PORADNIK DLA UCZNIA / EGZAMIN PRAKTYCZNY INF.03

PHP i baza danych krok po kroku

Wersja prosta: **if (isset(...))**, **mysqli_fetch_row()**, kolumny **[0], [1], [2]**, pętla **while** oraz **die()**. Przykłady z arkusza „Islandia”, czerwiec 2026.



Po tym poradniku potrafisz:

- Połączyć stronę PHP z bazą **islandia** i wysłać zapytanie SQL.
- Wyświetlić **jeden rekord** oraz **wszystkie rekordy** w pętli while.
- Korzystać z kolumn **$w[0]**, **$w[1]** i **$w[2]**.
- Odebrać **$_GET["id"]** przez **if (isset(...))**.

Schemat na cały egzamin: POŁĄCZ → ZAPYTANIE → POBRANIE WIERSZA → WYŚWIETLENIE → ZAMKNIJ

MODUŁ 01 / PHP + MYSQL

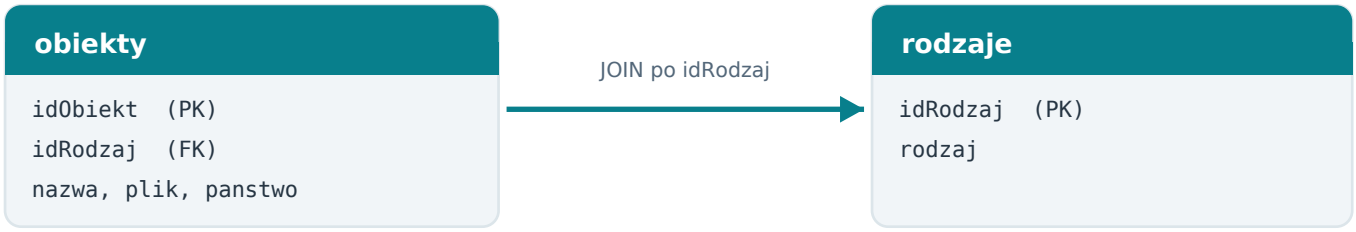
Przygotowanie do pracy

Działamy na prawdziwej bazie z załączników arkusza INF.03-01-26.06-SG. Wszystkie nazwy tabel i kolumn pochodzą z pliku **baza1.sql**.

Zrób to w XAMPP i phpMyAdmin

- Uruchom **Apache** i **MySQL** w panelu XAMPP.
- Otwórz **http://localhost/phpmyadmin**.
- Utwórz bazę **islandia**, ustaw porównywanie znaków **utf8_unicode_ci**.
- Wejdź do bazy, wybierz **Import** i wczytaj plik **baza1.sql**.
- Do folderu **htdocs/islandia** skopiuj pliki PHP i 9 zdjęć z ZIP.

Pola, które wykorzystamy



Obiekt ma przypisany rodzaj, np. Godafoss → wodospad.

Tabela	Ważne kolumny	Do czego służą
obiekty	idObiekt, plik, nazwa, idRodzaj, panstwo, opis, nazwaCechy, wartoscCechy	Miejsca, zdjęcia, szczegóły
rodzaje	idRodzaj, rodzaj	Typ miejsca (np. wodospad)
kontynenty	idKontynent, nazwa	Jest w bazie, lecz nie jest potrzebna do tych skryptów

Ważne w zadaniu: Arkusz wymaga dwóch stron: **islandia.php** i **obiekty.php**. W obu są listy z bazy, a na pierwszej dodatkowo galeria, na drugiej opis wybranego miejsca.

MODUŁ 02 / PHP + MYSQL

Najpierw zrozum SELECT

Zapytanie SQL wybiera dane. PHP dostaje wynik, a następnie wyświetla go w HTML. Uruchom poniższe zapytania w zakładce SQL w phpMyAdmin.

1 / Nazwy wodospadów

```
SELECT nazwa FROM obiekty
WHERE panstwo = 'Islandia' AND idRodzaj = 10;
```

Zwracana jest **jedna kolumna: nazwa**. Ma indeks **[0]**, ponieważ liczymy od zera.

2 / Dane do galerii (zapytanie 1 z arkusza)

```
SELECT idObiekt, plik, nazwa FROM obiekty
WHERE panstwo = 'Islandia';
```

[0] = idObiekt, **[1] = plik**, **[2] = nazwa**. To kolejność z SELECT, nie kolejność pól w strukturze tabeli.

3 / Opis miejsca z JOIN (zapytanie 2 z arkusza)

```
SELECT obiekty.plik, obiekty.nazwa,
       obiekty.nazwaCechy, obiekty.wartoscCechy,
       obiekty.opis, rodzaje.rodzaj
FROM obiekty
JOIN rodzaje ON obiekty.idRodzaj = rodzaje.idRodzaj
WHERE obiekty.idObiekt = 46;
```

JOIN łączy dwie tabele po wspólnym **idRodzaj**. Indeksy tego zapytania: **[0] plik, [1] nazwa, [2] nazwaCechy, [3] wartoscCechy, [4] opis, [5] rodzaj**.

4 / Zliczanie (zapytanie 4 z arkusza)

```
SELECT rodzaje.rodzaj,
       COUNT(obiekty.idObiekt) AS 'Liczba obiektów'
FROM rodzaje
JOIN obiekty ON rodzaje.idRodzaj = obiekty.idRodzaj
GROUP BY rodzaje.idRodzaj, rodzaje.rodzaj;
```

Do zapamiętania: Kolumny **[0], [1], [2]** to kolejność pól wymienionych po **SELECT**. Przy innym SELECT te same numery mogą znaczyć coś innego.

MODUŁ 03 / PHP + MYSQL

Połączenie z bazą PHP + mysqli

Zaczynamy od najprostszego schematu. Arkusz zakłada lokalny serwer **localhost**, użytkownika **root**, puste hasło i bazę **islandia**.

MINIMALNY SCHEMAT POŁĄCZENIA

```
<?php
mysqli_report(MYSQLI_REPORT_OFF); // dla prostych testów
$db = mysqli_connect("localhost", "root", "", "islandia");

if (!$db) {
    die("Brak połączenia z bazą!");
}

mysqli_set_charset($db, "utf8mb4");

// Tutaj wpisujemy zapytania i wyświetlamy dane.

mysqli_close($db);
?>
```

Co oznacza każda linijka?

Kod	Znaczenie
mysqli_connect(...)	Próbuje nawiązać połączenie z bazą MySQL/MariaDB.
if (!$db) { ... }	Jeżeli połączenie się nie udało, wykonaj instrukcje w klamrach.
mysqli_set_charset(...)	Ustawia kodowanie komunikacji z bazą, także dla polskich liter.
mysqli_close($db)	Zamyka połączenie. Umieść na końcu pracy z bazą.
mysqli_report(MYSQLI_REPORT_OFF)	W ćwiczeniu pozwala obsługiwać nieudane operacje przez if i die zamiast wyjątków PHP.

Nie kopiuj bez zastanowienia: Konto **root bez hasła** jest ustawieniem wyłącznie szkolnego środowiska z arkusza. W prawdziwej aplikacji używa się osobnego konta bazy z ograniczonymi uprawnieniami.

MODUŁ 04 / PHP + MYSQL

die() — co to jest i po co?

Instrukcja **die()** wyświetla komunikat (jeżeli go podasz) i natychmiast **kończy wykonanie skryptu PHP**. To odpowiednik **exit()**.

Przykład, który od razu pokazuje sens die()

PRZYKŁAD 1

```
<?php
echo "Początek<br>";
die("Wykryto błąd. Koniec programu.");
echo "Tego tekstu NIE zobaczysz.";
?>
```

Przeglądarka pokaże **Początek** i **Wykryto błąd. Koniec programu.** Ostatnie echo nigdy nie zostanie wykonane.

Po co używamy die() przy bazie?

PRZYKŁAD 2

```
$db = mysqli_connect("localhost", "root", "", "islandia");

if (!$db) {
    die("Nie można połączyć się z bazą.");
}

echo "Połączenie działa.";
```

Jeżeli nie ma połączenia, nie ma sensu wykonywać kolejnych zapytań SQL. **die()** zatrzymuje kod i jasno informuje o błędzie.

Gdy błędne jest zapytanie SQL

PRZYKŁAD 3 — W LABORATORIUM

```
$wynik = mysqli_query($db, "SELECT nazwa FROM obiekty");
if (!$wynik) {
    die("Błąd zapytania: " . mysqli_error($db));
}
```

Ważne: Szczegółowy komunikat **mysqli_error()** pokazuj tylko na swoim komputerze podczas nauki i testów. Na publicznej stronie nie ujawnia się błędów bazy ani szczegółów serwera.

MODUŁ 05 / PHP + MYSQL

Jeden rekord, czyli mysqli_fetch_row()

mysqli_fetch_row($wynik) pobiera jeden kolejny wiersz jako tablicę z numerami kolumn: **[0]**, **[1]**, **[2]**...

Przykład: pierwsze znalezione miejsce

KOD Z INDEKSAMI KOLUMN

```
$sql = "SELECT idObiekt, plik, nazwa FROM obiekty
        WHERE panstwo='Islandia'";
$wynik = mysqli_query($db, $sql);
if (!$wynik) die("Błąd zapytania");

$w = mysqli_fetch_row($wynik);
echo "ID: " . $w[0] . "<br>";
echo "Plik: " . $w[1] . "<br>";
echo "Nazwa: " . $w[2];
```

SELECT	Indeks w PHP	Przykładowa wartość
idObiekt	$w[0]	46
plik	$w[1]	Strokkur.jpg
nazwa	$w[2]	Strokkur

Uwaga: [0] NIE oznacza pierwszego rekordu!

$w[0] oznacza **pierwszą kolumnę aktualnie pobranego wiersza**. Aby przejść do następnego rekordu, wywołaj jeszcze raz **mysqli_fetch_row()** albo zastosuj pętlę **while**.

DWA KOLEJNE WIERSZE (tylko przykład)

```
$pierwszy = mysqli_fetch_row($wynik);
$drugi   = mysqli_fetch_row($wynik);
```

Pamiętaj: Bez **ORDER BY** nie należy zakładać stałej kolejności rekordów. Wartości Strokkur, 46 i Strokkur.jpg pokazują przykład danych obecnych w załączniku, a nie gwarantowany pierwszy wiersz.

MODUŁ 06 / PHP + MYSQL

Wiele rekordów — pętla while

To najważniejszy schemat na egzamin. Pętla pobiera następny rekord tak długo, jak długo zapytanie zwraca kolejne wiersze.

Pętla wypisująca nazwy obiektów

PRZYKŁAD A — JEDNA KOLUMNA

```
$sql = "SELECT nazwa FROM obiekty
        WHERE panstwo='Islandia'";
$wynik = mysqli_query($db, $sql);
if (!$wynik) die("Błąd zapytania");

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

W zapytaniu wybraliśmy tylko **nazwa**, dlatego nazwa jest w **$w[0]**. Dla każdego obiektu powstaje osobny paragraf HTML.

Pętla wykorzystująca trzy kolumny

PRZYKŁAD B — TRZY KOLUMNY

```
$sql = "SELECT idObiekt, plik, nazwa FROM obiekty
        WHERE panstwo='Islandia'";
$wynik = mysqli_query($db, $sql);
if (!$wynik) die("Błąd zapytania");

while ($w = mysqli_fetch_row($wynik)) {
    echo "<p>Numer: " . $w[0] . "</p>";
    echo "<p>Zdjęcie: " . $w[1] . "</p>";
    echo "<h3>" . $w[2] . "</h3>";
}
```

Przeczytaj pętlę po polsku: „**Dopóki** pobranie kolejnego wiersza się udaje, zapisz go do zmiennej **$w** i wykonaj wszystkie instrukcje z klamer.” Pętla kończy się automatycznie po ostatnim wierszu.

MODUŁ 07 / PHP + MYSQL

if (isset(...)) — prosty odbiór GET

isset() sprawdza, czy dana zmienna istnieje i ma wartość inną niż **null**. Na egzaminie używamy go najczęściej, aby sprawdzić parametr w adresie.

Adres z parametrem

PRZYKŁADOWY ADRES

```
http://localhost/islandia/obiekty.php?id=46
```

Tutaj **id** to nazwa parametru, a **46** to jego wartość. PHP może odczytać ją przez **$_GET["id"]**.

Zwykły if zamiast zapisu ze znakiem ?

WZÓR NA EGZAMIN

```
<?php
if (isset($_GET["id"])) {
    $id = (int)$_GET["id"];
    echo "Wybrano obiekt numer: " . $id;
} else {
    echo "Nie podano identyfikatora obiektu.";
}
?>
```

Co oznaczają elementy tego kodu?

Element	Znaczenie
isset($_GET["id"])	Czy w adresie przesłano parametr id?
(int)$_GET["id"]	Zamienia pobraną wartość na liczbę całkowitą.
if (...) { ... }	Wykonuje kod tylko wtedy, gdy warunek jest prawdziwy.
else { ... }	Alternatywa, gdy id nie zostało przesłane.

Bezpieczniejszy warunek: (int) upraszcza obsługę liczbowego ID i nie pozwala wstawić do tego zapytania tekstu SQL. Dla danych tekstowych i bardziej rozbudowanych formularzy stosuj zapytania przygotowane.

MODUŁ 08 / PHP + MYSQL

Skrypty 1 i 2 — listy z bazy

Arkusz wymaga na **obu stronach** list: wodospady (idRodzaj = 10) i siedliska zwierząt (idRodzaj = 14). Wystarczy SELECT, while i echo z <li>.

Skrypt 1: wodospady

KOD DO WSTAWIENIA WEWNĄTRZ

```
<li>Wodospady:
  <ol>
  <?php
  $sql = "SELECT nazwa FROM obiekty
          WHERE panstwo='Islandia' AND idRodzaj=10";
  $wynik = mysqli_query($db, $sql);
  if (!$wynik) die("Błąd listy wodospadów");

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

Skrypt 2: siedliska zwierząt

KOD DO WSTAWIENIA WEWNĄTRZ

```
<li>Siedliska zwierząt:
  <ol>
  <?php
  $sql = "SELECT nazwa FROM obiekty
          WHERE panstwo='Islandia' AND idRodzaj=14";
  $wynik = mysqli_query($db, $sql);
  if (!$wynik) die("Błąd listy siedlisk");

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

Tutaj zapytanie **SELECT nazwa** zwraca jedną kolumnę. Dlatego w obu pętlach wystarczy **$w[0]**.

MODUŁ 09 / PHP + MYSQL

Skrypt 3 — galeria z linkami

Zadanie wymaga, aby obraz był odnośnikiem do **obiekty.php**, a identyfikator obiektu był przekazywany przez GET. Ponadto nazwa miejsca ma pojawiać się w alt i title.

Kolejność indeksów

ZAPYTANIE Z ARKUSZA

```
SELECT idObiekt, plik, nazwa
FROM obiekty WHERE panstwo='Islandia';
```

Indeks	Zawartość	Użycie
$w[0]	idObiekt	Parametr GET w linku
$w[1]	plik	Atrybut src zdjęcia
$w[2]	nazwa	Atrybuty alt i title

Kod galerii

PĘTLA TWORZY WIELE ODNOŚNIKÓW I ZDJĘĆ

```
<h2>Galeria</h2>
<section>
<?php
$sql = "SELECT idObiekt, plik, nazwa FROM obiekty
        WHERE panstwo='Islandia'";
$wynik = mysqli_query($db, $sql);
if (!$wynik) die("Błąd galerii");

while ($w = mysqli_fetch_row($wynik)) {
    echo "<a href='obiekty.php?id=" . $w[0] . "'>";
    echo "<img class='miniatury' src='" . $w[1] . "'";
    echo " alt='" . $w[2] . "' title='" . $w[2] . "'>";
    echo "</a>";
}
?>
</section>
```

Dla obiektu o ID 46 zbuduje się link **obiekty.php?id=46** i zdjęcie **Strokkur.jpg**. Każdy kolejny przebieg while tworzy następny obraz.

Praktyczna uwaga: W szkolnej bazie nazwy plików i obiektów pochodzą z przygotowanego importu. Przy treściach od użytkowników należy wyświetlać tekst w HTML bezpiecznie, np. przez **htmlspecialchars()** (pokazane później).

MODUŁ 10 / PHP + MYSQL

Skrypt 4 — opis wybranego obiektu

Kliknięcie obrazka prowadzi np. do **obiekty.php?id=46**. Teraz chcemy pobrać tylko ten jeden obiekt i wyświetlić jego dane.

Krok 1: if isset, SELECT z JOIN, jeden wiersz

SCHEMAT ODCZYTU Z BAZY

```
<?php
if (isset($_GET["id"])) {
    $id = (int)$_GET["id"];

    $sql = "SELECT obiekty.plik, obiekty.nazwa,
            obiekty.nazwaCechy, obiekty.wartoscCechy,
            obiekty.opis, rodzaje.rodzaj
            FROM obiekty
            JOIN rodzaje ON obiekty.idRodzaj=rodzaje.idRodzaj
            WHERE obiekty.idObiekt=$id";

    $wynik = mysqli_query($db, $sql);
    if (!$wynik) die("Błąd zapytania");

    if (mysqli_num_rows($wynik) > 0) {
        $w = mysqli_fetch_row($wynik);
        // Tutaj wyświetlamy dane z $w[0] ... $w[5].
    } else {
        echo "Nie znaleziono obiektu.";
    }
} else {
    echo "Brak identyfikatora w adresie.";
}
?>
```

Krok 2: kolejność kolumn po SELECT

Kolumna	Znaczenie
$w[0] — plik	Ścieżka do zdjęcia
$w[1] — nazwa	Nagłówek h2 i alt obrazu
$w[2] — nazwaCechy	Np. wysokość
$w[3] — wartoscCechy	Np. 30
$w[4] — opis	Paragraf z opisem
$w[5] — rodzaj	Nagłówek h3, np. gorące źródło

MODUŁ 11 / PHP + MYSQL

Skrypt 4 — wyświetlanie [0]...[5]

Wklej poniższe instrukcje **zamiast komentarza** wewnątrz bloku **if (mysqli_num_rows($wynik) > 0)** z poprzedniej strony.

INSTRUKCJE ECHO

```
$w = mysqli_fetch_row($wynik);

echo "<img src='" . $w[0] . "' alt='" . $w[1] . "'>";
echo "<h2>" . $w[1] . "</h2>";
echo "<h3>" . $w[5] . "</h3>";
echo "<p>" . $w[2] . ": " . $w[3] . "</p>";
echo "<p>" . $w[4] . "</p>";
```

Uwaga — nie wklejaj drugi raz: W szablonie z poprzedniej strony **$w = mysqli_fetch_row($wynik)** już występuje. Użyj go tylko raz: potem wklej **pięć instrukcji echo**.

Dla obiektu numer 46 zobaczysz

Zdjęcie Strokkur.jpg, nagłówek „Strokkur”, nazwę rodzaju „gorące źródło”, parametr „wysokość: 30” oraz tekst opisowy z bazy.

Prostsze czytanie całego mechanizmu

- **if (isset(...))** — sprawdź, czy adres zawiera ID.
- **(int)** — potraktuj ID jako liczbę.
- **mysqli_query()** — znajdź obiekt w bazie.
- **mysqli_num_rows()** — sprawdź, czy jest taki rekord.
- **mysqli_fetch_row()** — pobierz dane do **$w**.
- **echo** — zbuduj stronę ze zdjęciem i opisem.

Dlaczego tu nie ma pętli while?

Zapytanie filtruje po jednym, unikalnym **idObiekt**. Spodziewamy się najwyżej jednego wiersza. Dlatego wystarcza **jeden** mysqli_fetch_row() zamiast while.

MODUŁ 12 / PHP + MYSQL

Pełny przykład: islandia.php (1/2)

Poniższy plik pokazuje wszystkie elementy wymagające obsługi bazy na stronie galerii. Druga część jest na kolejnej stronie. CSS dopracuj według wymagań arkusza.

PIERWSZA CZĘŚĆ PLIKU

```
<?php
mysqli_report(MYSQLI_REPORT_OFF);
$db = mysqli_connect("localhost", "root", "", "islandia");
if (!$db) die("Błąd połączenia");
mysqli_set_charset($db, "utf8mb4");
?>
<!DOCTYPE html>
<html lang="pl">
<head>
  <meta charset="UTF-8">
  <title>Islandia</title>
  <link rel="stylesheet" href="styl.css">
</head>
<body>
<header><h1><a href="islandia.php">Zwiedzaj Islandię</a></h1></header>
<aside>
  <h3>Do zwiedzania</h3>
  <ul><li>Wodospady:<ol>
  <?php
  $sql = "SELECT nazwa FROM obiekty
          WHERE panstwo='Islandia' AND idRodzaj=10";
  $wynik = mysqli_query($db, $sql);
  if (!$wynik) die("Błąd listy 1");
  while ($w = mysqli_fetch_row($wynik)) {
      echo "<li>" . $w[0] . "</li>";
  }
  ?>
  </ol></li>
```

Dalej: Nie zamykaj teraz **aside**, **body** ani połączenia. Wklej ciąg dalszy z kolejnej strony.

MODUŁ 13 / PHP + MYSQL

Pełny przykład: islandia.php (2/2)

Ten kod dopisz **bezpośrednio** po fragmencie z poprzedniej strony.

DRUGA CZĘŚĆ PLIKU

```
  <li>Siedliska zwierząt:<ol>
  <?php
  $sql = "SELECT nazwa FROM obiekty
          WHERE panstwo='Islandia' AND idRodzaj=14";
  $wynik = mysqli_query($db, $sql);
  if (!$wynik) die("Błąd listy 2");
  while ($w = mysqli_fetch_row($wynik)) {
      echo "<li>" . $w[0] . "</li>";
  }
  ?>
  </ol></li></ul>
</aside>
<main><h2>Galeria</h2><section>
<?php
$sql = "SELECT idObiekt, plik, nazwa FROM obiekty
        WHERE panstwo='Islandia'";
$wynik = mysqli_query($db, $sql);
if (!$wynik) die("Błąd galerii");
while ($w = mysqli_fetch_row($wynik)) {
    echo "<a href='obiekty.php?id=" . $w[0] . "'>";
    echo "<img class='miniatury' src='" . $w[1] . "'";
    echo " alt='" . $w[2] . "' title='" . $w[2] . "'>";
    echo "</a>";
}
?>
</section></main>
<footer><hr><p>Autor: 00000000000</p></footer>
<?php mysqli_close($db); ?>
</body></html>
```

Wstaw swój numer zdającego zamiast zer. Po otwarciu **http://localhost/islandia/islandia.php** powinny pojawić się dwie listy i 9 zdjęć.

MODUŁ 14 / PHP + MYSQL

Pełny przykład: obiekty.php (1/2)

Ta strona korzysta z tego samego połączenia. Najpierw odbiera GET i buduje opis. Na końcu dodamy listy boczne, by spełnić wymóg arkusza.

PIERWSZA CZĘŚĆ PLIKU

```
<?php
mysqli_report(MYSQLI_REPORT_OFF);
$db = mysqli_connect("localhost", "root", "", "islandia");
if (!$db) die("Błąd połączenia");
mysqli_set_charset($db, "utf8mb4");
?>
<!DOCTYPE html>
<html lang="pl">
<head>
  <meta charset="UTF-8">
  <title>Islandia</title>
  <link rel="stylesheet" href="styl.css">
</head>
<body>
<header><h1><a href="islandia.php">Zwiedzaj Islandię</a></h1></header>
<main><h2>Opis miejsca</h2><section>
<?php
if (isset($_GET["id"])) {
    $id = (int)$_GET["id"];
    $sql = "SELECT obiekty.plik, obiekty.nazwa,
            obiekty.nazwaCechy, obiekty.wartoscCechy,
            obiekty.opis, rodzaje.rodzaj FROM obiekty
            JOIN rodzaje ON obiekty.idRodzaj=rodzaje.idRodzaj
            WHERE obiekty.idObiekt=$id";
    $wynik = mysqli_query($db, $sql);
    if (!$wynik) die("Błąd zapytania");
    if (mysqli_num_rows($wynik) > 0) {
        $w = mysqli_fetch_row($wynik);
        echo "<img src='" . $w[0] . "' alt='" . $w[1] . "'>";
        echo "<h2>" . $w[1] . "</h2>";
        echo "<h3>" . $w[5] . "</h3>";
        echo "<p>" . $w[2] . ": " . $w[3] . "</p>";
        echo "<p>" . $w[4] . "</p>";
    } else { echo "Nie znaleziono obiektu."; }
} else { echo "Brak identyfikatora."; }
?>
</section></main>
```

MODUŁ 15 / PHP + MYSQL

Pełny przykład: obiekty.php (2/2)

Kod dopisz po zamknięciu **</main>** z poprzedniej strony. W arkuszu blok boczny jest wymagany na obu podstronach.

DRUGA CZĘŚĆ PLIKU

```
<aside>
  <h3>Do zwiedzania</h3>
  <ul>
    <li>Wodospady:<ol>
    <?php
    $sql = "SELECT nazwa FROM obiekty
            WHERE panstwo='Islandia' AND idRodzaj=10";
    $wynik = mysqli_query($db, $sql);
    if (!$wynik) die("Błąd listy 1");
    while ($w = mysqli_fetch_row($wynik)) {
        echo "<li>" . $w[0] . "</li>";
    }
    ?>
    </ol></li>
    <li>Siedliska zwierząt:<ol>
    <?php
    $sql = "SELECT nazwa FROM obiekty
            WHERE panstwo='Islandia' AND idRodzaj=14";
    $wynik = mysqli_query($db, $sql);
    if (!$wynik) die("Błąd listy 2");
    while ($w = mysqli_fetch_row($wynik)) {
        echo "<li>" . $w[0] . "</li>";
    }
    ?>
    </ol></li>
  </ul>
</aside>
<footer><hr><p>Autor: 00000000000</p></footer>
<?php mysqli_close($db); ?>
</body></html>
```

Po połączeniu obu części przetestuj adresy **obiekty.php?id=46** i **obiekty.php?id=53**. Sprawdź również adres bez parametru id.

MODUŁ 16 / PHP + MYSQL

Najczęstsze błędy — i jak je naprawić

Gdy wynik się nie pokazuje, sprawdzaj kolejno połączenie, SELECT, indeks kolumny i sposób wyświetlania.

Problem	Co sprawdzić
Nie ma połączenia	Czy działa MySQL? Czy baza nazywa się islandia?
Błąd zapytania SQL	Czy nazwy kolumn i tabel są zgodne z plikiem baza1.sql?
Pętla nie wypisuje danych	Czy warunek WHERE daje wyniki w phpMyAdmin?
Wyświetla złą kolumnę	Czy nie pomylono $w[0], $w[1], $w[2]?
Obraz się nie ładuje	Czy plik JPG znajduje się obok plików PHP? Czy src jest poprawny?
Nie działa strona szczegółów	Czy link przekazuje ?id=... i czy if(isset($_GET["id"]))?
Nie wyświetlają się polskie znaki	Sprawdź UTF-8 w HTML i mysqli_set_charset().

Przydatna metoda sprawdzania zapytań

TYLKO DO TESTÓW LOKALNYCH

```
$wynik = mysqli_query($db, $sql);
if (!$wynik) {
    die("Błąd SQL: " . mysqli_error($db));
}
echo "Liczba rekordów: " . mysqli_num_rows($wynik);
```

A jeśli w bazie wystąpią niebezpieczne znaki HTML?

WERSJA BEZPIECZNIEJSZA NIŻ echo $w[0]

```
echo htmlspecialchars($w[0], ENT_QUOTES, "UTF-8");
```

htmlspecialchars() zamienia znaki specjalne na zapis bezpieczny w HTML. W przykładach szkolnych pokazaliśmy najpierw zwykłe echo, by pętla i indeksy były czytelne. Gdy dane mogą pochodzić od użytkowników, stosuj htmlspecialchars, także w atrybutach zdjęć.

MODUŁ 17 / PHP + MYSQL

Ćwiczenia ucznia i gotowa ściąga

Najpierw spróbuj wykonać zadania samodzielnie, potem porównaj swój kod z przykładami w poradniku.

Trzy zadania

- **1. Łatwe:** SELECT nazwa dla idRodzaj=10. Wypisz wszystkie wodospady jako elementy **<li>** w **<ol>**.
- **2. Średnie:** SELECT idObiekt, plik, nazwa dla Islandii. Wypisz dla każdego obiektu **$w[0]**, **$w[1]** i **$w[2]**.
- **3. Egzaminacyjne:** Po kliknięciu galerii odbierz **id** przez **if (isset($_GET["id"]))** i pokaż opis obiektu z JOIN.

Ściąga: 6 komend, które trzeba znać

```
// 1. Połącz
$db = mysqli_connect("localhost", "root", "", "islandia");
if (!$db) die("Błąd połączenia");

// 2. Zapytaj
$wynik = mysqli_query($db, "SELECT nazwa FROM obiekty");
if (!$wynik) die("Błąd zapytania");

// 3. Pobierz wiele wierszy i wyświetl pierwszą kolumnę
while ($w = mysqli_fetch_row($wynik)) {
    echo "<li>" . $w[0] . "</li>";
}

// 4. Sprawdź GET
if (isset($_GET["id"])) {
    $id = (int)$_GET["id"];
}

// 5. Die = natychmiastowe zakończenie skryptu
// 6. Na końcu: mysqli_close($db);
```

Przed oddaniem egzaminu: Sprawdź dwie strony PHP, cztery zapytania w kwerendy.txt, zrzuty ekranu, styl.css, 9 zdjęć, plik przeglądarka.txt i zamknięcie połączenia. W pełnym arkuszu są także szczegółowe wymogi wyglądu i organizacji plików.

Źródło: arkusz INF.03-01-26.06-SG, s. 2 oraz 4–6, i plik baza1.sql z załączników. Objaśnienia o die(), kodowaniu i ochronie HTML są dodatkiem dydaktycznym.