Nazwa kwalifikacji:	**Programowanie, tworzenie i administrowanie stronami internetowymi i bazami danych**
Oznaczenie kwalifikacji:	**EE.09**
Numer zadania:	**01**
Kod arkusza:	**EE.09-01-23.06-SG**
wersja arkusza	**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 obejmuje 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 danych do tabeli *psy.uzytkownicy* , czynność udokumentowano plikiem o nazwie *egzamin* w formacie PNG, na którym widoczna jest zawartość tabeli *uzytkownicy* lub informacja o imporcie danych zakończonym sukcesem
R.1.2	Zapisano plik tekstowy o nazwie *kwerendy* zawierający co najmniej jedno zapytanie SQL, wynikające z treści zadania

R.1.3	Wykonano zapytanie 1 tworzące tabelę *uzytkownicy* . Klucz główny id jest typu całkowitego dodatniego, jawnie zapisany oraz automatycznie inkrementowany. Klucz główny nie może przyjmować wartości pustych. Pola login i haslo są typu napisowego maksymalnie 50 znakowego. W pliku z kwerendami lub na zrzucie istnieje zapis np.:
	`CREATE TABLE uzytkownicy (`
	`id INTEGER UNSIGNED NOT NULL AUTO_INCREMENT,`
	`login VARCHAR(50) NULL,`
	`haslo VARCHAR(50) NULL,`
	`PRIMARY KEY(id));`
	oraz wynik działania zapytania udokumentowano zrzutem, na którym widoczny jest efekt poprawnie wykonanego zapytania

R.1.4	Wykonano zapytanie 2 dodające do tabeli użytkownika o loginie Grzegorz z hasłem „g” (bez szyfrowania). W pliku z kwerendami lub na zrzucie istnieje zapis np.: `INSERT INTO` `uzytkownicy` (`login`, `haslo`) `VALUES ('Grzegorz', 'g');` możliwe inne zapisy INSERT INTO oraz wynik działania zapytania udokumentowano zrzutem, na którym widoczne jest poprawnie wykonane zapytanie lub tabela z wstawionym rekordem
R.1.5	Wykonano zapytanie 3 wybierające jedynie pole login z tabeli uzytkownicy. W pliku z kwerendami lub na zrzucie istnieje zapis np.: `SELECT login FROM uzytkownicy;` oraz wynik działania zapytania udokumentowano zrzutem, na którym widoczne jest jedynie pole login i wypisane wszystkie loginy z tabeli (widocznych jest minimum 11 rekordów)
R.1.6	Wykonano zapytanie 4 wybierające jedynie pola id oraz login z tabeli uzytkownicy dla użytkowników, których hasło zaczyna się od cyfry 4. W pliku z kwerendami lub na zrzucie istnieje zapis np.: `SELECT id, login FROM uzytkownicy WHERE haslo LIKE "4%";` oraz wynik działania zapytania udokumentowano zrzutem, na którym widoczne są jedynie pola id i login dla loginów Heniek; Ola
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 *Jeśli witryna nie uruchamia się z powodu błędów PHP – sprawdzić wg kryteriów w kodzie źródłowym strony*
R.2.1	Plik *obraz.jpg* odbito poziomo (jak na obrazie 1 w arkuszu) oraz przeskalowano z zachowaniem proporcji tak, że jego wysokość wynosi dokładnie 500 px, odpowiadająca temu szerokość to 323 px ±5 px
R.2.2	Witrynę internetową zapisano w pliku o nazwie *logowanie.php*
R.2.3	Zastosowano właściwy standard kodowania polskich znaków dla witryny
R.2.4	Nadano tytuł strony: "Forum o psach”
R.2.5	Wygląd strony zdefiniowano dzieląc ją na bloki: baner, blok lewy oraz dwa bloki prawe, stopkę. Zastosowano znaczniki sekcji
R.2.6	W bloku banera zapisano nagłówek h1: "Forum wielbicieli psów" oraz w blokach prawych zapisano nagłówki h2: "Zapisz się" i "Zapraszamy wszystkich"
R.2.7	W bloku prawym pierwszym umieszczono formularz z trzema polami edycyjnymi i przyciskiem
R.2.8	W bloku prawym drugim umieszczono listę numerowaną z czterema elementami, zastosowano znaczniki ,
R.2.9	W bloku lewym umieszczono obraz o nazwie *obraz.jpg* z tekstem alternatywnym o treści „foksterier”

R.2.10	W stopce umieszczono napis „Stronę wykonał: ”, dalej wstawiony numer PESEL zdającego lub w przypadku jego braku inny numer, którym został podpisany arkusz
R.3	**Rezultat 3: Działanie witryny internetowej**
	Uwaga: Jeśli witryna nie uruchamia się z powodu błędów PHP – uruchomić plik lokalnie z dysku lub sprawdzić wg kryteriów w kodzie źródłowym strony
R.3.1	**Układ bloków** po uruchomieniu strony w przeglądarce jest zgodny z Obrazem 1 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 o treści „Przeczytaj regulamin forum”, którego kliknięcie powoduje przejście do strony *regulamin.html*
R.3.3	Formularz wysyła dane do tej samej strony metodą post, poprawnie również bez atrybutu action
R.3.4	Przynajmniej jedno pole edycyjne do wpisywania hasła jest typu password oraz przycisk wysyła dane z formularza
R.3.5	Strona zawiera działające połączenie z zewnętrznym arkuszem stylów o nazwie *styl4.css* , formatowanie pochodzi jedynie z tego arkusza Uwaga: kryterium to nie jest spełnione również, gdy źle przypisano 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
R.4.1	Ustawiono domyślny krój czcionki dla całej strony na Garamond (dla selektora * lub body lub html lub kontenera całej strony)
R.4.2	Ustawiono kolor tła SaddleBrown dla banera i stopki, Tan dla bloku lewego i obu prawych
R.4.3	Ustawiono biały kolor czcionki dla banera i stopki
R.4.4	Ustawiono wysokość bloków: 50px dla banera, 500px dla bloku lewego, 250px dla obu bloków prawych
R.4.5	Ustawiono szerokość bloku lewego 40% oraz obu prawych 60% (zastosowano np.: width lub grid-template-columns lub flex)
R.4.6	Ustawiono wyrównanie tekstu do środka dla banera
R.4.7	Ustawiono marginesy wewnętrzne (padding) dla formularza 10px (góra i dół) i 30px (lewy i prawy) oraz marginesy zewnętrzne (margin) dla pola edycyjnego na 5px
R.4.8	Ustawiono pozostałe cechy formularza: kolor tła SaddleBrown, biały kolor czcionki, szerokość 300px
R.4.9	Ustawiono dla paragrafu styl pseudoelementu before. Przed każdym paragrafem znajduje się tekst o treści "STATUS: " (content: "STATUS: ")
R.5	**Rezultat 5: Skrypt**
	Uwaga: Rezultat sprawdzić w wersji XAMPP 8.1.2, jeśli witryna nie uruchamia się z powodu błędów PHP - sprawdzić 5.1 ÷ 5.3, 5.8, 5.10 wg kryteriów w kodzie źródłowym strony. Kryteria 5.4 ÷ 5.7, 5.9 należy sprawdzić wpisując do formularza dane wprowadzanego użytkownika zgodnie z zapisanymi w kryteriach warunkami. Instrukcje muszą być zgodne ze składnią oraz zawierać dane wynikające z zadania. Kryteria uznajemy za spełnione jeżeli zastosowano biblioteki MySQLi lub PDO
R.5.1	Skrypt realizuje połączenie z serwerem bazy danych i wybór bazy *psy*
R.5.2	Ostatnią operacją na bazie jest jej zamknięcie
R.5.3	Skrypt zawiera instrukcję wysyłającą zapytanie 2 **lub** zapytanie 3
R.5.4	Wyświetlono komunikat „wypełnij wszystkie pola”, jeżeli nie wypełniono przynajmniej jednego pola formularza
R.5.5	Wyświetlono komunikat „login występuje w bazie danych, konto nie zostało dodane”, jeżeli podany login znajduje się w bazie danych
R.5.6	Wyświetlono komunikat „hasła nie są takie same, konto nie zostało dodane”, jeżeli hasła wpisane do pola „hasło” oraz pola „powtórz hasło” różnią się
R.5.7	Wstawiono do bazy danych rekord z danymi pobranymi z formularza, jeżeli login nie występuje w bazie a oba hasła są takie same. Należy sprawdzić w bazie danych czy wstawiono pole login i hasło
R.5.8	Zastosowano funkcję sha1 do zaszyfrowania hasła
R.5.9	Po dodaniu użytkownika wyświetlono komunikat „Konto zostało dodane”
R.5.10	Przynajmniej jeden komunikat jest zapisany w paragrafie