

Nazwa
kwalifikacji:

Tworzenie i administrowanie stronami i aplikacjami internetowymi oraz bazami danych

Oznaczenie
kwalifikacji:

INF.03

Numer zadania:

02

Kod arkusza:

INF.03-02-26.01-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 wykadrowanych zrzutów ekranu. Jeżeli część rekordów nie jest widoczna, sprawdzić w PHPMyAdmin</i>
R.1.1	Wykonany import tabel do bazy danych <i>bazar</i> - plik <i>import.png</i>
R.1.2	W pliku <i>kwerendy.txt</i> zapisane co najmniej jedno zapytanie SQL, wynikające z treści zadania
R.1.3	Wykonane zapytanie 1 wybierające 10 pierwszych rekordów z tabeli <i>towar</i> , jedynie pola nazwa i plik. W pliku z kwerendami lub na zrzucie istnieje zapis np. <code>SELECT nazwa, plik FROM towar LIMIT 10;</code> oraz na zrzucie widocznych jest dokładnie 10 wierszy z nazwami: gruszka, jabłko, banan, ananas, mango, winogrona, jagody, maliny, pomarańcza, cytryna
R.1.4	Wykonane zapytanie 2 wybierające jedynie id i nazwę towarów. W pliku z kwerendami lub na zrzucie istnieje zapis np. <code>SELECT id, nazwa FROM towar;</code> oraz na zrzucie widoczne są wyniki działania zapytania (<i>nie wszystkie rekordy będą widoczne, 39 lub 40 rekordów</i>)
R.1.5	Wykonane zapytanie 3 wybierające jedynie rodzaj, nazwę i cenę towaru dla id = 1. W pliku z kwerendami lub na zrzucie istnieje zapis np. <code>SELECT rodzaj, nazwa, cena FROM towar WHERE id = 1;</code> oraz na zrzucie widoczny jest dokładnie jeden rekord: owoc gruszka 5.00
R.1.6	Wykonane zapytanie 4 wstawiające do tabeli <i>zamowienie</i> rekord o wartościach: towar o id 9, sklep o id 2, liczba kilogramów 20, klucz główny nadawany automatycznie. W pliku z kwerendami lub na zrzucie istnieje zapis np. <code>INSERT INTO zamowienie(id_towar, id_sklep, liczba_kg) VALUES (9, 2, 20);</code> (<i>klucz główny pominięty lub NULL</i>) oraz na zrzucie jest widoczny komunikat o poprawnie wykonanym zapytaniu lub tabela z dodanym rekordem
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 – uruchomić plik lokalnie z dysku lub sprawdzić wg kryteriów w kodzie źródłowym strony</i>
R.2.1	Grafika <i>jablko.png</i> lub <i>jabłko.png</i> ma przezroczyste tło
R.2.2	Grafika <i>market.png</i> przeskalowana z zachowaniem proporcji do szerokości dokładnie 300 px, odpowiadająca temu wysokość to 378 px (± 5 px)
R.2.3	W pliku <i>index.php</i> znajduje się deklaracja HTML5 <code><!DOCTYPE HTML></code> i deklaracja języka zawartości witryny: polski np. <code><html lang="pl"></code>
R.2.4	W sekcji head znajduje się jawnie podany standard kodowania polskich znaków i tytuł strony "Zdrowy bazarek"
R.2.5	W sekcji body zdefiniowany jest podział na bloki za pomocą semantycznych znaczników HTML5: header, nav, main->aside+section, footer (<i>dopuszcza się zastosowanie bloków pomocniczych</i>)

R.2.6	W bloku nagłówkowym umieszczony <h1>: "Zdrowy bazarek", w bloku sekcji <p>, w stopce <p> z numerem zdającego
R.2.7	W bloku bocznym umieszczony obraz z pliku <i>market.png</i> , z tekstem alternatywnym "bazarek"
R.2.8	W bloku sekcji umieszczony formularz
R.2.9	Na stronie umieszczona lista rozwijana (w <i>HTML</i> lub <i>skrypcie</i>), zastosowane znaczniki <select> i <option>, pole edycyjne i przycisk "Zamów"
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 ilustracją 3 w arkuszu egzaminacyjnym (prawidłowo zastosowano właściwości CSS decydujące o układzie bloków na stronie, np. <i>float</i> i <i>clear</i> albo <i>display flex</i> albo <i>grid</i> albo <i>position</i>)
R.3.2	Strona zawiera działające połączenie z zewnętrznym arkuszem stylów o nazwie <i>styl.css</i> formatowanie CSS pochodzi jedynie z tego arkusza
R.3.3	Formularz wysyła dane metodą POST do tego samego pliku (poprawne również bez atrybutu <i>action</i>)
R.3.4	Przycisk realizuje funkcję wysłania danych z formularza (<i>input</i> lub <i>button</i> z typem <i>submit</i> lub <i>button</i> bez typu)
R.3.5	Pole edycyjne jest typu <i>number</i>
R.4	Rezultat 4: Styl CSS witryny internetowej
<i>Uwaga: Kryteria muszą działać na stronie (sprawdzić w inspektorze). Dopuszcza się sprawdzenie w kodzie CSS tylko w przypadku, gdy formatowany element nie istnieje w HTML lub plik CSS nie został powiązany z witryną, a kod CSS jest zgodny ze składnią Ustawione:</i>	
R.4.1	domyślne formatowanie wszystkich selektorów (*): krój czcionki Georgia, serif (dozwolona tylko taka kolejność)
R.4.2	kolor tła: blok nagłówkowy i stopka - ForestGreen, blok nawigacyjny - DarkKhaki, bloki boczny i sekcji - Khaki
R.4.3	kolor czcionki: blok nagłówkowy i stopka - biały
R.4.4	wysokość: blok nawigacyjny - 150px, bloki boczny i sekcji - 400px
R.4.5	szerokość: bloki boczny i sekcji - 50% (<i>width</i> lub <i>grid-template-columns</i> lub <i>flex</i>)
R.4.6	marginesy wewnętrzne (<i>padding</i>): blok nagłówkowy i stopka - 2px
R.4.7	wyrównanie tekstu do środka: blok nagłówkowy i stopka
R.4.8	przepełnienie bloku: blok nawigacyjny - hidden
R.4.9	styl jedynie dla obrazu w bloku bocznym: marginesy wewnętrzne - 2px, zewnętrzne - 10px, obramowanie 2px dotted ForestGreen, zaokrąglenie rogów 40px (dowolną metodą: <i>id</i> , <i>klasa</i> , <i>potomek</i>)
R.4.10	styl selektora p: rozmiar czcionki 120%, kursywa
R.5	Rezultat 5: Skrypt połączenia z bazą
<i>Uwaga: rezultat R.5 sprawdzić w wersji XAMPP 8.2.12. Jeśli witryna nie uruchamia się z powodu błędów PHP - sprawdzić kryteria 5.1 ÷ 5.3, 5.6, 5.7, 5.10 w kodzie źródłowym strony. Instrukcje muszą być zgodne ze składnią oraz zawierać dane wynikające z zadania. Kryteria są sprawdzane tylko w przypadku użycia bibliotek MySQLi lub PDO</i>	
R.5.1	Skrypt realizuje połączenie z serwerem bazy danych i wybór bazy <i>bazar</i> oraz ostatnią operacją na bazie jest jej zamknięcie
R.5.2	Skrypt zawiera instrukcję wysyłającą do bazy danych zapytanie 1 lub zapytanie 2
R.5.3	Skrypt zawiera instrukcję wysyłającą do bazy danych zmodyfikowane zapytanie 3: sprawdzane jest id przesłane formularzem lub zmodyfikowane zapytanie 4: wstawiany jest rekord z id towaru i liczbą kg przesłaną zapytaniem
R.5.4	Skrypt 1 wyświetla obrazy dla każdego wiersza pobranego z bazy, źródłem jest pole plik, tekstem alternatywnym jest nazwa (obrazy mogą być ukryte za sprawą właściwości CSS <i>overflow</i> , zamiast obrazu z jabłkiem może się wyświetlać tekst alternatywny)

R.5.5	Skrypt 2 wypełnia listę rozwijaną z danymi pobranymi z bazy, atrybut value to id, wyświetlana wartość opcji to nazwa towaru
R.5.6	Skrypt 3 stosuje tablicę \$_POST dla danych wysłanych z formularza
R.5.7	Skrypt 3 oblicza wartość iloczynu ceny zwróconej zapytaniem oraz liczby kilogramów wysłanej z formularza
R.5.8	Skrypt 3 wyświetla w paragrafie tekst: „<rodzaj> <nazwa> wartość: <wartość> zł”, gdzie w znakach <> umieszczono zwrócone zapytaniem pola i wyliczoną wartość
R.5.9	Skrypt 3 wstawia do tabeli zamówienie rekord z danymi pobranymi z formularza (<i>należy sprawdzić w phpMyAdmin</i>)
R.5.10	W skryptach zastosowana przynajmniej jedna zmienna oraz nazwy wszystkich zmiennych są w języku polskim lub angielskim i odzwierciedlają cel zastosowania