

Zadanie egzaminacyjne

UWAGA: Katalog z rezultatami pracy oraz płytę należy opisać numerem zdającego, którym został podpisany arkusz, czyli numerem PESEL, lub w przypadku jego braku numerem paszportu. Dalej w zadaniu numer ten jest nazwany numerem zdającego.

Wykonaj aplikację internetową portalu biura podróży, wykorzystując pakiet XAMPP oraz edytor zaznaczający składnię.

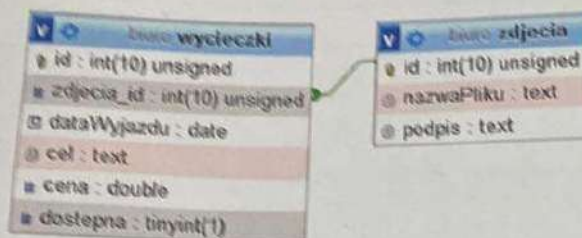
Aby wykonać zadanie, należy zalogować się na konto **Egzamin** bez hasła. Na pulpicie znajduje się archiwum ZIP o nazwie **zad4.zip** zabezpieczone hasłem: **&poDrozeE**

Archiwum należy rozpakować.

Na pulpicie konta **Egzamin** należy utworzyć folder. Jako nazwy folderu należy użyć numeru zdającego. Rozpakowane pliki należy umieścić w tym folderze. Po skończonej pracy wszystkie wyniki należy zapisać w tym folderze.

Operacje na bazie danych

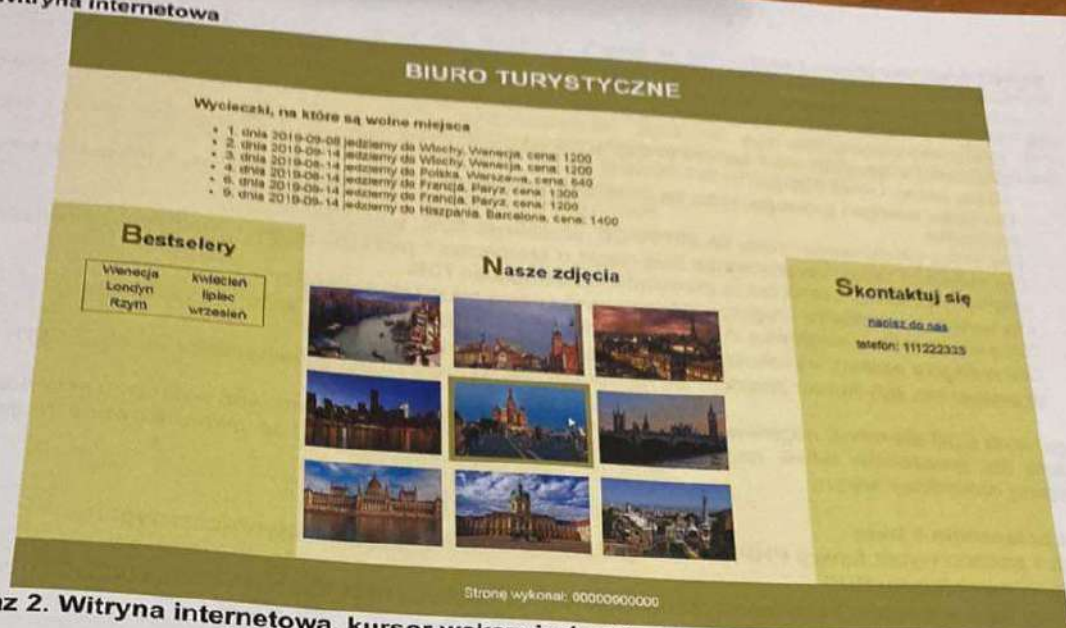
Do wykonania zadania należy użyć tabel wycieczki i zdjęcia przedstawionych na obrazie 1. Wycieczka jest dostępna, jeśli pole dostępna przyjmuje wartość TRUE



Obraz 1. Tabele wycieczki i zdjęcia

Za pomocą narzędzia phpMyAdmin wykonaj operacje na bazie danych:

- Utwórz bazę danych o nazwie *biuro*
- Do bazy *biuro* zaimportuj tabele z pliku *egzamin3.sql* z rozpakowanego archiwum
- Wykonaj zrzut ekranu po imporcie. Zrzut zapisz w formacie PNG pod nazwą *import*. Nie kadruj zrzutu. Zrzut powinien obejmować cały ekran monitora, z widocznym paskiem zadań. Na zrzucie powinny być widoczne elementy wskazujące na poprawnie wykonany import tabel
- Wykonaj zapytania SQL działające na bazie *biuro*. Zapytania zapisz w pliku *kwerendy.txt*. Wykonaj zrzuty ekranu przedstawiające wyniki działania kwerend. Zrzuty zapisz w formacie JPEG i nadaj im nazwy *kw1*, *kw2*, *kw3*, *kw4*. Zrzuty powinny obejmować cały ekran monitora z widocznym paskiem zadań
- Zapytanie 1: wybierające jedynie pola *id*, *dataWyjazdu*, *cel* i *cena* z tabeli *wycieczki* dla dostępnych wycieczek
- Zapytanie 2: wybierające jedynie pola *nazwaPliku* i *podpis* z tabeli *zdjecia*, sortując je malejąco według kolumny *podpis*
- Zapytanie 3: wybierające jedynie *cel* i *cena* wycieczki z tabeli *wycieczki* oraz odpowiadający im *podpis* z tabeli *zdjecia* dla wycieczek, których *cena* jest wyższa niż 1200 zł. Zapytanie wykorzystuje relację
- Zapytanie 4: dodające do tabeli *wycieczki* pole *liczbaDni* przyjmujące tylko liczby naturalne, pole powinno być wstawione po polu *dataWyjazdu*



Obraz 2. Witryna internetowa, kursor wskazuje środkowy obrazek, co powoduje ustawienie tła

Cechy witryny:

- Składa się ze strony o nazwie wycieczki.php
- Zapisana w języku HTML 5
- Jawnie zastosowany właściwy standard kodowania polskich znaków
- Zadeklarowany język witryny: polski
- Tytuł strony widoczny na karcie przeglądarki: „Wycieczki po Europie”
- Arkusz stylów w pliku o nazwie styl4.css prawidłowo połączony z kodem strony
- Podział strony na bloki: na górze baner, poniżej blok z danymi, poniżej obok siebie trzy bloki: lewy, środkowy i prawy, poniżej stopka. Podział zrealizowany za pomocą znaczników sekcji tak, aby uruchomieniu w przeglądarce wygląd układu bloków był zgodny z Obrazem 2
- Zawartość banera: nagłówek pierwszego stopnia o treści: „BIURO TURYSTYCZNE”
- Zawartość bloku z danymi:

- Nagłówek trzeciego stopnia o treści „Wycieczki, na które są wolne miejsca”
- Lista punktowana (nieuporządkowana) wypełniona skryptem 1

Zawartość lewego bloku:

- Nagłówek drugiego stopnia o treści: „Bestsellery”
- Tabela o dwóch kolumnach i trzech wierszach wypełniona danymi:

Wenecja	kwiecień
Londyn	lipiec
Rzym	wrzesień

Zawartość środkowego bloku:

- Nagłówek drugiego stopnia o treści: „Nasze zdjęcia”
- Efekt działania skryptu 2

Zawartość prawego bloku:

- Nagłówek drugiego stopnia o treści: „Skontaktuj się”
- Odnośnik do adresu e-mail turysta@wycieczki.pl o treści: „napisz do nas”
- Paragraf (akapit) o treści: „telefon: 111222333”
- Zawartość stopki: akapit (paragraf) o treści: „Stronę wykonał: ”, dalej wstawiony numer z

Styl CSS witryny internetowej

Styl CSS zdefiniowany w całości w zewnętrznym pliku o nazwie `styl4.css`. Arkusz CSS zawiera formatowanie:

- Ustawione domyślne wartości dla wszystkich selektorów stylu CSS: krój czcionki Helvetica
- Wspólne dla banera i stopki: kolor tła #B3BC6D, biały kolor czcionki, wyrównanie tekstu do środka, marginesy wewnętrzne 5 px
- Dla bloku z danymi: kolor tła #FFFFCE, kolor czcionki Olive, marginesy wewnętrzne: górny i dolny 10 px, prawy i lewy 150 px
- Dla bloku lewego i prawego: kolor tła #E6EE9C, szerokość 25%, wysokość 450 px, wyrównanie tekstu do środka
- Dla bloku środkowego: kolor tła #FFFFCE, szerokość 50%, wysokość 450 px
- Dla selektora tabeli: obramowanie linią ciągłą o szerokości 1 px i kolorze Olive, marginesy zewnętrzne automatycznie wyliczane przez przeglądarkę, szerokość 70%
- Dla selektora nagłówka drugiego stopnia: wyrównanie tekstu do środka
- Dla pierwszej litery nagłówka drugiego stopnia: rozmiar czcionki 150%, kolor czcionki Olive
- Dla selektora obrazu: wysokość 100 px, marginesy wewnętrzne 7 px
- W momencie, gdy kursor znajduje się na obrazie, styl obrazu zmienia się na: kolor tła #B3BC6D

Uwaga: style CSS dla tabeli, nagłówka drugiego stopnia i obrazu należy zdefiniować wyłącznie przy pomocy selektora dla znaczników tabeli, nagłówka drugiego stopnia i obrazu. Jest to uwarunkowane projektem późniejszej rozbudowy witryny.

Skrypt połączenia z bazą

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

- Napisany w języku PHP
- Łączy się z serwerem bazodanowym na `localhost`, użytkownik `root` bez hasła, baza danych o nazwie `biuro`
- Skrypt 1
 - Wysyła do bazy danych zapytanie 1
 - Każdy zwrócony zapytaniem wiersz jest wyświetlany w elemencie listy, według wzoru:
„<id>. dnia <dataWyjazdu> jedziemy do <cel>, cena: <cena>”, gdzie w znakach < > zapisano pola zwrócone zapytaniem
- Skrypt 2
 - Wysyła do bazy danych zapytanie 2
 - Dane z każdego zwróconego zapytaniem wiersza są wykorzystane do wyświetlenia kolejnych obrazów, w ten sposób, że pole `nazwaPliku` jest źródłem obrazu, a pole `podpis` jest tekstem alternatywnym obrazu
- Na końcu działania skrypt zamyka połączenie z serwerem.

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

Funkcje biblioteki MySQLi	Zwracana wartość
<code>mysqli_connect(serwer, użytkownik, hasło, nazwa_bazy)</code>	id połączenia lub FALSE, gdy niepowodzenie
<code>mysqli_select_db(id_połączenia, nazwa_bazy)</code>	TRUE/FALSE w zależności od stanu operacji
<code>mysqli_error(id_połączenia)</code>	Tekst komunikatu błędu
<code>mysqli_close(id_połączenia)</code>	TRUE/FALSE w zależności od stanu operacji
<code>mysqli_query(id_połączenia, zapytanie)</code>	Wynik zapytania
<code>mysqli_fetch_row(wynik_zapytania)</code>	Tablica numeryczna odpowiadająca wierszowi zapytania
<code>mysqli_fetch_array(wynik_zapytania)</code>	Tablica zawierająca kolejny wiersz z podanych w wyniku zapytania lub FALSE, jeżeli nie ma więcej wierszy w wyniku zapytania
<code>mysqli_num_rows(wynik_zapytania)</code>	Liczba wierszy w podanym zapytaniu
<code>mysqli_num_fields(wynik_zapytania)</code>	Liczba kolumn w podanym zapytaniu