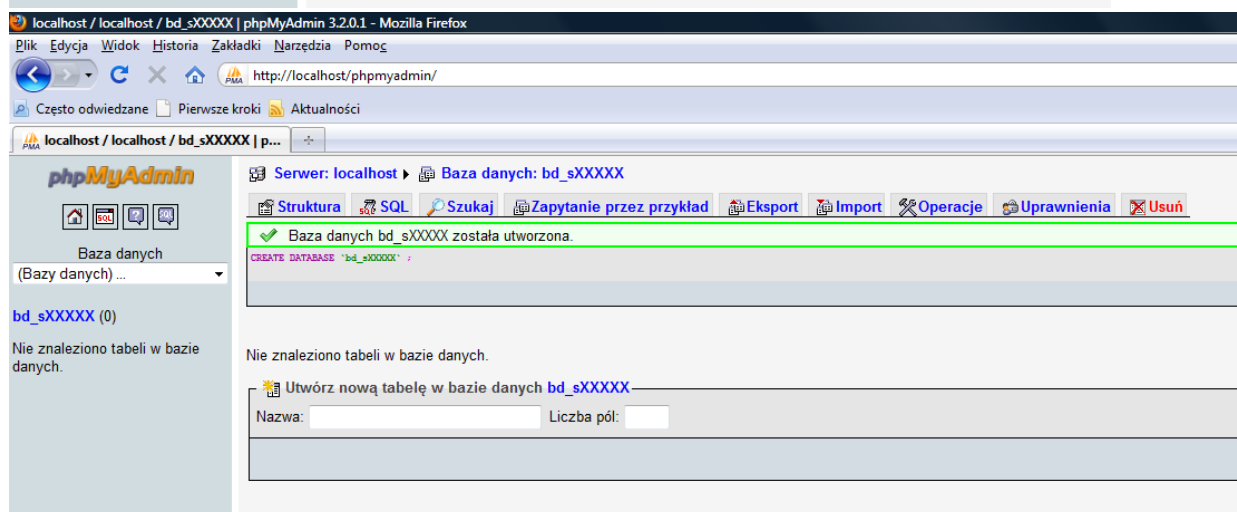
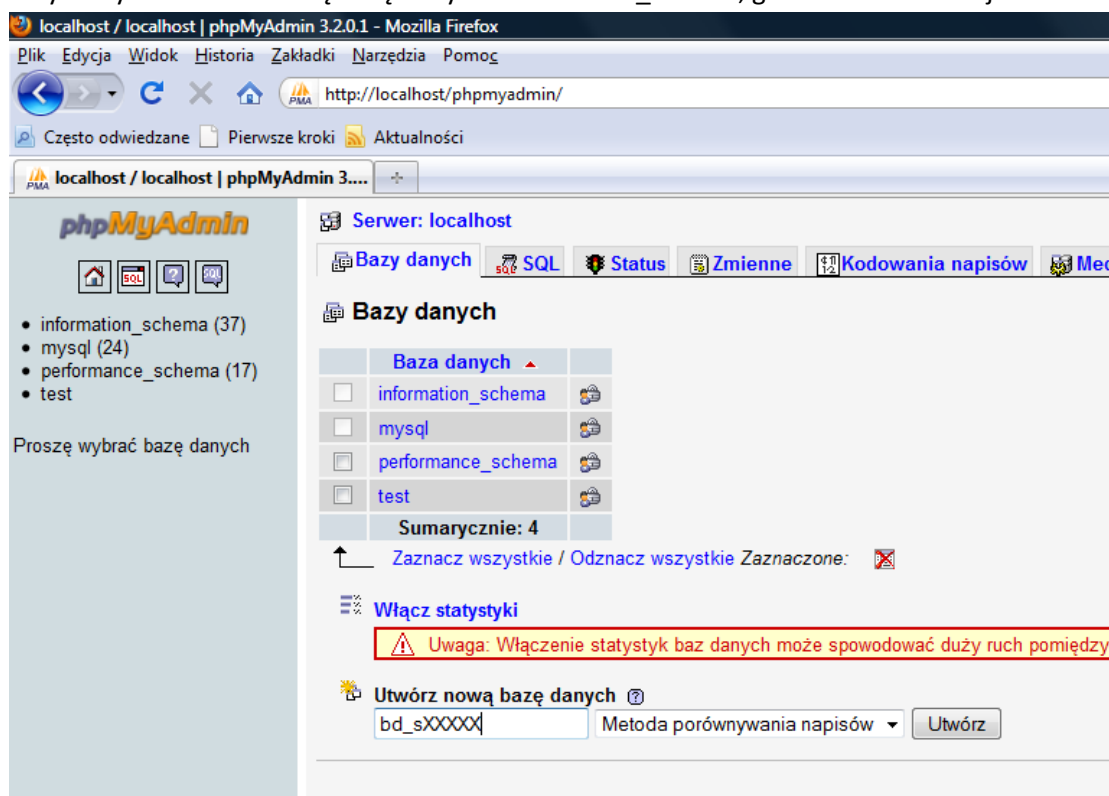


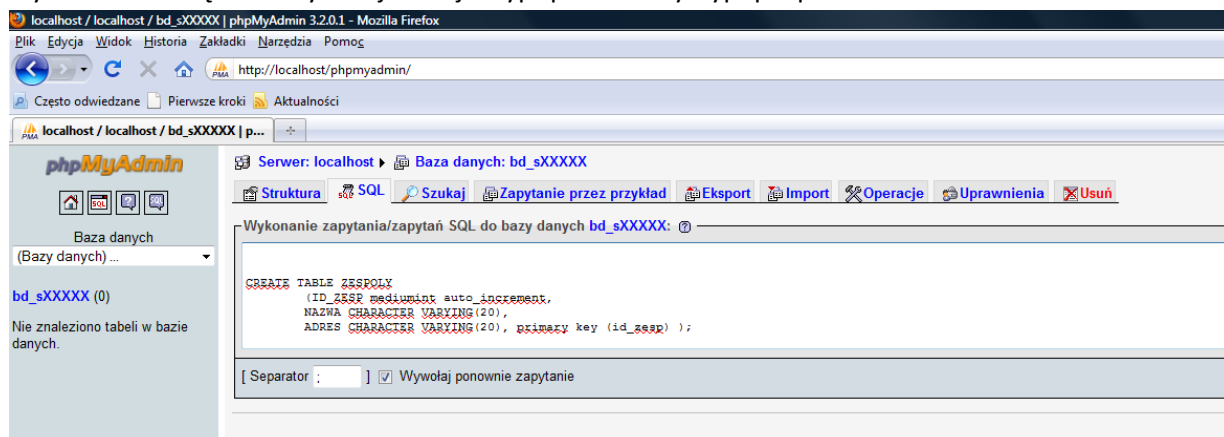
Swój kod umieszczaj w katalogu `c:\wamp\www`.

1. Zdefiniuj zmienną `$imie` i przypisz jej wartość ze swoim imieniem. Następnie wypisz w stronie wynikowej `Nazywam się <tutaj wartość ze zmiennej>`.
2. Zmodyfikuj poprzednie ćwiczenie tak, aby wykorzystać instrukcję warunkową `if`. Porównaj zawartość zmiennej z Twoim imieniem. Jeżeli jedno zgadza się z drugim wypisz komunikat jak poprzednio. W przeciwnym wypadku wypisz komunikat: `To nie moje imie`.
3. Wykorzystując jedynie jedną zmienną i operatory łączące operację arytmetyczną i przypisanie (np. `+=`) spróbuj uzyskać poniższy wynik:
`Value is now 8.`
`Add 2. Value is now 10.`
`Subtract 4. Value is now 6.`
`Multiply by 5. Value is now 30.`
`Divide by 3. Value is now 10.`
`Increment value by one. Value is now 11.`
`Decrement value by one. Value is now 10.`
4. Użyj pętli `while` tak aby wypisać na stronie wynikowej liczby od 1 do 10.
5. Zmodyfikuj poprzednie ćwiczenie tak, aby użyć pętli `for`.
6. Zmodyfikuj poprzednie ćwiczenie tak, aby pierwsza i ostatnia wypisywana wartość była podawana jako parametr strony metodą GET.
7. Zmodyfikuj poprzednie ćwiczenie tak, aby każda kolejna wypisywana wartość była innego (coraz większego) rozmiaru. Do zmiany wielkości tekstu na stronie użyj taga `<font>` i jego atrybutu `size`. Przetestuj skrypt dla rozmiarów od 1 do 5.
8. Napisz prosty kalkulator. Skrypt powinien przyjmować trzy parametry (np. `x`, `y` i `op`) metodą GET, a następnie w zależności od parametru `op` wykonywał jedną z 4 podstawowych operacji arytmetycznych na parametrach `x` i `y`. Na stronie wynikowej powinien pojawiać się tekst: `<wartość x> <op> <wartość y> = <wynik>`.
9. Stwórz osobną stronę w HTML (nie trzeba PHP), która zawiera pobiera od użytkownika dane i przekazuje je do kalkulatora utworzonego w poprzednim ćwiczeniu.
10. Połącz strony utworzone w poprzednim ćwiczeniu w jedną. Tzn. jeżeli wszystkie parametry nie zostały przekazane poprawnie, powinien być wyświetlany formularz. W przeciwnym wypadku wynik operacji matematycznej.
11. Utwórz tablicę zawierającą nazwy kilku stolic krajów europejskich. Wypisz wszystkie elementy tej tablicy za pomocą pętli `foreach`. Posortuj tablicę za pomocą funkcji `sort`: <http://php.net/manual/en/function.sort.php>. Wypisz jej zawartość jeszcze raz. Dodaj kilka elementów do tej tablicy, wypisz, posortuj i wypisz jeszcze raz.
12. Utwórz tablicę asocjacyjną, w której kluczem jest stolica a wartością nazwa kraju, przykładowo: Tokyo, Japan; Mexico City, Mexico; New York City, USA; Mumbai, India; Seoul, Korea; Shanghai, China; Lagos, Nigeria; Buenos Aires, Argentina; Cairo, Egypt; London, England. Utwórz formularz który pozwala użytkownikowi wybrać jeden z krajów (możesz tutaj wykorzystać funkcję `array_keys` http://www.w3schools.com/php/func_array_keys.asp). Po wybraniu kraju użytkownikowi powinna pojawić się informacja `Stolicą <kraj> jest <miasto>`.
13. Przygotuj tablicę z kilkoma wartościami liczbowymi. Napisz skrypt, który znajdzie wartość największą, najmniejszą i średnią.
14. Napisz prostą funkcję wypisującą na ekranie „Hello World”. Wywołaj funkcję w pętli 5 razy.
15. Napisz funkcję obliczającą pole prostokąta na podstawie długości jego boków. Funkcja powinna przyjmować dwa parametry (długości boków oczywiście) a zwracać pole. Wypróbuj funkcję dla dwóch zestawów parametrów.
16. Stwórz formularz pozwalający na podawanie długości boków prostokąta. Zmodyfikuj skrypt z funkcją tak, aby parametry aktualne podawane do funkcji były pobierane z formularza.
17. Zdefiniuj funkcję rekurencyjną obliczającą silnię. Wypróbuj funkcję.

18. Zaimplementuj licznik odwiedzeń do strony z wykorzystaniem mechanizmu sesji.
19. Zaimplementuj mechanizm logowania z wykorzystaniem mechanizmu sesji.
20. Wywołaj phpMyAdmina poprzez wpisanie do przeglądarki <http://localhost/phpmyadmin/>. Wybierz zakładkę Bazy Danych i utwórz nową bazę danych o nazwie bd_sXXXXX, gdzie XXXXX to Twój numer indeksu.



21. Wybierz zakładkę SQL i wykonaj w niej skrypt pracownicy.php.sql.



22. Odczytaj nazwiska i place wszystkich pracowników. W poniżej przedstawionej metodzie wykonywania zapytań, w wyniku metody query dostajemy obiekt reprezentujący wynik. Każde wykonanie metody fetch

pobiera kolejny wiersz wyniku w postaci tablicy. Każda kolumna w wyniku jest osiągalna z tej tablicy albo poprzez nazwę albo poprzez numer kolumny liczony od zera.

```
<?php
try {
    $dbh = new PDO('mysql:host=localhost;dbname=bd_sXXXXXX','root');
    $dbh->setAttribute(PDO::ATTR_ERRMODE, PDO::ERRMODE_EXCEPTION);

    $result=$dbh->query('SELECT nazwisko,placa_pod FROM pracownicy');
    $liczba=$result->columnCount();
    echo "Liczba kolumn w wyniku: $liczba <br/>" ;

    while ($row=$result->fetch())
    {
        echo $row['nazwisko'];
        echo $row[1];
        echo "<br/>";
    }
    $result=null;
    $dbh = null;
} catch (PDOException $e) {
    print "Error!: " . $e->getMessage() . "<br/>";
    die();
}
?>
```

23. Napisz skrypt pobierający z bazy danych pracowników z zespołu o podanym numerze. Numer powinien być przekazywany jako parametr metodą GET.
24. Napisz skrypt generujący formularz pozwalający wybrać jeden z zespołów. Lista zespołów powinna być brana z bazy danych. Formularz powinien odwoływać się do efektu poprzedniego ćwiczenia. HINT: Do stworzenia formularza użyj elementów <form>, <select> i <option>. W tym ostatnim przypadku w atrybucie value wpisz identyfikator zespołu, a jako zawartość elementu wpisz nazwę zespołu.
25. Napisz skrypt wstawiający nowego pracownika.

```
<?php
try {
    $dbh = new PDO('mysql:host=localhost;dbname=bd_sXXXXXX','root');
    $dbh->setAttribute(PDO::ATTR_ERRMODE, PDO::ERRMODE_EXCEPTION);

    $liczba=$dbh->exec('insert into pracownicy(nazwisko, placa_pod) values
(\ 'KOWALEWSKI\ ',1000) ');

    echo "Liczba wstawionych wierszy: $liczba <br/>" ;

    $liczba=null;
    $dbh = null;
} catch (PDOException $e) {
    print "Error!: " . $e->getMessage() . "<br/>";
    die();
}
?>
```

26. Napisz formularz, który pozwala wstawić nowego pracownika.
27. Zmodyfikuj mechanizm logowania z któregoś z poprzednich ćwiczeń tak, aby identyfikator użytkownika i hasło były pobierane z bazy danych. Utwórz wymagane tabelki i wypełnij je samodzielnie.
28. Zapoznaj się z mechanizmem przygotowywania poleceń SQL i ich wykonywania: <http://www.kitebird.com/articles/php-pdo.html> (rozdział „Using prepared statements”). Powtórz powyższe ćwiczenia dot. baz danych ale przy pomocy poleceń przygotowanych. Jakże są zalety takich poleceń?