

PHP I - zajęcia 10 – łączenie się z Bazą MySQL za pomocą PHP

1. Ustanawianie połączenia z bazą danych

- ★ `mysqli('nazwa_serwera','nazwa_uzytkownika','haslo','nazwa_bazy')` - tworzy egzemplarz klasy `mysqli` (`mysqli` jest biblioteką PHP odpowiedzialną za połączenie z bazą danych, posiada zarówno interfejs proceduralny jak i obiektowy. My korzystać będziemy z obiektowego)
 - Np.. `@$db = new mysqli('localhost','admin','admin1','Ksiegarnia');`
 - `@` - użyto w celu tłumienia błędów.
- ★ `mysqli_connect_error()` - zwraca numer błędu przy połączeniu z bazą danych lub 0 gdy połączenie zostało nawiązane.

2. Polecenia przygotowane

- ★ Pozwalają one na przyspieszenie realizacji zapytań w przypadkach, gdy kilkakrotnie należy wykonać to samo zapytanie, ale z różnymi danymi.
- ★ Zapytanie do bazy danych zapisujemy składnią MySQL np.:
 - `$zapytanie = "SELECT ISBN, Autor, Tytuł, Cena FROM Ksiegarnia WHERE $wartosc_szukana=?"`
 - `?` - pełni rolę zamiennika i przekazuje bazie danych informację "cokolwiek zostanie umieszczone w miejscu znaku zapytania potraktuj jako dane a nie jako kod". Chroni to naszą bazę i stronę przed tak zwanym sql injection czyli wstrzyknięciem złośliwego kodu po przez formularz. Nie jest jednak rozwiązaniem optymalnym, gdyż nie może być używane do nazw kolumn, tabel czy całych baz danych.
- ★ `prepare()` - metoda służy do utworzenia obiektu, polecenia, lub zasobu, który będzie następnie wykorzystany do wykonania zapytania oraz przetworzenia wyników np.:
 - `$polecenie=$db->prepare($zapytanie);`
- ★ `bind_param()` - metoda służy do określenia wartości które mają zostać w stawione w miejscu znaków zapytania. Pierwszym parametrem jest łańcuch znaków określający format (s-string, i-integer, d-double, b – BLOB, czyli duży obiekt binarny). Po tym parametrze należy podać zmienne, przy czym ich liczba powinna odpowiadać liczbie pytańników w zapytaniu. Wartości te będą podawane w odpowiedniej kolejności. Np.:
 - `$polecenie->bind_param('s',$wartosc_podana');`
 - *Patrz zapytanie do bazy
- ★ `execute()` - metoda powoduje wykonanie zapytania np..
 - `$polecenie->execute();`

3. Odczytywanie rezultatów zapytań

- ★ `bind_result()` - metoda służy do podawania listy zmiennych w których będą zapisywane poszczególne kolumny zwróconych rekordów np..
 - `$polecenie->bind_result($isbn,$autor,$tytul,$cena);`
- ★ `fetch()` - metoda powoduje ze kolumny z następnego wiersza ze zwróconego zbioru zostaną zapisane w powiązanych zmiennych. Użyta w pętli po `bind_result()` spowoduje pobranie kolejnego wiersza wyników i zapisanie umieszczonych w nim wartości do powiązanych zmiennych.
 - `while($polecenie->fetch()){ //kod// }`
- ★ `store_result()` - pobiera i zapisuje w buforze wszystkie zapisane wyniki np..
 - `$polecenie->store_result();`

- Liczba zwróconych wierszy jest przechowywana w składowej num_rows.
Można ją wyświetlić używając echo \$polecenie->num_rows

4. Zamykanie połączenia z baza danych

- ★ free_result() - zwalnia zestaw wyników np..
 - \$polecenie->free_result();
- ★ close() - zamyka połączenie z bazą danych np.
 - \$db->close();

5. Ćwiczenie 1

- ★ *Utwórz formularz pozwalający na wyszukiwanie haseł w bazie księgarni na podstawie tytułu, autora oraz ISBN (lista rozwijana)*
- ★ *Utwórz skrypt rezultaty.php który:*
 - *Sprawdza, czy została wybrana metoda wyszukiwania (z listy rozwijanej)*
 - *Sprawdza czy pole z treścią wyszukiwania nie jest puste*
 - *Nawiązuje połączenie z bazą danych i obsługuje błąd połączenia*
 - *Wyświetla ilość znalezionych rekordów*
 - *Wypisuje znalezione rekordy w formie:*
 - *Tytuł:*
 - *Autor:*
 - *ISBN:*
 - *Cena:*

6. Wstawianie nowych danych do bazy

- ★ Wstawianie nowych danych do bazy przebiega podobnie do pobierania informacji z niej. Konieczne jest nawiązanie połączenia, wysłanie zapytania i odczyt rezultatów. W tym wypadku zapytanie zamiast słowa SELECT będzie zawierało słowo INSERT np..
 - \$zapytanie = "INSERT INTO Ksiazki VALUES (?, ?, ?, ?)";
 - \$polecenie = \$db->prepare(\$zapytanie);
 - \$polecenie->bind_param('sssd', \$isbn, \$autor, \$tytul, \$cena);
 - \$polecenie->execute();
- ★ affected_rows() - metoda sprawdza ile rekordów zostało poprawnie przetworzonych.
 - If(\$polecenie->affected_rows()>0){//kod//}

7. Ćwiczenie 2

- ★ *Utwórz formularz pozwalający na wstawianie do bazy danych nowych książek (autor, tytuł, ISBN, Cena)*
- ★ *Utwórz skrypt wstaw_ksiazke.php który:*
 - *Sprawdza czy pole z treścią nie są puste*
 - *By zapewnić poprawność wstawiania ceny jako wartości zmiennoprzecinkowej możemy użyć funkcji doubleval(\$cena)*
 - *Nawiązuje połączenie z baza danych i obsługuje błąd połączenia*
 - *Wyświetla informacje o dodaniu lub nie książki do bazy*