

# Próba INF.03: pełny arkusz nr 2 — szkółka roślin

Czas: 2 × 45 min

praca samodzielna, warunki egzaminacyjne (bez internetu i notatek)

**Cel:** Samodzielne wykonanie w 90 minut kompletnego zadania egzaminacyjnego: baza danych z zapytaniami, przygotowanie grafiki, witryna HTML/CSS i skrypt PHP z mysql, a następnie samoocena według kryteriów i analiza błędów.

## Po zajęciach potrafisz:

- importuje bazę i zapisuje 4 zapytania SQL (SELECT z warunkiem, JOIN, GROUP BY, ALTER TABLE) wraz ze zrzutami
- skaluje i konwertuje grafikę zgodnie z poleceniem
- buduje witrynę z banerem, dwoma blokami i stopką dokładnie wg specyfikacji
- pisze skrypt PHP z mysql, który generuje formularz i wyniki z bazy
- ocenia swoją pracę wg kryteriów egzaminacyjnych i planuje poprawę słabych obszarów

## Sprzęt i materiały

- komputer z XAMPP, edytorem kodu (VS Code lub Notepad++), GIMP i przeglądarką
- folder `r2_30_arkusz` z Pliki\_do\_zadan:  
`szkolka.sql`, `ogrod.png`, `tekst.txt`

## Oprogramowanie

- **XAMPP** — Apache, MariaDB, phpMyAdmin  
[apachefriends.org/download.html](http://apachefriends.org/download.html)
- **GIMP** — Obraz → Skaluj obraz; Plik → Wyeksportuj jako  
[gimp.org/downloads](http://gimp.org/downloads)
- **Visual Studio Code lub Notepad++** — na egzaminie edytor bez odpowiedzi AI  
[code.visualstudio.com](http://code.visualstudio.com)

## Przygotowanie przed zajęciami

- Przejrzyj kryteria oceny arkuszy INF.03 i swoje błędy z zadania 19 (pierwszej próby).
- Na pulpicie utwórz folder o nazwie swojego nazwiska i imienia — wszystkie wyniki zapisuj w nim.

## Bezpieczeństwo

- Pracuj samodzielnie, bez internetu i asystentów AI — wynik ma pokazać Twój rzeczywisty poziom przed egzaminem.
- Co kilka minut zapisuj pliki (Ctrl+S); na egzaminie nie ma możliwości odzyskania utraconej pracy.

## Przebieg zadania

### 1 Część 1. Baza danych (ok. 20 min)

Tabela `rodzaje`: id, nazwa. Tabela `rosliny`: id, nazwa, rodzaj\_id (klucz obcy do `rodzaje`), cena, wysokosc\_cm, kwitnienie, dostepne (liczba sztuk w magazynie).

1. Zaimportuj plik `szkolka.sql` (tworzy bazę `szkolka`). Zrób zrzut struktury bazy w phpMyAdmin (np. Widok relacyjny / Projektant) i zapisz jako `baza.png`.
2. Zapisz w pliku `kwerendy.txt` i wykonaj zapytania; zrzut wyniku każdego zapisz jako `kw1.png` ... `kw4.png`.
3. Zapytanie 1: wybiera nazwę i cenę roślin o rodzaju z id 2, posortowane malejąco według ceny.
4. Zapytanie 2: wybiera nazwę rośliny, nazwę jej rodzaju (z tabeli `rodzaje`) i cenę dla roślin dostępnych w magazynie (`dostepne` większe od 0) i tańszych niż 20 zł.
5. Zapytanie 3: dla każdego rodzaju wyświetla jego nazwę, liczbę roślin i średnią cenę zaokrągloną do 2 miejsc, z aliasami `liczba` i `srednia_cena`.
6. Zapytanie 4: dodaje do tabeli `rosliny` kolumnę `promocja` typu logicznego z wartością domyślną `falsz`, niepustą.

### 2 Część 2. Grafika (ok. 5 min)

1. Plik `ogrod.png` przeskaluj do szerokości 400 px z zachowaniem proporcji i zapisz w formacie JPEG jako `ogrod.jpg` (jakość 80).

### 3 Część 3. Witryna i styl (ok. 35 min)

Plik `szkolka.php`, tytuł karty „Szkółka roślin”, język polski, kodowanie UTF-8, arkusz `styl.css`. Teksty weź z pliku `tekst.txt`. Układ: baner na całą szerokość, pod nim blok lewy (30%) i blok prawy (70%), na dole stopka.

1. Baner: obraz `ogrod.jpg` z tekstem alternatywnym „Ogród z kwitnącymi roślinami”, nagłówek h1 z `tekst.txt`.
2. Blok lewy: nagłówek h2 i lista punktowana 5 odnośników z `tekst.txt`; regulamin otwiera się w nowej karcie.
3. Blok prawy: nagłówek h2, akapit z `tekst.txt` i efekt skryptu (część 4). Stopka: akapit „Stronę wykonał: ” i Twoje imię i nazwisko.
4. Styl: krój Verdana dla wszystkich elementów; baner tło `#2E7D32`, biały tekst, wyśrodkowany; obraz w banerze pływa do lewej, wysokość 120 px, zaokrąglenie narożników 10 px; h1 rozmiar 40 px.
5. Styl: blok lewy tło `#A5D6A7`, prawy `#E8F5E9`, oba wysokość 600 px i marginesy wewnętrzne lewy i prawy 20 px; stopka tło `#1B5E20`, biały tekst, wyrównanie do prawej, marginesy wewnętrzne 20 px, czyszczenie opływania.
6. Styl: odnośniki pogrubione w kolorze `#1B5E20`, po najechaniu `#E65100`; blok wyniku `.roslina` — białe tło, lewe obramowanie 6 px ciągłe `#2E7D32`, marginesy zewnętrzne 10 px u góry i u dołu.

#### 4 Część 4. Skrypt PHP (ok. 25 min)

Skrypt w bloku prawym łączy się z bazą `szkołka` (localhost, root, bez hasła), na końcu zamyka połączenie.

1. Skrypt 1: wypisuje formularz wysyłany metodą GET do `szkołka.php` z listą rozwijaną `rodzaj` — opcje generowane z tabeli `rodzaje` (wartość = id, tekst = nazwa) — i przyciskiem „Pokaż”.
2. Skrypt 2: jeśli formularz został wysłany, wybiera z bazy nazwę, cenę i okres kwitnienia roślin wybranego rodzaju dostępnych w magazynie, posortowane alfabetycznie. Każdą roślinę wypisuje w bloku `div` klasy `roslina`: nagłówek `h3` z nazwą i akapit „Cena: X zł, kwitnie: Y”. Wartość z formularza przekaz bezpieczniej (rzutowanie na `int` lub `mysqli_prepare`).
3. Gdy rodzaj nie ma dostępnych roślin, skrypt wypisuje akapit „Brak dostępnych roślin tego rodzaju.”. Sprawdź działanie dla krzewów (Bez lilak ma 0 sztuk) i zrób zrzut strony `witryna.png`.

#### 5 Samoocena i analiza błędów (po czasie egzaminu)

1. Po 90 minutach przestań pisać. Otrzymasz kryteria oceniania — oceń swoją pracę punkt po punkcie i wpisz wynik w tabeli samooceny.
2. Porównaj wynik z zadaniem 19: wypisz 3 błędy lub braki, które powtórzyły się, i plan ich poprawy przed egzaminem.

### W sprawozdaniu umieść

- ☐ folder z plikami: `kwerendy.txt`, `baza.png`, `kw1.png` – `kw4.png`, `ogrod.jpg`, `szkołka.php`, `styl.css`, `witryna.png`
- ☐ tabela samooceny wg kryteriów i porównanie z zadaniem 19
- ☐ lista 3 powtarzających się błędów z planem poprawy

### Pytania kontrolne

1. Dlaczego w zapytaniu 2 potrzebne jest złączenie tabel i co by się stało bez warunku ON?
2. Czym różni się `mysqli_fetch_assoc` od `mysqli_fetch_row`? Którą funkcję wybrałeś i dlaczego?
3. Dlaczego wartości z `$_GET` nie wolno wklejać bezpośrednio do zapytania SQL?
4. Która część arkusza zajęła Ci najwięcej czasu i jak to zmienisz na egzaminie?

### Kryteria oceny

| Kryterium                                                                                                                         | Punkty    |
|-----------------------------------------------------------------------------------------------------------------------------------|-----------|
| Zapytania 1–4 poprawne, zapisane w <code>kwerendy.txt</code> , zrzuty <code>kw1</code> – <code>kw4</code> i <code>baza.png</code> | 5         |
| Grafika: 400 px szerokości, proporcje, JPEG                                                                                       | 1         |
| Struktura HTML: tytuł, lang, baner, bloki, lista z 5 odnośnikami ( <code>target _blank</code> ), stopka                           | 4         |
| Styl CSS zgodny ze specyfikacją (kolory, układ, rozmiary, hover, klasa <code>roslina</code> )                                     | 4         |
| Skrypt 1: połączenie, lista opcji z bazy, formularz GET, zamknięcie połączenia                                                    | 2         |
| Skrypt 2: zapytanie z warunkiem i sortowaniem, bezpieczny parametr, wynik w <code>div.roslina</code> , komunikat o braku          | 2         |
| Samoocena i analiza błędów                                                                                                        | 2         |
| <b>Razem</b>                                                                                                                      | <b>20</b> |

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