

Zgłoszenie na kartę wędkarską

Formularz - karta wędkarska

imię:

nazwisko:

adres:

Typy łowisk

- Zalewy
- Stawy
- Jeziora
- Rzeki

Stronę przygotował: 0000000000

Karta wędkarska

Obraz 2. Witryna internetowa

Przygotowanie grafiki:

- Plik *wedka.jpg*, wypakowany z archiwum, należy przeskalować z zachowaniem proporcji tak, aby jego wysokość wynosiła dokładnie 220 px

Cechy witryny:

- Składa się ze strony o nazwie *karta.html* oraz skryptu *zapisz.php*. Poniższe wymagania dotyczą tylko pliku *karta.html*
- Zastosowany właściwy standard kodowania polskich znaków
- Tytuł strony widoczny na karcie przeglądarki: „Karty wędkarskie”
- Arkusz stylów w pliku o nazwie *styl_1.css* prawidłowo połączony z kodem strony
- Podział strony na bloki: na górze blok banera, poniżej blok główny, na dole trzy bloki: lewy, środkowy i prawy. Podział zrealizowany za pomocą znaczników sekcji tak, aby po uruchomieniu w przeglądarce wygląd układu bloków był zgodny z Obrazem 2
- Zawartość bloku banera: nagłówek drugiego stopnia o treści „Zgłoszenie na kartę wędkarską”
- Zawartość bloku głównego:
 - Nagłówek czwartego stopnia o treści: „Formularz - karta wędkarska”
 - Formularz wysyłający dane metodą POST do pliku *zapisz.php* zawierający pola
 - Pole edycyjne podpisane powyżej tekstem: „imię: ”
 - Pole edycyjne podpisane powyżej tekstem: „nazwisko: ”
 - Pole edycyjne podpisane powyżej tekstem: „adres: ”; pole ma rozmiar 30
 - Przycisk czyszczący zawartość formularza z etykietą „CZYŚĆ”
 - Przycisk wysyłający zawartość formularza z etykietą „ZAPISZ”
- Zawartość bloku lewego:
 - Nagłówek czwartego stopnia o treści: „Typy łowisk”
 - Lista punktowana (nieuporządkowana) zawierająca elementy: Zalewy, Stawy, Jeziora, Rzeki

- Zawartość bloku środkowego: obraz *wedka.jpg* z tekstem alternatywnym o treści: „karta wędkarska”
- Zawartość bloku prawego:
 - Akapit (paragraf) o treści: „Stronę przygotował: ”, dalej wstawiony numer PESEL zdającego lub w przypadku jego braku inny numer, którym został podpisany arkusz
 - Odnośnik do strony „http://www.kartawedkarska.pl/” o treści: „Karta wędkarska”

Styl CSS witryny internetowej

Cechy formatowania CSS:

- Są zawarte wyłącznie w zewnętrznym arkuszu stylów
- Domyślne dla całej strony: krój czcionki Georgia
- Dla bloku banera: kolor tła DodgerBlue, biały kolor czcionki, marginesy wewnętrzne 10 px, wyrównanie tekstu do środka
- Dla bloku głównego: kolor tła LightBlue, marginesy wewnętrzne 100 px
- Dla bloku lewego: kolor tła DodgerBlue, biały kolor czcionki, szerokość 20%, wysokość 220 px
- Wspólne dla bloku środkowego i bloku prawego: kolor tła DodgerBlue, biały kolor czcionki, szerokość 40%, wysokość 220 px, wyrównanie tekstu do środka
- Dla znacznika odnośnika: obramowanie linią kreskową o szerokości 1 px i kolorze DimGray, kolor tła Gold, kolor czcionki DarkRed, marginesy wewnętrzne 10 px

Skrypt połączenia z bazą

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

- Napisany w języku PHP, w pliku *zapisz.php*
- Skrypt działa na danych przesłanych z formularza
- Skrypt łączy się z serwerem bazodanowym na *localhost*, użytkownik **root** bez hasła, baza danych o nazwie *wedkowanie*
- Skrypt wysyła do bazy danych zmodyfikowane zapytanie 1, które w polach *data_zezwolenia* i *punkty* zawiera wartość NULL, pozostałe wartości wypełnione są danymi z formularza
- 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_polaczenia, nazwa_bazy)</code>	TRUE/FALSE w zależności od stanu operacji
<code>mysqli_error(id_polaczenia)</code>	Tekst komunikatu błędu
<code>mysqli_close(id_polaczenia)</code>	TRUE/FALSE w zależności od stanu operacji
<code>mysqli_query(id_polaczenia, 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