Tablica informatyczna · egzamin INF.03

PHP

Połączenie z bazą MySQL/MariaDB, formularze i generowanie HTML — skrypty z arkuszy INF.03

Arkusz dołącza tabelę „Wybór funkcji języka PHP do obsługi bazy MySQL i MariaDB” (biblioteka mysqli, zapis proceduralny) oraz funkcję isset(). Reszta składni (pętle, tablice, formularze, łączenie napisów) należy do wiedzy zdającego.

●●● ponad połowa arkuszy 2023–2026 ●●○ co najmniej 15% ●○○ rzadziej ○○○ poza arkuszami, wiedza uzupełniająca

Liczone: treść zadania lub tabela pomocnicza arkusza

Podstawy języka

<?php … ?>

●●●

```
<?php  kod  ?>
```

Skrypt PHP wstawiasz w plik .php (index.php); wokół niego zwykły HTML. Serwer (Apache z XAMPP) wykonuje kod, a przeglądarce wysyła gotowy HTML.

```
<?php
  echo "<p>Witaj</p>";
?>
```

Uwaga. Plik uruchamiaj przez http://localhost/folder/index.php, nie przez dwukrotne kliknięcie. Folder zadania w htdocs.

Zmienne, echo

●●●

```
$nazwa = wartość;   echo "tekst";   echo $x;
```

Zmienne zaczynają się od $. echo wypisuje tekst i HTML na stronę. Kropka (.) skleja napisy; w cudzysłowie zmienna jest podstawiana.

```
$cena = 12;
echo "<p>Cena: " . $cena . " zł</p>";
echo "<p>Cena: $cena zł</p>";
```

Operatory

●●○

```
+ - * / %   .   == ===   != <>   && ||   !
```

Arytmetyczne, sklejanie napisów (.), porównania, logiczne. W obliczeniach cen: iloczyn, suma, połowa ceny (/ 2).

```
$wartosc = $dorosli * $cena + $dzieci * $cena / 2;
```

if … else

●●○

```
if (warunek) { … } elseif (inny) { … } else { … }
```

Instrukcja warunkowa. W arkuszach: „w przypadku opony letniej obraz lato.png, w przypadku zimowej zima.png”, „w przeciwnym wypadku wyświetla ‘Wprowadź wszystkie dane’”.

```
if ($wiersz["polubienia"] > 40) {
  echo "<p>Wiele osób polubiło ten obraz</p>";
} else {
  echo "<p>Autor: …</p>";
}
```

for, while, foreach

●●○

```
for ($i = 1; $i <= 20; $i++) { … }   while (warunek) { … }
```

Pętle. for — znana liczba powtórzeń (np. „wyświetla 20 razy grafikę”); while — dopóki są dane (wiersze z bazy); foreach — po elementach tablicy.

```
for ($i = 1; $i <= 20; $i++) {
  echo "<img src='obraz.png' alt='obraz'>";
}
```

Tablice

○○○

```
$t = ["a", "b"];   $t[0]   $w["nazwa"]   count($t)
```

Indeksowane od 0 albo asocjacyjne (klucz => wartość). count() — liczba elementów.

```
$miesiace = ["styczeń", "luty", "marzec"];
echo $miesiace[1];   // luty
```

Funkcje liczbowe i tekstowe

●○○

```
round()  floor()  ceil()  rand(a, b)  strlen()
strtoupper()  substr()  str_replace()  explode()  date()
```

round(x, 2) — zaokrągla; floor/ceil — w dół/w górę; rand(a, b) — losowa całkowita; strlen — długość; substr($s, od, ile) — fragment; strtolower/strtoupper — wielkość liter; date("Y-m-d") — bieżąca data.

```
$kod = strtolower(substr($imie, 0, 2) . substr($nazwisko,
0, 2));
$cena2 = round($cena * 0.85, 2);
```

Baza danych — funkcje mysqli (tabela z arkusza)

Zestaw funkcji, który arkusz podaje jako pomoc. Zapis proceduralny: pierwszy parametr to identyfikator połączenia z mysqli_connect.

mysqli_connect

●●●

```
$polaczenie = mysqli_connect("localhost", "root", "",
"baza");
```

Łączy z serwerem: serwer, użytkownik, hasło, nazwa bazy. Zwraca identyfikator połączenia lub FALSE. W arkuszach: localhost, root bez hasła, baza o podanej nazwie.

```
$polaczenie = mysqli_connect("localhost", "root", "",
"samochody");
```

Uwaga. Hasło puste: "". Nazwa bazy dokładnie jak w phpMyAdmin (z zadania).

mysqli_select_db

●●●

```
mysqli_select_db($polaczenie, "baza");
```

Wybiera bazę, gdy nie podano jej w mysqli_connect. Zwraca TRUE/FALSE.

```
mysqli_select_db($polaczenie, "biblioteka");
```

© Dawid Chaber · TablicaPro (tablicapro.pl) · materiał chroniony prawem autorskim — kopiowanie i udostępnianie bez zgody autora zabronione

mysqli_query ●●●

```
$wynik = mysqli_query($polaczenie, "SELECT …");
```

Wysyła zapytanie do bazy. Dla SELECT zwraca zbiór wyników; dla INSERT/UPDATE/DELETE — TRUE/FALSE. „Wysyła do bazy danych zapytanie 3”.

```
$zapytanie = "SELECT nazwa, cena FROM towary";
$wynik = mysqli_query($polaczenie, $zapytanie);
```

> **Uwaga.** Zapytanie przepisz z pliku kwerendy.txt. „Zmodyfikowane zapytanie” — podstaw wartość z formularza w warunku WHERE.

mysqli_fetch_row ●●●

```
while ($wiersz = mysqli_fetch_row($wynik)) { …
$wiersz[0] … }
```

Pobiera kolejny wiersz wyniku jako tablicę indeksowaną numerami kolumn (0, 1, 2…). Po ostatnim wierszu zwraca FALSE — pętla while kończy się sama.

```
while ($wiersz = mysqli_fetch_row($wynik)) {
  echo "<li>" . $wiersz[0] . " - " . $wiersz[1] . "
</li>";
}
```

mysqli_fetch_array / _assoc ●●●

```
$wiersz = mysqli_fetch_array($wynik);   $wiersz["nazwa"]
```

Kolejny wiersz jako tablica z kluczami numerycznymi i nazwami kolumn (fetch_array) lub tylko nazwami (fetch_assoc). Zwraca FALSE, gdy nie ma więcej wierszy.

```
while ($wiersz = mysqli_fetch_assoc($wynik)) {
  echo $wiersz["nazwa"] . " " . $wiersz["cena"];
}
```

mysqli_num_rows, mysqli_num_fields ●●●

```
mysqli_num_rows($wynik)   mysqli_num_fields($wynik)
```

Liczba wierszy i kolumn w wyniku zapytania. Przydatne do sprawdzenia, czy zapytanie coś zwróciło, oraz do pętli po kolumnach.

```
if (mysqli_num_rows($wynik) > 0) { … }
```

mysqli_error ●●●

```
echo mysqli_error($polaczenie);
```

Tekst komunikatu o błędzie ostatniej operacji — pomoc przy debugowaniu błędnego zapytania.

```
mysqli_query($polaczenie, $zapytanie) or
die(mysqli_error($polaczenie));
```

> **Uwaga.** W PHP 8.1 i nowszym błąd zapytania domyślnie zgłaszany jest wyjątkiem (mysqli_sql_exception) i komunikat widać od razu na stronie — warunek if (!$wynik) i or die() się wtedy nie wykonają. mysqli_error zwraca tekst błędu, gdy tryb wyjątków jest wyłączony: mysqli_report(MYSQLI_REPORT_OFF). W większości arkuszy sprawdzanie, czy operacja się powiodła, nie jest wymagane.

mysqli_close ●●●

```
mysqli_close($polaczenie);
```

Zamyka połączenie. Wymagane na końcu skryptu: „Na końcu działania skrypt zamyka połączenie z serwerem”.

```
mysqli_close($polaczenie);
```

Styl obiektowy mysqli

```
$p = new mysqli("localhost", "root", "", "baza");   $p-
>query(…)   $w->fetch_assoc()   $p->close();
```

Ta sama biblioteka w zapisie obiektowym — równoważny, jeśli jest spójny w całym skrypcie. Rozwiązania wzorcowe CKE korzystają z obu zapisów.

```
$p = new mysqli("localhost", "root", "", "baza");
$wynik = $p->query("SELECT nazwa FROM towary");
while ($w = $wynik->fetch_assoc()) { echo $w["nazwa"]; }
$p->close();
```

Występuje w rozwiązaniach wzorcowych CKE

mysqli_real_escape_string

```
mysqli_real_escape_string($polaczenie, $tekst)
```

Zabezpiecza dane z formularza przed wstrzyknięciem SQL (ucieczka apostrofów). W arkuszach nie jest wymagana, ale to dobry nawyk; docelowo używa się zapytań przygotowanych.

```
$imie = mysqli_real_escape_string($polaczenie,
$_POST["imie"]);
```

Dane z formularza: $_POST, $_GET, isset

$_POST ●●●

```
$_POST["nazwa_pola"]
```

Tablica z danymi wysłanymi formularzem metodą post. Kluczem jest atrybut name kontrolki; wartość select/radio to jej value.

```
$miejsce = $_POST["miejsce"];
$dorosli = $_POST["dorosli"];
```

isset() ●○○

```
if (isset($_POST["przycisk"])) { … }
```

TRUE, gdy zmienna istnieje — tak sprawdzasz, czy formularz wysłał dane („wykonywany tylko, gdy formularz wysłał dane”). Funkcja jest w tabeli pomocniczej arkusza.

```
if (isset($_POST["miejsce"])) {
  // zapytanie zmodyfikowane danymi z formularza
}
```

> **Uwaga.** Wzorzec „tylko jeżeli wysłano formularz”: najpierw isset, dopiero potem zapytanie. Przy braku danych strona wyświetla się bez wyniku.

$_GET ●○○

```
index.php?id=3   →   $_GET["id"]
```

Parametry z adresu URL (odnośnik z ?id=…). Używane np. do szczegółów rekordu lub usuwania wiersza przez link.

```
$id = $_GET["id"];
$wynik = mysqli_query($polaczenie, "SELECT * FROM potrawy
WHERE id = $id");
```

> **Uwaga.** Tabela „Metoda GET i odnośniki” pojawiła się w arkuszach 2026. W odnośnikach generowanych pętlą: echo "…";

Zapytanie zmodyfikowane danymi z formularza ●●○

```
"SELECT … WHERE kolumna = '$zmienna'"
```

Wartość z formularza wstawiasz do warunku zapytania. Tekst w apostrofach, liczbę bez. Skrypt łączy napisy kropką lub podstawia zmienną w cudzysłowie.

```
$zapytanie = "SELECT nazwa, cena FROM towary WHERE nazwa
= '" . $_POST["towar"] . "'";
```

© Dawid Chaber · TablicaPro (tablicapro.pl) · materiał chroniony prawem autorskim — kopiowanie i udostępnianie bez zgody autora zabronione

INSERT z formularza ●○○

`mysqli_query($p, "INSERT INTO t (a, b) VALUES ('$a', '$b')");`

Dane z formularza zapisujesz do bazy zapytaniem INSERT; po wykonaniu wyświetlasz potwierdzenie („Dane uczestnika … zostały dodane”). Klucz główny AUTO_INCREMENT wstaw jako NULL albo pomiń kolumnę.

```
$zapytanie = "INSERT INTO uczestnicy (imie, nazwisko,
wiek) VALUES ('$imie', '$nazwisko', $wiek)";
mysqli_query($polaczenie, $zapytanie);
echo "<p>Dane uczestnika $imie $nazwisko zostały
dodane</p>";
```

DELETE przez odnośnik ●○○

`<a href="index.php?usun=5">Usuń</a>`

Odnośnik z parametrem GET wywołuje na tej samej stronie zapytanie DELETE dla wskazanego id.

```
if (isset($_GET["usun"])) {
  mysqli_query($polaczenie, "DELETE FROM zadania WHERE
id_zadania = " . $_GET["usun"]);
}
```

Wzorce generowania HTML z wyników zapytania

Lista z wyników ●●○

`<ul> + while + <li>`

Wypisanie wszystkich zwróconych rekordów jako elementów listy. Skrypt ma wypisać tyle wierszy, ile zwróci zapytanie — nie na sztywno.

```
echo "<ul>";
while ($w = mysqli_fetch_row($wynik)) {
  echo "<li>$w[0] $w[1]</li>";
}
echo "</ul>";
```

Lista rozwijana z bazy ●●○

`<select> + while + <option value="id">nazwa</option>`

Każdy wiersz wyniku staje się opcją: id trafia do atrybutu value, nazwa jest treścią opcji. Po wysłaniu formularza używasz $_POST[...] jako id.

```
echo "<select name='towar'>";
while ($w = mysqli_fetch_row($wynik)) {
  echo "<option value='$w[0]'>$w[1]</option>";
}
echo "</select>";
```

Tabela z wyników ●○○

`wiersz <tr><th> w HTML + while + <tr><td>`

Wiersz nagłówkowy zapisz w HTML, a wiersze danych dołóż w pętli; każdą kolumnę w osobnym td.

```
echo "<table><tr><th>Nr</th><th>Nazwa</th></tr>";
while ($w = mysqli_fetch_row($wynik)) {
  echo "<tr><td>$w[0]</td><td>$w[1]</td></tr>";
}
echo "</table>";
```

Bloki z obrazem ●○○

`<div class="klasa"><img src="pole" alt="…"><h4>…</h4></div>`

Dla każdego rekordu osobny blok (z klasą CSS), a w nim obraz o nazwie z bazy, nagłówki i paragrafy. Nazwę pliku może też tworzyć skrypt (kod + „.jpg”).

```
while ($w = mysqli_fetch_row($wynik)) {
  echo "<div class='opona'>";
  echo "<img src='$w[0].jpg' alt='kurs'>";
  echo "<h4>Opona: $w[1] $w[2]</h4>";
  echo "</div>";
}
```

Pojedynczy rekord ●●○

`$w = mysqli_fetch_row($wynik);  echo $w[0];`

Gdy zapytanie zwraca jeden wiersz (np. po id), pobierasz go raz, bez pętli.

```
$w = mysqli_fetch_row($wynik);
echo "<h2>$w[0] model $w[1]</h2>";
```

Warunkowy wygląd rekordu ●○○

`if ($w["sezon"] == "letnia") { … }`

W zależności od wartości pola wybierz obraz lub tekst (np. opona letnia → lato.png, zimowa → zima.png).

```
if ($sezon == "letnia") $obraz = "lato.png";
elseif ($sezon == "zimowa") $obraz = "zima.png";
else $obraz = "uniwer.png";
```

Odświeżanie strony ●○○

`header("Refresh: 10");  <meta http-equiv="refresh" content="10">`

Automatyczne ponowne wczytanie strony co n sekund („Strona jest odświeżana co 10 sekund”). header() musi być wywołany przed jakimkolwiek wypisaniem treści.

`<?php header("Refresh: 10"); ?>`

Ciasteczka, sesje, pliki (starsze arkusze)

setcookie, $_COOKIE ○○○

`setcookie("ciastko", "1", time() + 3600);  $_COOKIE["ciastko"]`

Zapis i odczyt ciasteczka. Trzeci parametr — czas wygaśnięcia jako znacznik czasu (time() + 3600 = godzina). setcookie przed jakąkolwiek treścią.

```
if (!isset($_COOKIE["ciastko"])) {
  setcookie("ciastko", "1", time() + 3600);
  echo "Witaj po raz pierwszy";
}
```

Uwaga. Pojawiło się w arkuszu z 2019 roku; od 2023 nie wystąpiło w opisach zadań, ale pozostaje w podstawie programowej.

session_start, $_SESSION ●○○

`session_start();  $_SESSION["login"] = "ania";`

Sesja — dane zapamiętane po stronie serwera między odsłonami strony. session_start() na początku pliku.

```
<?php
session_start();
$_SESSION["koszyk"] = 3;
?>
```

© Dawid Chaber · TablicaPro (tablicapro.pl) · materiał chroniony prawem autorskim — kopiowanie i udostępnianie bez zgody autora zabronione

include / require

```
require "polaczenie.php";
```

Dołącza inny plik PHP — wspólny kod połączenia z bazą w jednym miejscu.

```
require "polaczenie.php";
```

Występuje w rozwiązaniach wzorcowych CKE

Pliki: file_put_contents

```
file_put_contents("plik.txt", $tekst, FILE_APPEND);
```

Zapis tekstu do pliku (z dopisywaniem). W starszych zadaniach m.in. licznik odwiedzin.

```
file_put_contents("licznik.txt", $wartosc);
```

Występuje w rozwiązaniach wzorcowych CKE

Z arkusza do kodu: typowe sformułowania

Polecenia w arkuszach powtarzają się. Po lewej zwrot z treści zadania, po prawej jego zapis.

Łączy się z serwerem bazodanowym na localhost, użytkownik root bez hasła, baza biblioteka

```
$p = mysqli_connect("localhost", "root", "",
"biblioteka");
```

Wysyła do bazy danych zapytanie 2 (z pliku kwerendy.txt)

```
$w = mysqli_query($p, "SELECT … (przepisane z
kwerendy.txt)");
```

Każdy zwrócony wiersz wypisuje w elemencie listy / wierszu tabeli

```
while ($r = mysqli_fetch_row($w)) { echo "<li>$r[0]
</li>"; }
```

Wartość pola zwrócona zapytaniem w nagłówku drugiego stopnia

```
echo "<h2>$r[0]</h2>";
```

Zwrócone nazwy są opcjami listy rozwijanej, id w atrybucie value

```
echo "<option value='$r[0]'>$r[1]</option>";
```

Wykonywany tylko, gdy formularz wysłał dane

```
if (isset($_POST["przycisk"])) { … }
```

Zapytanie zmodyfikowane: w warunku wartość wysłana z formularza

```
$z = "SELECT … WHERE id = " . $_POST["id"];
```

Wylicza wartość na podstawie ilości i ceny zwróconych z bazy

```
echo "<p>" . $r[0] * $r[1] . " zł</p>";
```

Wstawia dane z formularza do bazy (zmodyfikowane zapytanie INSERT)

```
mysqli_query($p, "INSERT INTO t VALUES (NULL, '$a',
'$b')");
```

Wyświetla 20 razy grafikę (pętla)

```
for ($i = 0; $i < 20; $i++) echo "<img src='obraz.png'
alt='obraz'>";
```

Odnośnik przesyła metodą GET identyfikator

```
echo "<a href='przepisy.php?id=$r[0]'>$r[1]</a>";
```

Na końcu działania skrypt zamyka połączenie z serwerem

```
mysqli_close($p);
```

Dwa skrypty w jednym pliku

```
Każdy w osobnym bloku <?php … ?> w miejscu strony, które
ma wypełnić; połączenie otwórz raz na górze i zamknij raz
na dole.
```

Przygotowanie stron na podstawie gotowego (kopia + zmiana warunku)

```
Skopiuj plik, zmień nazwę i tylko warunek zapytania (id
rasy = 6) oraz obraz.
```

Typowe błędy

Plik .html zamiast .php lub otwarty z dysku

Kod PHP wykona się tylko w pliku .php uruchomionym przez serwer: http://localhost/.../index.php. Serwis Apache i MySQL włącz w XAMPP Control Panel.

Brak znaku $ lub średnika

Każda zmienna zaczyna się od $, każda instrukcja kończy średnikiem. Komunikat „Parse error” wskazuje linię.

Cudzysłów w cudzysłowie

W echo "..." zapisuj atrybuty HTML w apostrofach: <img src='a.jpg'> — albo poprzedź cudzysłów ukośnikiem: \".

Zapytanie bez apostrofów wokół tekstu

WHERE nazwa = 'Opel' — tekst w apostrofach, liczby bez.

fetch_row bez pętli while

Pętla while ($w = mysqli_fetch_row($wynik)) wypisuje wszystkie wiersze; samo wywołanie — tylko pierwszy.

Zła kolejność parametrów mysqli_query

Najpierw połączenie, potem zapytanie: mysqli_query($polaczenie, $zapytanie).

Brak mysqli_close albo nazwy bazy z literówką

Skopiuj nazwę bazy z zadania; zamknięcie połączenia zapisz na końcu skryptu.

Wypisywanie wyniku mimo braku danych z formularza

Osłoń blok warunkiem isset($_POST["..."]), by w stanie początkowym strony nie było komunikatów błędów.

header() po wypisaniu treści

Wywołaj header lub setcookie przed jakimkolwiek HTML i przed echo.

Opracowanie TablicaPro na podstawie analizy arkuszy części praktycznej egzaminu zawodowego (E.14, EE.09, INF.03) z lat 2016–2026 opublikowanych przez CKE i OKE: 123 arkuszy z treścią, w tym 70 z lat 2023–2026. Znaczniki częstości są szacunkiem na podstawie automatycznego wyszukiwania w treści zadań i tabelach pomocniczych — mają orientacyjny charakter. Arkusze są własnością CKE; tablica zawiera własne opisy i przykłady.

© Dawid Chaber · TablicaPro (tablicapro.pl) · materiał chroniony prawem autorskim — kopiowanie i udostępnianie bez zgody autora zabronione