

Zadanie egzaminacyjne

Wykonaj aplikację internetową dla salonu pielęgnacji psów i kotów. Wykorzystaj pakiet XAMPP jako środowisko bazodanowo – aplikacyjne. Stronę internetową zbuduj przy wykorzystaniu edytora zaznaczającego składnię.

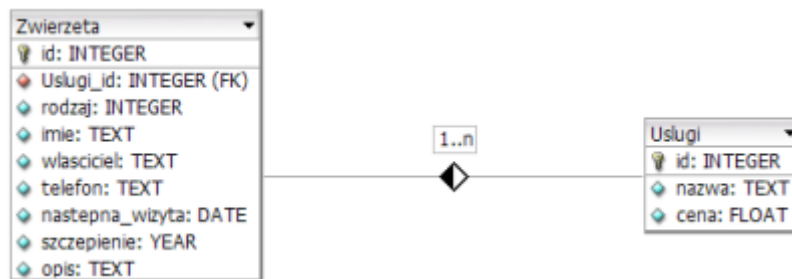
Aby wykonać zadanie zaloguj się na konto **Egzamin** bez hasła. Na pulpicie znajdziesz archiwum ZIP o nazwie *dane2.zip*, zabezpieczone hasłem: **Zwi3rz3ta**

Plik należy rozpakować. Zawartość pliku *zwierzeta.sql* poniżej w ramce

Wyniki swojej pracy zapisz w folderze utworzonym na pulpicie konta **Egzamin**. Jako nazwy folderu użyj swojego numeru PESEL. Rozpakowane pliki umieść w tym folderze.

Operacje na bazie danych

Na obrazie 1 przedstawiono fragment bazy Salon. Pole rodzaj z tabeli Zwierzeta określa rodzaj zwierzęcia i przyjmuje wartość 1 dla psa lub wartość 2 dla kota. Pole Usługi_id tabeli Zwierzeta jest kluczem obcym wiążącym do klucza głównego tabeli Usługi.



Obraz 1. Fragment bazy Salon

Uruchom usługi MySQL i Apache z XAMPP Control Panel i przejdź do narzędzia phpMyAdmin. Następnie wykonaj operacje na bazie danych:

- Utwórz nową bazę danych o nazwie Salon
- Do bazy Salon zaimportuj tabele z pliku *zwierzeta.sql* z rozpakowanego archiwum

```
• -- phpMyAdmin SQL Dump
• -- version 4.5.1
• -- http://www.phpmyadmin.net
• --
• -- Host: 127.0.0.1
• -- Czas generowania: 04 Kwi 2016, 10:24
• -- Wersja serwera: 10.1.9-MariaDB
• -- Wersja PHP: 5.6.15
•
• SET SQL_MODE = "NO_AUTO_VALUE_ON_ZERO";
• SET time_zone = "+00:00";
•
• /*!40101 SET @OLD_CHARACTER_SET_CLIENT=@@CHARACTER_SET_CLIENT
*/;
• /*!40101 SET
@OLD_CHARACTER_SET_RESULTS=@@CHARACTER_SET_RESULTS */;
```

```

• /*!40101 SET
  • @OLD_COLLATION_CONNECTION=@@COLLATION_CONNECTION */;
• /*!40101 SET NAMES utf8mb4 */;
•
• --
• -- Baza danych: `wet2`
• --
•
• -----
•
• --
• -- Struktura tabeli dla tabeli `uslugi`
• --
•
• CREATE TABLE `uslugi` (
•   `id` int(10) UNSIGNED NOT NULL,
•   `nazwa` text,
•   `cena` float DEFAULT NULL
• ) ENGINE=InnoDB DEFAULT CHARSET=latin1;
•
• --
• -- Zrzut danych tabeli `uslugi`
• --
•
• INSERT INTO `uslugi` (`id`, `nazwa`, `cena`) VALUES
• (1, 'pazury', 30),
• (2, 'mycie', 20),
• (3, 'czesanie', 10),
• (4, 'uszy', 30);
•
• -----
•
• --
• -- Struktura tabeli dla tabeli `zwierzeta`
• --
•
• CREATE TABLE `zwierzeta` (
•   `id` int(10) UNSIGNED NOT NULL,
•   `usluga_id` int(11) NOT NULL,
•   `rodzaj` int(10) UNSIGNED DEFAULT NULL,
•   `imie` text,
•   `wlasciciel` text,
•   `telefon` text,
•   `nastepna_wizyta` date DEFAULT NULL,
•   `szczepienie` year(4) DEFAULT NULL,
•   `opis` text
• ) ENGINE=InnoDB DEFAULT CHARSET=latin1;
•
• --
• -- Zrzut danych tabeli `zwierzeta`
• --
•
• INSERT INTO `zwierzeta` (`id`, `usluga_id`, `rodzaj`, `imie`, `wlasciciel`, `telefon`,
  • `nastepna_wizyta`, `szczepienie`, `opis`) VALUES
• (1, 3, 1, 'Fafik', 'Adam Kowalski', '111222333', '2017-06-30', 2016, 'problemy z uszami'),
• (2, 2, 1, 'Brutus', 'Anna Wysocka', '222333444', '2017-06-26', 2016, 'zapalenie krtani'),
• (4, 1, 1, 'Saba', 'Monika Nowak', '333444555', NULL, 2015, 'kropiówka'),
• (5, 0, 1, 'Alma', 'Jan Kowalewski', '444555666', '2017-07-03', NULL, 'antybiotyki'),
• (6, 4, 2, 'Figaro', 'Anna Kowalska', '555666777', NULL, NULL, 'problemy z uszami'),

```

```

• (7, 0, 2, 'Dika', 'Katarzyna Kowal', '666777888', '2017-06-30', 2016, 'operacja'),
• (8, 2, 2, 'Fuks', 'Jan Nowak', '888999111', '2017-07-04', 2016, 'antybiotyk');
•
• --
• -- Indeksy dla zrzutów tabel
• --
• --
• --
• -- Indexes for table `uslugi`
• --
• ALTER TABLE `uslugi`
• ADD PRIMARY KEY (`id`);
•
• --
• -- Indexes for table `zwierzeta`
• --
• ALTER TABLE `zwierzeta`
• ADD PRIMARY KEY (`id`);
•
• --
• -- AUTO_INCREMENT for dumped tables
• --
• --
• -- AUTO_INCREMENT dla tabeli `uslugi`
• --
• ALTER TABLE `uslugi`
• MODIFY `id` int(10) UNSIGNED NOT NULL AUTO_INCREMENT,
• AUTO_INCREMENT=6;
•
• --
• -- AUTO_INCREMENT dla tabeli `zwierzeta`
• --
• ALTER TABLE `zwierzeta`
• MODIFY `id` int(10) UNSIGNED NOT NULL AUTO_INCREMENT,
• AUTO_INCREMENT=9;
• /*!40101 SET CHARACTER_SET_CLIENT=@OLD_CHARACTER_SET_CLIENT */;
• /*!40101 SET CHARACTER_SET_RESULTS=@OLD_CHARACTER_SET_RESULTS
• */;
• /*!40101 SET COLLATION_CONNECTION=@OLD_COLLATION_CONNECTION */;

```

Powyżej Zawartość pliku zwierzeta.sql

- Wykonaj zrzut ekranu po imporcie. Zrzut zapisz w folderze z Twoim numerem PESEL w formacie PNG i nazwij *import-salon.png*. Na zrzucie powinny być widoczne elementy wskazujące na poprawnie wykonany import bazy. Nie skaluj, ani nie przycinaj obrazu
- Utwórz następujące zapytania SQL do bazy Salon i sprawdź poprawność ich działania:
 - Zapytanie 1: wybierające jedynie pola nazwa i cena z tabeli Usługi

- Zapytanie 2: wybierające jedynie pola imie, rodzaj, nastepna_wizyta, telefon z tabeli Zwierzeta dla tych rekordów, dla których nastepna_wizyta jest różna od 0
- Zapytanie 3: korzystające z relacji i wybierające jedynie pola rodzaj z tabeli Zwierzeta oraz odpowiadające im pola nazwa z tabeli Usługi
- Zapytanie 4: zwracające minimalną cenę spośród wszystkich usług zapisanych w tabeli Usługi
- Utworzone zapytania zapisz w folderze z Twoim numerem PESEL, w pliku *kwerendy.txt*. Zapytania ponumeruj stosując format zapisu: „zapytanie 1: ... treść zapytania ...” Wykonaj zrzuty ekranu przedstawiające wyniki działania kwerend. Zrzuty zapisz w formacie PNG i nadaj im nazwy *kw1*, *kw2*, *kw3*, *kw4*. Zrzuty powinny być czytelne i wykonane bez skalowania i kadrowania, a także z widocznym paskiem zadań i godziną ich wykonania.

Witryna internetowa



Obraz 2. Witryna internetowa

Przygotowanie grafiki:

- Wykonaj obraz i jego miniaturę na podstawie grafiki *rys.png* wypakowanej z archiwum



Plik *rys.png*

- Obraz przytnij / kadruj do ramki widocznej na *rys.png*. W skadrowanym obrazie ramka ta nie powinna występować (powinna być odcięta)

- Skaluj obraz z zachowaniem proporcji do szerokości 400 px i zapisz tak przygotowany obraz jako pies w formacie JPEG
- Skaluj obraz z zachowaniem proporcji do szerokości 120 px i zapisz tak przygotowany obraz jako pies-mini w formacie JPEG

Cechy witryny:

- Nazwa strony: *salon.php*
- Zastosowany właściwy standard kodowania polskich znaków
- Tytuł strony, widoczny na karcie przeglądarki: „Salon pielęgnacji”
- Arkusz stylów w pliku o nazwie *salon.css*, prawidłowo połączony z kodem strony
- Podział strony na bloki: baner, obok siebie panele lewy, środkowy, prawy zrealizowany za pomocą znaczników sekcji
- Zawartość banera: nagłówek pierwszego stopnia o treści: „SALON PIELĘGNACJI PSÓW I KOTÓW”
- Zawartość panelu lewego:
 - Nagłówek trzeciego stopnia o treści: „SALON ZAPRASZA W DNIACH”,
 - Lista punktowana z dwoma punktami: „Poniedziałek, 12:00 - 18:00” i „Wtorek, 12:00 - 18:00”
 - Obraz *pies-mini*, który jest jednocześnie odnośnikiem prowadzącym do grafiki *pies*, obraz powinien opływać tekst następnego akapitu z lewej strony (nie należy wykorzystywać tabeli, opływanie należy zdefiniować w kodzie CSS)
 - Tekst akapitu (paragrafu) o treści „Umów się telefonicznie na wizytę lub po prostu przyjdź!”
- Zawartość panelu środkowego
 - Nagłówek trzeciego stopnia o treści: „PRZYPOMNIENIE O NASTĘPNEJ WIZYCIE”
 - Wyniki działania skryptu nr 1
- Zawartość panelu prawego:
 - Nagłówek trzeciego stopnia o treści: „USŁUGI”
 - Wyniki działania skryptu nr 2

Styl CSS witryny internetowej

Plik *salon.css* zawiera formatowanie dla:

- Banera: kolor RGB tła (30, 137, 157), biały kolor czcionki, rozmiar czcionki 120%, wyrównanie tekstu do prawej strony, wysokość 100 px
- Panelu lewego i prawego: kolor RGB tła (58, 159, 178), szerokość 20%, wysokość 550 px
- Panelu środkowego: kolor RGB tła (143, 212, 225), szerokość 60%, wysokość 550 px
- Nagłówka trzeciego stopnia: biały kolor czcionki
- Obrazu: zdefiniowane opływanie do lewej strony

Niewymienione właściwości obiektów przybierają wartości domyślne.

Skrypt połączenia z bazą

W tabeli 1 podano wybór funkcji PHP do obsługi bazy danych. Wymagania dotyczące skryptu:

- Napisany w języku PHP

- Nie jest wymagane sprawdzenie czy operacja na bazie powiodła się
- Skrypt łączy się z serwerem bazodanowym na *localhost*, użytkownik **root** bez hasła, baza danych o nazwie salon
- Na końcu działania skryptu powinno zostać obsłużone zamknięcie połączenia z serwerem
- Działanie skryptu nr 1:
 - Skrypt wysyła do bazy zapytanie 2 (z pliku *kwerendy.txt*)
 - Z każdego zwróconego rekordu wyświetlane są informacje według wzoru (tekst napisany w nawiasach < > oznacza zwrócone zapytaniem pole):
 - Jeżeli pole rodzaj ma wartość 1 w pierwszej linii wypisane jest: „Pies: <imie>”
 - Jeżeli pole rodzaj ma wartość 2 w pierwszej linii wypisane jest: „Kot: <imie>”
 - Druga linia: „Data następnej wizyty: <nastepna_wizyta>, telefon właściciela: <telefon>”
- Działanie skryptu nr 2:
 - Skrypt wysyła do bazy zapytanie 1 (z pliku *kwerendy.txt*)
 - Każdy zwrócony rekord jest wyświetlany w osobnej linii

Tabela 1. Wybór funkcji języka PHP do obsługi bazy MySQL i MariaDB

Funkcje biblioteki mysql	Funkcje biblioteki mysqli	Zwracana wartość
mysql_connect(serwer, użytkownik, hasło)	mysqli_connect(serwer, użytkownik, hasło, nazwa_bazy)	id połączenia lub FALSE, gdy niepowodzenie
mysql_select_db('nazwa_bazy' [,id_połączenia])	mysqli_select_db(id_połączenia, nazwa_bazy)	TRUE/FALSE w zależności od stanu operacji
mysql_error([id_połączenia])	mysqli_error(id_połączenia)	Tekst komunikatu błędu
mysql_close([id_połączenia])	mysqli_close(id_połączenia)	TRUE/FALSE w zależności od stanu operacji
mysql_query(zapytanie [,id_połączenia])	mysqli_query(id_połączenia, zapytanie)	Wynik zapytania
mysql_fetch_row(wynik_zapytania)	mysqli_fetch_row(wynik_zapytania)	Tablica numeryczna odpowiadająca wierszowi zapytania
mysql_fetch_array(wynik_zapytania)	mysqli_fetch_array(wynik_zapytania)	Tablica zawierająca kolejny wiersz z podanych w wyniku zapytania lub FALSE, jeżeli nie ma więcej wierszy w wyniku zapytania
mysql_num_rows(wynik_zapytania)	mysqli_num_rows(wynik_zapytania)	Liczba wierszy w podanym zapytaniu
mysql_num_fields(wynik_zapytania)	mysqli_num_fields(wynik_zapytania)	Liczba kolumn w podanym zapytaniu

UWAGA: po zakończeniu pracy nagraj płytę z rezultatami pracy. W folderze z Twoim numerem PESEL powinny się znajdować pliki: import_salon.png, kw1.png, kw2.png, kw3.png, kw4.png, kwerendy.txt, pies.jpg, pies-mini.jpg, salon.php, salon.css, ewentualnie inne przygotowane przez Ciebie pliki. Po nagraniu płyty sprawdź poprawność nagrania i opisz płytę swoim numerem PESEL.

Czas przeznaczony na wykonanie zadania wynosi 150 minut.

Ocenie będą podlegać 4 rezultaty:

- operacje na bazie danych,
- witryna internetowa,
- styl CSS witryny internetowej,
- skrypt połączenia z bazą.