

PHP — pierwsze skrypty

Czas: 2 × 45 min

praca indywidualna

Cel: Uruchomienie pierwszych skryptów PHP na serwerze Apache z XAMPP: zmienne i typy, echo, instrukcje warunkowe i match, pętle generujące tabelę i kalendarz oraz czytanie komunikatów o błędach.

Po zajęciach potrafisz:

- uruchamia skrypty PHP z folderu htdocs przez adres `http://localhost`
- używa zmiennych różnych typów, operatorów, `echo` i skróconego `<?= ?>`
- stosuje `if/elseif/else` i wyrażenie `match`
- generuje tabelę HTML pętlami `for` i `while`
- odczytuje komunikat o błędzie PHP (rodzaj, plik, wiersz) i poprawia kod

Sprzęt i materiały

- komputer z Windows 11 i XAMPP
- folder `r2_05_start` z Pliki_do_zadan (`styl.css`, `bledy.php`)

Oprogramowanie

- **XAMPP (Apache, PHP 8, MariaDB, phpMyAdmin)** — projekt w folderze `C:\xampp\htdocs`
apachefriends.org/download.html
- **Visual Studio Code** — opcjonalnie rozszerzenie PHP Intelephense
code.visualstudio.com
- **Przeglądarka Chrome, Edge lub Firefox** — narzędzia deweloperskie F12 (Elementy, Konsola, Sieć, Aplikacja)

Przygotowanie przed zajęciami

- Przeglądnij wykład 2.04. Sprawdź, czy port 80 nie jest zajęty (XAMPP Control Panel → Netstat).

Bezpieczeństwo

- Plik z `phpinfo()` i włączone `display_errors` są tylko do pracy na własnym komputerze — na serwerze publicznym ujawniają konfigurację. Usuń `info.php` po wykonaniu polecenia.
- Nie wyłączaj zapory Windows, aby Apache był dostępny z sieci — pracuj na `localhost`.

Przebieg zadania

1 Serwer i pierwszy skrypt

Pliki PHP nie działają po otwarciu dwuklikiem — wykonuje je serwer Apache z modułem PHP, a przeglądarka dostaje gotowy HTML.

1. Uruchom Apache w XAMPP Control Panel. Utwórz folder `C:\xampp\htdocs\nazwisko_z05` i skopiuj do niego pliki z `r2_05_start`.
2. Utwórz `info.php` z wywołaniem `phpinfo()`; i otwórz `http://localhost/nazwisko_z05/info.php`. Zapisz: wersję PHP, ścieżkę pliku `php.ini` (Loaded Configuration File), wartości `display_errors` i `date.timezone`. Następnie usuń plik.
3. Otwórz plik `.php` dwuklikiem w przeglądarce (adres `file:///...`) i porównaj z adresem przez `localhost`. Sprawdź Ctrl+U, że do przeglądarki nie trafia kod PHP.

2 Zmienne, typy i powitanie (index.php)

Na początku skryptu ustaw strefę czasową: `date_default_timezone_set('Europe/Warsaw');`. Dołącz `styl.css` i dodaj menu z odnośnikami do trzech podstron.

1. Utwórz zmienne `$imie` (tekst), `$klasa` (liczba całkowita), `$srednia` (liczba z przecinkiem), `$zdany` (logiczna). Wyświetl zdanie z ich wartościami — raz z operatorem `.`, raz w cudzysłowie z interpolacją — oraz typ każdej zmiennej funkcją `gettype()`. Co wyświetla `echo true`?
2. Wyrażeniem `match (true)` dobierz powitanie do godziny `(int) date('G')`: do 5 — „Dobrej nocy”, do 12 — „Dzień dobry”, do 18 — „Miłego popołudnia”, później — „Dobry wieczór”. Wyświetl je z imieniem, godziną i datą w formacie `21.10.2026`.
3. Instrukcją `if/elseif/else` opisz średnią (od 4,75 — „świadectwo z wyróżnieniem”, od 3,0 — „dobra średnia”, niżej — „trzeba popracować”). Średnią wyświetl przez `number_format` z przecinkiem.
4. Wyświetl wyniki: `17 / 5`, `intdiv(17, 5)`, `17 % 5`, `2 ** 10` i stałą `PHP_VERSION`.

3 Pętle: tabliczka i kalendarz

Do mieszania PHP z HTML możesz użyć składni alternatywnej `for (...): ?> ... <?php endfor;`.

1. W `tabliczka.php` wygeneruj tabliczkę mnożenia `$n × $n` (`$n = 10`) z nagłówkami wierszy i kolumn (`th`). Komórki na przekątnej (kwadraty) oznacz klasą `kwadrat`. Sprawdź, że zmiana `$n` na 12 powiększa tabelę.
2. Pod tabelą oblicz pętlą `while` sumę kwadratów od 1 do `$n` i wyświetl ją.
3. W `kalendarz.php` wygeneruj kalendarz bieżącego miesiąca od poniedziałku: nazwa miesiąca po polsku z tablicy, liczba dni z `date('t', ...)`, dzień tygodnia pierwszego dnia z `date('N', mktime(0, 0, 0, $miesiac, 1, $rok))`, puste komórki przed pierwszym dniem i po ostatnim. Soboty i niedziele — klasa `weekend`, dzisiejszy dzień — klasa `dzis`.

4 Czytanie błędów (bledy.php)

Plik ma cztery błędy. Uruchamiaj go po każdej poprawce — PHP zgłasza błędy po kolei.

1. Dla każdego komunikatu zapisz w tabeli: rodzaj (Parse error, Warning, Fatal error), treść, numer wiersza i poprawkę. Nie usuwaj kodu — popraw go (np. dzielenie wykonaj tylko, gdy liczba osób jest większa od 0; brakującą funkcję `oblicz_rabat` napisz: 5% rabatu od kwoty co najmniej 50 zł).
2. Zrób zrzuty ekranu: jednego komunikatu o błędzie, `index.php`, `tabliczka.php` i `kalendarz.php`. Spakuj folder.

W sprawozdaniu umieść

- ☐ dane z `phpinfo()`: wersja PHP, plik `php.ini`, `display_errors`, `date.timezone`
- ☐ zrzuty ekranu trzech podstron i jednego komunikatu o błędzie

- ☐ tabela błędów z `bledy.php` (rodzaj, treść, wiersz, poprawka)
- ☐ kod generujący kalendarz
- ☐ spakowany folder `nazwisko_z05`

Pytania kontrolne

1. Dlaczego plik PHP otwarty przez `file:///` nie działa, a przez `http://localhost/` działa? Co widać w źródle strony (Ctrl+U)?
2. Czym różni się Warning od Fatal error — czy skrypt wykonuje się dalej?
3. Czym różni się `match` od `switch` (porównanie, zwracanie wartości, brak `break`)?
4. Dlaczego na serwerze produkcyjnym `display_errors` powinno być wyłączone?

Kryteria oceny

Kryterium	Punkty
Serwer, phpinfo (4 informacje), porównanie <code>file://</code> i <code>localhost</code>	3
<code>index.php</code> : zmienne 4 typów, <code>gettype</code> , interpolacja, <code>match</code> dla 4 pór dnia, <code>if/elseif</code> , operatory	5
Tabliczka mnożenia z nagłówkami, wyróżnioną przekątną i sumą (<code>while</code>)	4
Kalendarz: poprawny pierwszy dzień, puste komórki, weekendy, dzień dzisiejszy	4
Tabela 4 błędów z poprawkami, zrzuty, odpowiedzi na pytania	4
Razem	20

Sprawozdanie przygotuj na wzorze dołączonym do zadania (plik .docx o tym samym numerze).