

Obsługa MySQL w PHP – biblioteka mysqli i PDO

W celu połączenia witryny internetowej z bazą danych MySQL z użyciem PHP możemy skorzystać z jednego z trzech dostępnych w języku PHP rozszerzeń (czyli bibliotek). W świecie programistów takie rozszerzenie często nazywamy **interfejsem API** – czyli zestawem funkcji lub klas, które pozwalają na komunikację z bazą danych.

Interfejs API może mieć charakter **proceduralny** (funkcje) lub **obiektowy** (klasy i metody). W zależności od wybranej biblioteki, różni się nieco sposób korzystania z niej w kodzie PHP.

- **Proceduralny interfejs API** – każda operacja (np. połączenie z bazą, wykonanie zapytania) realizowana jest przez wywołanie osobnej funkcji, np. `mysqli_connect()`, `mysqli_query()`.
- **Obiektowy interfejs API** – komunikacja z bazą odbywa się poprzez obiekty, np. obiekt klasy `PDO` lub `mysqli` (w wersji obiektowej), a operacje wykonujemy metodami tego obiektu.

Dostępne interfejsy PHP do obsługi MySQL

1. **mysql** – stare rozszerzenie, istniejące od PHP 2.0. Nie zalecane do użycia. Od PHP 5.5 uznane za przestarzałe (ang. *deprecated*), a od PHP 7.0 całkowicie usunięte. Interfejs wyłącznie proceduralny. **Nie używamy go w nowych projektach!**
2. **mysqli** – nowoczesna, poprawiona wersja poprzedniego interfejsu (stąd „i” jak *improved*). Obsługuje zarówno styl proceduralny, jak i obiektowy. Aktywnie rozwijana i zalecana do stosowania w nowych projektach.
3. **PDO (PHP Data Objects)** – w pełni obiektowy interfejs API, który obsługuje wiele różnych silników baz danych (nie tylko MySQL!). To bardzo elastyczne rozwiązanie, szczególnie przy pracy z większymi aplikacjami lub migracji między bazami.

Porównanie interfejsów API

Rozszerzenie (API)	mysql	mysqli	PDO
Wprowadzono w wersji PHP	2.0	5.0	5.0
Wsparcie dla rozszerzenia	usunięte (od PHP 7.0)	aktywnie wspierane	aktywnie wspierane
Zalecane dla nowych projektów	nie	tak	tak
Interfejs proceduralny	tak	tak	nie
Interfejs obiektowy	nie	tak	tak
Obsługa innych silników SQL	nie	nie	tak

Podsumowanie

W obecnych czasach **nie powinniśmy** już używać rozszerzenia

`mysql` . Zamiast niego sięgamy po `mysqli` (gdy piszemy aplikację ściśle dla MySQL/MariaDB) albo po `PDO` , jeśli zależy nam na wielosilnikowości i piszemy aplikację bardziej uniwersalną.

Dla uczniów technikum – na egzaminie INF.03 wymagane jest dobre opanowanie mysqli – najlepiej zarówno w stylu proceduralnym, jak i obiektowym.