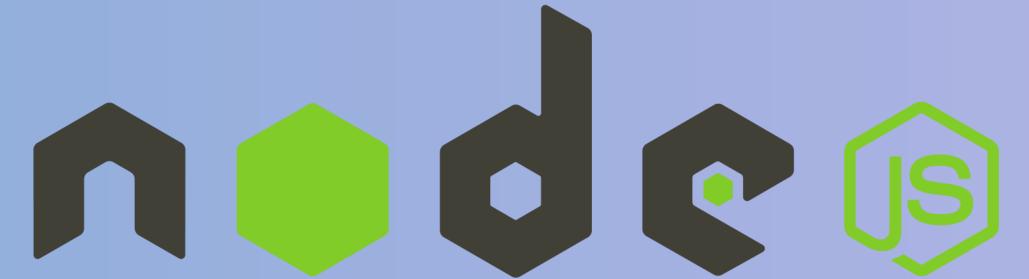
- *HISTORIA BEZPIECZEŃSTWA*
- *ZARZĄDZANIE TYPAMI DANYCH*
- *SKŁADNIA*
- *DYNAMICZNOŚĆ*
- *BRAK WSPARCIA DLA ASYNCHRONICZNOŚCI*

PHP, Node.JS oraz inne koncepty na Backend

CZYM JEST NODE.JS?

**PLUSY I MINUSY
UŻYCIA**

**PRZEWAGI/BRAKI
WZGLĘDEM INNYCH
TECHNOLOGII**

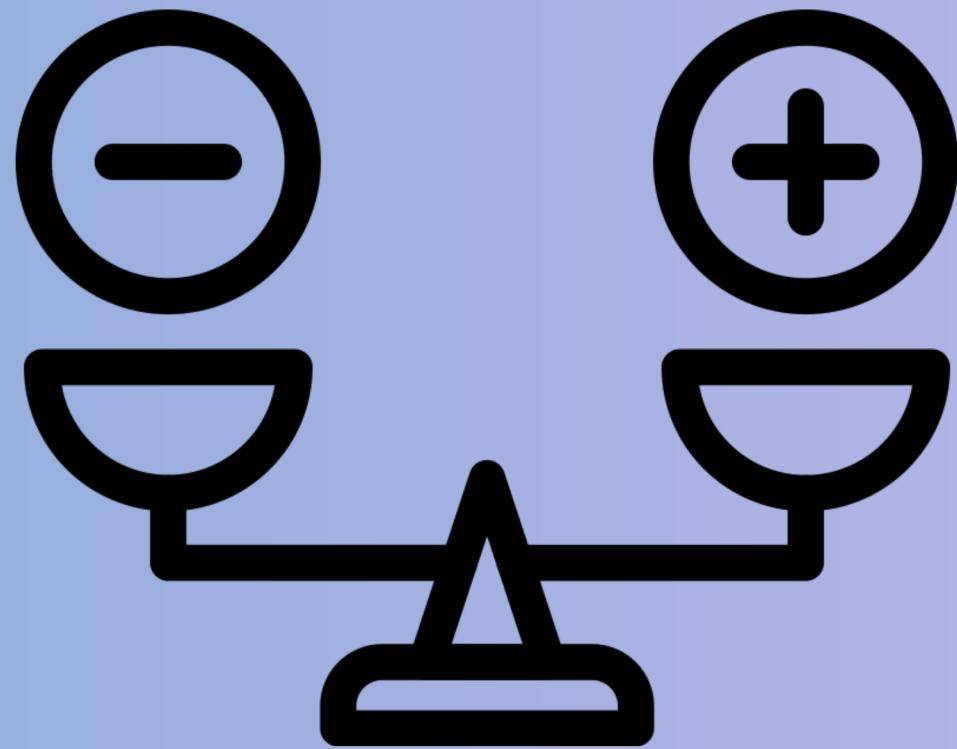


**NODE.JS TO MIĘDZYPLATFORMOWE
ŚRODOWISKO URUCHOMIENIOWE
POZWALAJĄCE NA WYKONANIE KODU
JĘZYKA JAVASCRIPT POZA KONTEKSTEM
PRZEGŁĄDARKI.**



**UŻYCIE JAVASCRIPT POZA PRZEGŁĄDARKĄ
JEST MOŻLIWE DZIĘKI ZINTEGROWANIU
ŚRODOWISKA Z V8 JAVASCRIPT ENGINE.**





MINUSY:

- BRAK DOSTĘPU DO DOCUMENT OBJECT MODEL
- WYDAJNOŚĆ W OPERACJACH OBliczeniowych
- PROBLEMY Z KOMPATyBILNOŚCIĄ WSTECZNą

PLUSY:

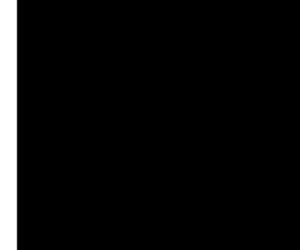
- MOŻLIWOŚĆ ODCZYTU I ZAPISÓW PLIKÓW NA KOMPUTERZE, DOSTĘP DO SYSTEMÓW PLIKÓW MASZYNY
- MOŻLIWOŚĆ ŁĄCZENIA Z BAZĄ DANYCH
- UŻYWAMY ZNANY NAM JAVASCRIPT
- MASA DOSTĘPNYCH PACZEK I NARZĘDZI UDOSTĘPNIONYCH PRZEZ SPOŁECZNOŚĆ
- WSPARCIE DLA ASYNCHRONICZNOŚCI I ZDARZENIOWY MODEL WEJŚCIA/WYJŚCIA
- ŁATWOŚĆ WDRAŻANIA

PRZEWAGI:

- EFEKTYWNOŚĆ W OBSŁUDZE DUŻYCH ILOŚCI RÓWNOCZESNYCH POŁĄCZEŃ
- WYDAJNOŚĆ W OBSŁUDZE WEJŚCIA-WYJŚCIA
- WSPARCIE DLA TWORZENIA APLIKACJI CZASU RZECZYWISTEGO

BRAKI:

- WYKONANIE OPERACJI OBliczeniowych
- BRAK WSPARCIA DLA WIELOWĄTKOWOŚCI
- ZŁOŻONOŚĆ KODU PRZY OBSŁUDZE ZAGNIEŻDŻONYCH FUNKCJI ZWROTNYCH
- WYDAJNOŚĆ W OBSŁUDZE DUŻYCH ILOŚCI ZAPYTAŃ SIECIOWYCH

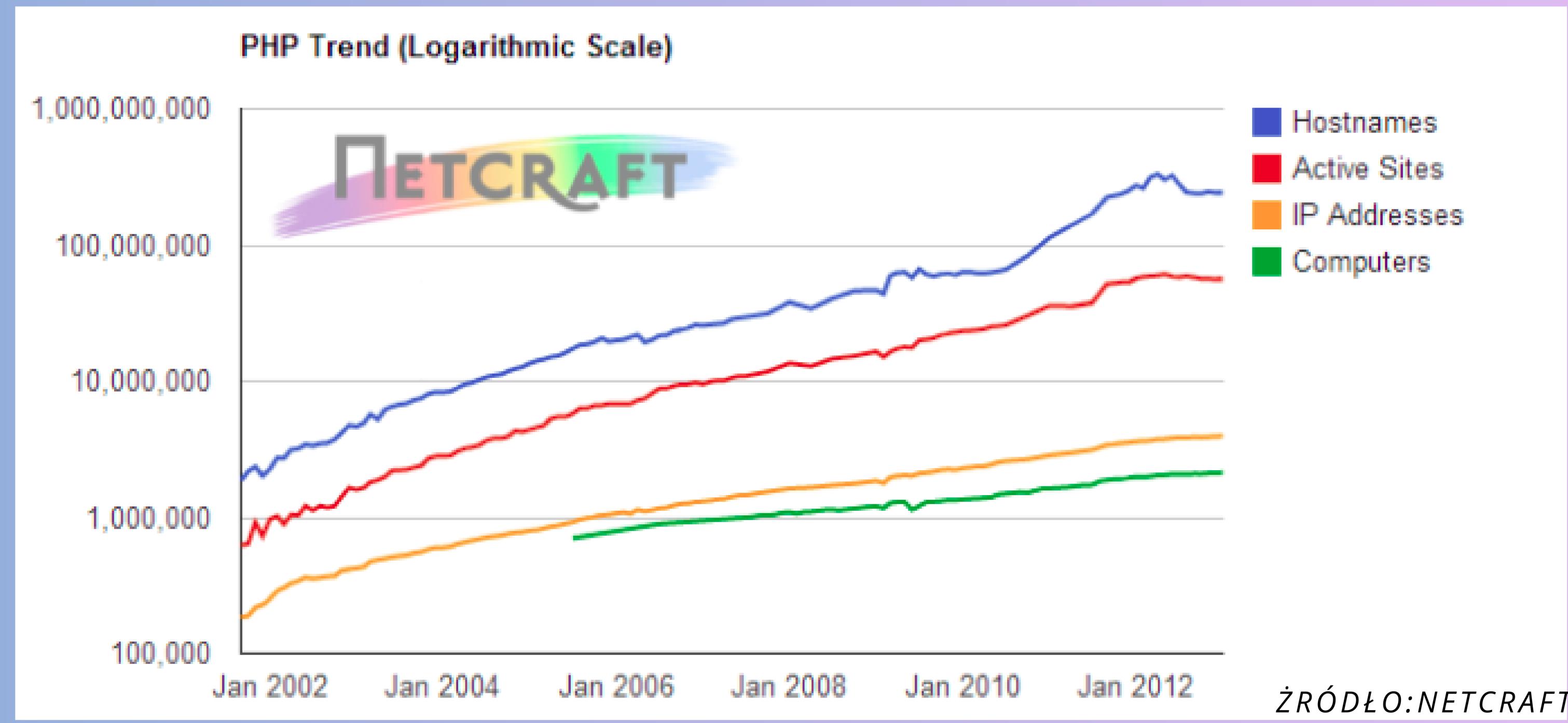


**PODSUMOWANIE TEGO CO
DZISIAJ SIĘ NAUCZYLIŚMY NA
TEMAT TYCH TECHNIK**

**Backend i co
dalej ?**



PHP: Czy warto ?



Thank You

