Nazwa kwalifikacji:	**Programowanie, tworzenie i administrowanie stronami internetowymi i bazami danych**
Oznaczenie kwalifikacji:	**EE.09**
Numer zadania:	**02**
Kod arkusza:	**EE.09-02-23.01-SG**
wersja	**SG**

Lp.	Elementy podlegające ocenie/kryteria oceny
R.1	**Rezultat 1: Operacje na bazie danych**
	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
R.1.1	Wykonano import tabel do bazy danych *gory*, czynność udokumentowano plikiem o nazwie *import* w formacie JPEG
R.1.2	Zapisano plik tekstowy o nazwie *kwerendy* zawierający co najmniej jedno zapytanie SQL, wynikające z treści zadania
R.1.3	Utworzono zapytanie 1 wybierające jedynie pola nazwa i pasmo z tabeli *szczyty* dla szczytów wyższych niż 1500 m. W pliku z kwerendami lub na zrzucie istnieje zapis np.: `SELECT nazwa, pasmo FROM szczyty WHERE wysokosc > 1500;` oraz wynik działania zapytania udokumentowano zrzutem, na którym widoczne są dokładnie 3 rekordy: Babia Góra (Diablak), Beskid Żywiecki; Śnieżka, Karkonosze; Rysy, Tatry
R.1.4	Utworzono zapytanie 2 wybierające jedynie nazwę najwyższego szczytu w tabeli *szczyty*. W pliku z kwerendami lub na zrzucie istnieje zapis np.: `SELECT nazwa FROM szczyty ORDER BY wysokosc DESC LIMIT 1;` (Uwaga! Zapytanie można zapisać na kilka sposobów - ważny jest rezultat wykonania) oraz wynik działania zapytania udokumentowano zrzutem, na którym widoczna jest jedynie nazwa Rysy
R.1.5	Utworzono zapytanie 3 tworzące użytkownika **leon@localhost** z hasłem **Leon%Zawodowiec** W pliku z kwerendami lub na zrzucie istnieje zapis z jawnym hasłem, np.: `CREATE USER 'leon'@'localhost' IDENTIFIED BY 'Leon%Zawodowiec';` oraz wynik działania zapytania udokumentowano zrzutem, wskazującym na poprawne założenie konta użytkownika
R.1.6	Utworzono zapytanie 4 Nadające użytkownikowi **leon@localhost** jedynie wszystkie prawa do manipulowania danymi we wszystkich tabelach bazy *gory*. W pliku z kwerendami lub na zrzucie istnieje zapis np.: `GRANT UPDATE, DELETE, INSERT ON gory.* TO 'leon'@'localhost';` (możliwe też gory.szczyty - zamiast gwiazdki; opcjonalnie może być też SELECT) oraz wynik działania zapytania udokumentowano zrzutem wskazującym na poprawnie nadane prawa
R.1.7	W pliku *grants.png* widoczne są przydzielone prawa dla użytkownika **leon** dla tabel bazy *gory*
R.2	**Rezultat 2: Zawartość witryny internetowej**
	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
R.2.1	Plik *zdjecie .jpg* wykadrowano tak, aby był widoczny zamek oraz fragment gór. Po wykadrowaniu obraz ma orientację pionową
R.2.2	Witrynę internetową zapisano w pliku o nazwie *galeria.html*
R.2.3	Zastosowano deklarację języka HTML5, jawnie zapisany standard kodowania polskich znaków oraz zadeklarowano język witryny (np. <html lang = 'pl'>)
R.2.4	Nadano tytuł stronie: "Galeria"
R.2.5	Wygląd strony zdefiniowano dzieląc ją na bloki: baner, lewy i prawy, stopka. Zastosowano znaczniki sekcji

R.2.6	W bloku banera zapisano nagłówek h1: "Galeria obrazów", w bloku lewym - h3: "Zdjęcia opublikowali", w bloku stopki paragraf: „Galerię wykonał: "
R.2.7	W bloku lewym umieszczono tabelę o trzech wierszach i dwóch kolumnach, zastosowano znaczniki <table>, <tr>, <td>
R.2.8	W bloku lewym umieszczono obraz *zdjecie.jpg* z tekstem alternatywnym „zdjęcie dnia" oraz w bloku prawym obraz *1.jpg* z tekstem alternatywnym „galeria"
R.2.9	W bloku prawym umieszczono pięć zdjęć *1.jpg ÷ 5.jpg*, przynajmniej jedno z nich ma tekst alternatywny "miniatura"
R.2.10	W bloku prawym umieszczono dwa przyciski opisane "prev" i "next"
R.3	**Rezultat 3: Działanie witryny internetowej**
R.3.1	**Układ bloków** po uruchomieniu strony w przeglądarce jest zgodny z Obrazem 2 w arkuszu egzaminacyjnym (prawidłowo zastosowano właściwości CSS układające bloki na stronie, np. float i clear albo display flex albo grid)
R.3.2	Na stronie umieszczono odnośnik, którego kliknięcie inicjuje otwarcie strony *https://pixabay.com* w nowej karcie przeglądarki
R.3.3	Dla dowolnego przycisku zdefiniowano obsługę zdarzenia kliknięcia (atrybut onClick lub słuchacz zdarzenia click, lub właściwość onclick obiektu DOM)
R.3.4	Dla dowolnej miniatury zdefiniowano obsługę zdarzenia kliknięcia (atrybut onClick lub słuchacz zdarzenia click, lub właściwość onclick obiektu DOM)
R.3.5	Strona zawiera działające połączenie z zewnętrznym arkuszem stylów o nazwie *styl2.css*, formatowanie pochodzi jedynie z tego arkusza Uwaga: kryterium to nie jest spełnione również, gdy zdający źle przypisał co najmniej jeden selektor do bloku (np. kropka zamiast #)
R.4	**Rezultat 4: Styl CSS witryny internetowej**
	Uwaga: W przypadku, gdy nie jest spełnione kryterium 3.5, kryteria w rezultacie R.4 należy ocenić w kodzie CSS (w pliku CSS lub znaczniku <style>). Składnia musi być zgodna ze specyfikacją CSS *Kryterium R.4.7, R.4.10 jest spełnione jedynie, gdy zdefiniowano styl dla selektora, nie jest spełnione gdy zdefiniowano styl klasy lub inny*
R.4.1	Ustawiono domyślne wartości dla selektorów (*): krój czcionki Garamond, wyrównanie tekstu do środka
R.4.2	Ustawiono domyślne wartości dla selektorów (*): kolor tła MistyRose, kolor czcionki rgb(62, 39, 35)
R.4.3	Ustawiono wysokość 600px dla bloku lewego i prawego
R.4.4	Ustawiono szerokość 20% dla bloku lewego i 65% dla bloku prawego (width lub grid-template-columns lub flex)
R.4.5	Ustawiono marginesy zewnętrzne (margin) 30px dla bloku lewego, 30px dla dużego obrazu w bloku prawym i marginesy wewnętrzne (padding) 2px dla miniatur
R.4.6	Ustawiono dla bloku lewego lub dla dużego obrazu z bloku prawego cień o cechach: oba przesunięcia 15 px, rozmycie 10px, kolor DimGray, np. box-shadow: 15px 15px 10px DimGray (lub 15px 15px 10px 0 DimGray)
R.4.7	Ustawiono szerokość 100% dla selektora img umieszczonego **jedynie** w bloku lewym np. z użyciem selektora dziecka lub potomka, 500px dla dużego obrazu z bloku prawego oraz 140px dla miniatur
R.4.8	Ustawiono zaokrąglenie rogów (border-radius) 5% jedynie dla miniatur oraz paski przewijania w przypadku przepełnienia bloku lewego (overflow: auto;)
R.4.9	Ustawiono pseudoklasę hover dla miniatur. Gdy kursor znajdzie się na miniaturze pojawia się ramka o szerokości 1 px, czarną linią kropkową (dotted)
R.4.10	Ustawiono dla selektora tabeli obramowanie o szerokości 1 px, czarną linią kreskową (dashed) oraz szerokość 100%
R.5	**Rezultat 5: Skrypt**
	Uwaga: W przypadku, gdy skrypt się nie uruchamia, kryteria 5.1 ÷ 5.3 należy sprawdzić w kodzie witryny
R.5.1	Skrypt zapisano w języku JavaScript i zawarto w nim przynajmniej jedną poprawną składniowo instrukcję języka

R.5.2	Zapisano w skrypcie przynajmniej jedną poprawną składniowo funkcję zmieniającą w bloku prawym wartość atrybutu .src dużego zdjęcia (prawidłowa definicja funkcji i operacja getElementBy…)
R.5.3	Funkcje skryptu są kontrolowane przez zmienną przechowującą numer zdjęcia. Zmienna jest modyfikowana zgodnie z kontekstem aplikacji, wg wzoru: 1. przycisk next -> gdy zdjęcie 5 -> 1 lub inkrementacja 2. przycisk prev -> gdy zdjęcie 1 -> 5 lub dekrementacja 3. miniatura -> zmienna przyjmuje wartość numeru zdjęcia np. zmienna = parseInt(nazwa);
R.5.4	Po kliknięciu przycisku prev duże zdjęcie jest zmieniane na poprzednie lub w przypadku pierwszego - na piąte
R.5.5	Po kliknięciu przycisku next duże zdjęcie jest zmieniane na następne lub w przypadku piątego - na pierwsze
R.5.6	Po kliknięciu miniatury duże zdjęcie jest zmieniane na zdjęcie odpowiadające miniaturze (przetestować na zmianę z przyciskami)

Więcej materiałów znajdziesz na zawodowe.edu.pl