

Nazwa
kwalifikacji:

Tworzenie i administrowanie stronami i aplikacjami internetowymi oraz bazami danych

Oznaczenie
kwalifikacji:

INF.03

Numer zadania:

07

Kod arkusza:

INF.03-07-24.06-SG

Wersja arkusza:

SG

Lp.	Elementy podlegające ocenie/kryteria oceny
R.1	Rezultat 1: Operacje na bazie danych
	<i>Uwaga: W przypadku oceny zrzutów należy uznać za prawidłowe jeżeli widoczny jest cały obszar ekranu z widocznym paskiem zadań, a zapytanie ma charakter uniwersalny dla każdego zestawu danych. Nie należy oceniać wykadowanych zrzutów ekranu. Jeżeli wszystkie rekordy nie są widoczne to sprawdzić w phpMyAdmin</i> WAŻNE! Przed sprawdzeniem każdej pracy egzaminacyjnej należy <u>usunąć</u> wstawione wiersze czyli wydać polecenie <code>DELETE FROM `wagi` WHERE id > 22;</code>
R.1.1	Wykonano import tabel do bazy danych <i>wazienietirow</i> czynność udokumentowano plikiem o nazwie <i>import</i> w formacie PNG
R.1.2	Zapisano plik <i>kwerendy.txt</i> zawierający co najmniej jedno zapytanie SQL, wynikające z treści zadania
R.1.3	Wykonano zapytanie 1 aktualizujące tabelę <i>wagi</i> . Rekordy, w których pole <i>lokalizacje_id</i> jest równe 2, 3, 4 mają zmienioną datę i czas na wartości aktualne. W pliku z kwerendami lub na zrzucie istnieje zapis np.: <code>UPDATE wagi SET dzien=CURRENT_DATE, czas=CURRENT_TIME WHERE lokalizacje_id BETWEEN 2 AND 4;</code> (dowolne funkcje wstawiające aktualną datę/czas, lub także z pustymi nawiasami; dowolny warunek sprawdzający przedział 2..4) oraz wynik działania udokumentowano zrzutem, na którym jest widoczny komunikat o poprawnie wykonanym zapytaniu lub wiersze z <i>lokalizacje_id</i> równym 2, 3, 4 mają datę i czas zgodny z terminem egzaminu
R.1.4	Wykonano zapytanie 2 wybierające jedynie ulicę z tabeli <i>lokalizacje</i> . W pliku z kwerendami lub na zrzucie istnieje zapis np.: <code>SELECT ulica FROM lokalizacje;</code> oraz wynik działania udokumentowano zrzutem, na którym są widoczne jedynie dane: Ślężna, Tyska, Średzka, Wilanowska, Żmigrodzka, Karkonowska
R.1.5	Wykonano zapytanie 3 wstawiające rekord do tabeli <i>wagi</i> z danymi: <i>lokalizacje_id</i> : 5, <i>waga</i> : liczba pseudolosowa z przedziału 1..10, rejestracja: DW12345, aktualna data i czas. Klucz główny nadawany automatycznie. W pliku z kwerendami lub na zrzucie istnieje zapis np.: <code>INSERT INTO `wagi`(`lokalizacje_id`, `waga`, `rejestracja`, `dzien`, `czas`) VALUES ('5', FLOOR(1+RAND()*10), 'DW12345', CURRENT_DATE, CURRENT_TIME);</code> (dowolny sposób liczenia liczby pseudolosowej z przedziału 1..10; dowolne funkcje wstawiające aktualny czas/datę, też z pustymi nawiasami) oraz wynik działania udokumentowano zrzutem, na którym jest widoczny komunikat o poprawnie wykonanym zapytaniu lub wstawiony do tabeli wiersz

R.1.6	Wykonano zapytanie 4 wybierające jedynie pola rejestracja, waga, dzien, czas z tabeli <i>wagi</i> i odpowiadające mu pole ulica z tabeli <i>lokalizacje</i> dla pojazdów, których waga na oś przekracza 5 t. W pliku z kwerendami lub na zrzucie istnieje zapis np.: SELECT rejestracja, ulica, waga, dzien, czas FROM wagi JOIN lokalizacje ON wagi.lokalizacje_id = lokalizacje.id WHERE waga > 5; (możliwe również INNER JOIN lub bez JOIN i porównanie kluczy po WHERE) <u>oraz</u> wynik działania udokumentowano zrzutem, na którym widoczne są rekordy z wagą powyżej 5; jedynie pola: rejestracja, waga, dzien, czas, ulica. Liczba wierszy dowolna, w zależności od wykonania zapytań 1 i 3
R.2	Rezultat 2: Zawartość witryny internetowej
	<i>Uwaga: W napisach widocznych na stronie dopuszcza się drobne błędy literowe (nie zmieniające sensu tekstu), błędy wielkości liter i znaków diakrytycznych, tekst może być pisany w cudzysłowach lub bez</i> <i>Jeśli witryna nie uruchamia się z powodu błędów PHP – sprawdzić wg kryteriów w kodzie źródłowym strony</i>
R.2.1	Plik <i>obraz1.png</i> zmodyfikowano w ten sposób że przy samochodzie widnieje tonaż 5T. Napis 8T nie jest widoczny
R.2.2	Plik <i>obraz2.jpg</i> odbito wzdłuż osi pionowej oraz przeskalowano z zachowaniem proporcji do szerokości 350 px. Odpowiadająca temu wysokość wynosi 285 px (± 5 px)
R.2.3	Witrynę internetową zapisano w pliku <i>wazenie.php</i> oraz zapisano jawnie standard kodowania polskich znaków i deklarację HTML5 <!DOCTYPE HTML> i zadeklarowano język witryny polski np. <HTML lang="pl">. Strona zawiera sekcje head i body
R.2.4	Nadano tytuł stronie: <i>Ważenie samochodów ciężarowych</i>
R.2.5	Układ strony zdefiniowano dzieląc ją na bloki: dwa bloki banera, blok lewy, środkowy, prawy oraz stopkę. Zastosowano semantyczne znaczniki sekcji HTML5 zgodnie z ich znaczeniem, w tym do części nagłówkowej - header, stopki - footer, dopuszcza się zagnieżdżenie bloków div w znacznikach semantycznych
R.2.6	W pierwszym banerze zapisano nagłówek h1: <i>"Ważenie pojazdów we Wrocławiu"</i> , w bloku lewym h2: <i>"Lokalizacje wag"</i> i <i>"Kontakt"</i> , w bloku środkowym h2: <i>"Alerty"</i> oraz w stopce paragraf z numerem zdającego
R.2.7	W bloku lewym umieszczono jedną listę numerowaną z elementami, zastosowano znaczniki , (listę wygenerowano skryptem lub w HTML)
R.2.8	W bloku środkowym umieszczono jedną tabelę o pięciu kolumnach, zastosowano znaczniki <table>, <tr>, <td> (tabelę wygenerowano skryptem lub w HTML)
R.2.9	Pierwszy wiersz tabeli zawiera komórki nagłówkowe o treści, kolejno: <i>„rejestracja”</i> , <i>„ulica”</i> , <i>„waga”</i> , <i>„dzień”</i> , <i>„czas”</i> , zastosowano znaczniki <th>
R.2.10	W drugim banerze umieszczono obraz z limitem wagi z tekstem alternatywnym <i>"waga"</i> oraz w bloku prawym umieszczono obraz z ciężarówką z tekstem alternatywnym <i>"tir"</i>
R.3	Rezultat 3: Działanie witryny internetowej
	<i>Uwaga: Jeśli witryna nie uruchamia się z powodu błędów PHP – uruchomić plik lokalnie z dysku lub sprawdzić wg kryteriów w kodzie źródłowym strony</i>
R.3.1	Układ bloków po uruchomieniu strony w przeglądarce jest zgodny z obrazem 2 w arkuszu (prawidłowo zastosowano właściwości CSS decydujące o układzie strony, np. float i clear albo display flex albo grid)
R.3.2	Odknośnik zapisano jako pocztowy (zastosowano <i>"mailto:"</i>), prowadzi on do adresu <i>wazenie@wroclaw.pl</i>
R.3.3	Strona zawiera działające połączenie z zewnętrznym arkuszem stylów o nazwie <i>styl.css</i> , formatowanie CSS pochodzi jedynie z tego arkusza <i>Uwaga: kryterium to nie jest spełnione również, gdy zdający źle przypisał co najmniej jeden selektor do bloku (np. kropka zamiast #)</i>
R.4	Rezultat 4: Styl CSS witryny internetowej

<i>Uwaga: W przypadku, gdy nie jest spełnione kryterium 3.3, kryteria w rezultacie R.4 należy ocenić w CSS. Składnia musi być zgodna ze specyfikacją CSS</i> <i>Kryteria 4.6 i 4.8 są spełnione jedynie, gdy zdefiniowano styl dla selektora td lub table, nie są spełnione gdy zdefiniowano styl klasy lub inny</i>	
R.4.1	Ustawiono domyślne formatowanie wszystkich selektorów (dla *): krój czcionki Tahoma
R.4.2	Ustawiono kolor tła #519657 dla obu banerów, #E1E1E1 dla bloku lewego, środkowego i prawego oraz #80C683 dla stopki oraz biały kolor czcionki dla obu banerów
R.4.3	Ustawiono wysokość obu banerów na 150px, bloków lewego, środkowego i prawego na 450px
R.4.4	Ustawiono szerokość pierwszego banera 70%, drugiego banera 30%, bloku lewego i prawego na 25%, bloku środkowego 50% (<i>width</i> lub <i>grid-template-columns</i> lub <i>flex</i>) oraz 100% szerokości jedynie dla obrazu o nazwie <i>obraz2.jpg</i>
R.4.5	Ustawiono wyrównanie tekstu do środka dla obu banerów i stopki
R.4.6	Ustawiono marginesy wewnętrzne (<i>padding</i>) dla selektora td 2px oraz jedynie dla obrazu o nazwie <i>obraz2.jpg</i> : jedynie górny i dolny 100px
R.4.7	Ustawiono dla bloku środkowego automatyczne paski przewijania (<i>overflow: auto</i> lub <i>scroll</i>)
R.4.8	Ustawiono dla selektora table: obramowanie 1px solid #80C683
R.4.9	Gdy kursor myszy znajdzie się na wierszu tabeli, kolor tła wiersza zmienia się na #80C683 (dla <i>tr:hover</i>)
R.4.10	Ostatni wiersz tabeli ma tło czerwone np. dla <i>tr:nth-last-child(1)</i>
R.5	Rezultat 5: Skrypt połączenia z bazą
<i>Uwaga: rezultat R.5 sprawdzić w wersji XAMPP 8.1.2. Jeśli witryna nie uruchamia się z powodu błędów PHP - sprawdzić 5.1 - 5.3, 5.8 wg kryteriów w kodzie źródłowym strony. Kryteria uznajemy za spełnione jeżeli zastosowano biblioteki MySQLi lub PDO</i>	
R.5.1	Skrypt zawiera działające połączenie z serwerem bazy danych i wybór bazy <i>wazenietirow</i>
R.5.2	Ostatnią operacją na bazie jest jej zamknięcie (zgodne ze składnią)
R.5.3	Skrypt 1 wysyła do bazy danych zapytanie 2 lub skrypt 2 wysyła zapytanie 4 lub skrypt 3 wysyła zapytanie 3
R.5.4	Skrypt 1 wyświetla zwrócone zapytaniem nazwy ulic według wzoru: " <i>ulica <nazwa_ulicy></i> ", gdzie nawiasy <> oznaczają wartość pobraną z bazy danych
R.5.5	Skrypt 2 wyświetla zwrócone zapytaniem dane z każdego wybranego z bazy wiersza
R.5.6	Skrypt 3 odświeża aktualną stronę co 10 sekund, np. <i>header("refresh: 10");</i>
R.5.7	Skrypt 3 wstawia rekord do bazy (sprawdzić w phpMyAdmin)
R.5.8	Zastosowano znaczące nazewnictwo dla wszystkich zmiennych lub funkcji w języku polskim lub angielskim. Nazwy odzwierciedlają cel zastosowania