

Nazwa  
kwalifikacji:

**Programowanie, tworzenie i administrowanie stronami internetowymi i bazami danych**

Oznaczenie  
kwalifikacji:

**EE.09**

Numer zadania: **01**

Kod arkusza: **EE.09-01-SG\_zo**

Lp.	Elementy podlegające ocenie/kryteria oceny
<b>R.1</b>	<b>Rezultat 1: Operacje na bazie danych</b>
	<i>Uwaga: W przypadku oceny zrzutów należy uznać za prawidłowe jeżeli widoczny jest cały obszar ekranu, a zapytanie ma charakter uniwersalny dla każdego zestawu danych. Nie należy oceniać wykadrowanych zrzutów ekranu W R.1.3 ÷ R.1.5 zapisana liczba rekordów dotyczy stanu bazy bezpośrednio po imporcie, w przypadku innej liczby rekordów należy sprawdzić zapytanie zdającego w phpMyAdmin</i>
R.1.1	Wykonano import tabel do bazy danych dane, czynność udokumentowano plikiem o nazwie egzamin 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 imie, nazwisko, opis, zdjecie z tabeli osoby, dla wszystkich osób, których hobby ma id równe 1, 2, 6. W pliku z kwerendami lub na zrzucie istnieje zapis np.: <code>SELECT imie, nazwisko, opis, zdjecie FROM osoby WHERE Hobby_id = 1 OR Hobby_id = 2 OR Hobby_id = 6;</code> <u>oraz</u> wynik działania zapytania udokumentowano zrzutem, na którym widocznych jest dokładnie 13 rekordów, jedynie pola imie, nazwisko, opis, zdjecie
R.1.4	Utworzono zapytanie 2 wybierające jedynie pola id i nazwisko z tabeli osoby oraz odpowiadające im pole nazwa z tabeli hobby dla osób, które urodziły się po 2000 roku. W pliku z kwerendami lub na zrzucie istnieje zapis np.: <code>SELECT osoby.id, nazwisko, nazwa FROM osoby JOIN hobby ON osoby.hobby_id = hobby.id WHERE rok_urodzenia &gt; 2000;</code> możliwe też INNER JOIN lub bez JOIN z porównaniem kluczy w warunku <u>oraz</u> wynik działania zapytania udokumentowano zrzutem, na którym widocznych jest dokładnie 16 rekordów, jedynie pola osoby.id, nazwisko, nazwa
R.1.5	Utworzono zapytanie 3 wybierające jedynie pola id i zdjecie z tabeli osoby dla osób które mają na imię Monika. Kwerenda wybiera początkowe 5 wierszy, należy jawnie wskazać liczbę wierszy. W pliku z kwerendami lub na zrzucie istnieje zapis np.: <code>SELECT id, zdjecie FROM osoby WHERE imie='Monika' LIMIT 5;</code> <u>oraz</u> wynik działania zapytania udokumentowano zrzutem, na którym widocznych jest dokładnie 5 rekordów, dane: 2 osoba1.jpg; 4 osoba2.jpg; 7 osoba1.jpg; 12 osoba1.jpg; 17 osoba4.jpg
R.1.6	Utworzono zapytanie 4 tworzące tabelę o nazwie wpisy, kluczem głównym id typu całkowitego dodatniego oraz polach wpis typu tekstowego i uzytkownik_id typu całkowitego dodatniego. Klucz główny i pole uzytkownik_id nie mogą przyjmować wartości pustych. Klucz główny powinien być jawnie zapisany oraz automatycznie inkrementowany. W pliku z kwerendami lub na zrzucie istnieje zapis np.: <code>CREATE TABLE wpisy ( id INTEGER UNSIGNED NOT NULL AUTO_INCREMENT, wpis TEXT, (dopuszczalna dowolna nazwa dla pola tekstowego) uzytkownik_id INTEGER UNSIGNED NOT NULL, PRIMARY KEY(id) ); (dowolne typy całkowite i tekstowe)</code> <u>oraz</u> wynik działania zapytania udokumentowano zrzutem, na którym widoczny jest komunikat o poprawnie wykonanej kwerendzie lub utworzona tabela
<b>R.2</b>	<b>Rezultat 2: Zawartość witryny internetowej</b>
	<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 Jeśli witryna nie uruchamia się z powodu błędów PHP – sprawdzić wg kryteriów w kodzie źródłowym strony</i>

R.2.1	Plik <i>osoba1.jpg</i> zawiera widoczną twarz osoby, rozmiar zdjęcia dokładnie 100 px na 100 px
R.2.2	Witrynę internetową zapisano w pliku PHP o nazwie <i>lista</i>
R.2.3	Zastosowano właściwy standard kodowania polskich znaków dla witryny
R.2.4	Nadano tytuł strony: "Lista przyjaciół"
R.2.5	Wygląd strony zdefiniowano dzieląc ją na bloki: baner, główny i dwa bloki stopki. Zastosowano znaczniki sekcji
R.2.6	W bloku banera zapisano nagłówek h1: "Portal Społecznościowy - moje konto" oraz w bloku głównym nagłówki h2: "Moje zainteresowania" i "Moi znajomi"
R.2.7	W bloku głównym umieszczono listę punktowaną z trzema elementami, znaczniki <code>&lt;ul&gt;</code> , <code>&lt;li&gt;</code> oraz ich zamknięcia
R.2.8	W pierwszym bloku stopki umieszczono numer PESEL zdającego
<b>R.3</b>	<b>Rezultat 3: Działanie witryny internetowej</b>
	<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	<b>Układ bloków</b> po uruchomieniu strony w przeglądarce jest zgodny z Obrazem 2 w arkuszu egzaminacyjnym (nie uwzględnia się zestawu bloków zawierających: zdjęcie, opis, linia)
R.3.2	Na stronie umieszczono odnośnik pocztowy "napisz do mnie" łączący z adresem email ( <code>mailto:ja@portal.pl</code> )
R.3.3	Strona zawiera działające połączenie z zewnętrznym arkuszem stylów o nazwie <i>styl.css</i> , formatowanie pochodzi jedynie z tego arkusza <i>Uwaga: kryterium to nie jest spełnione również, gdy źle przypisano co najmniej jeden selektor do bloku (np. kropka zamiast #)</i>
<b>R.4</b>	<b>Rezultat 4: Styl CSS witryny internetowej</b>
	<i>Uwaga: W przypadku, gdy nie jest spełnione kryterium 3.3, kryteria w rezultacie R.4 należy ocenić w kodzie CSS (w pliku CSS lub znaczniku &lt;style&gt;). Składnia musi być zgodna ze specyfikacją CSS</i> <i>Kryteria R.4.2, R.4.3, R.4.6, R.4.7 są spełnione jedynie, gdy zdefiniowano styl dla selektora, nie są spełnione gdy zdefiniowano styl klasy lub inny</i>
R.4.1	Ustawiono domyślny krój czcionki dla całej strony na Georgia (dla selektora <code>* lub body lub html lub kontenera całej strony</code> )
R.4.2	Ustawiono kolor tła #005B4F dla banera i stopek, #E1E2E1 dla bloku głównego oraz #00897B dla selektora h2
R.4.3	Ustawiono biały kolor czcionki dla banera i stopek oraz selektora a
R.4.4	Ustawiono wysokość bloków dla banera i stopek na 20px oraz bloku głównego na 400px
R.4.5	Ustawiono szerokość stopek na 50% oraz bloku zawierającego zdjęcie na 20%
R.4.6	Ustawiono wyrównanie tekstu do środka dla banera i stopek oraz dla selektora h2
R.4.7	Ustawiono marginesy wewnętrzne (padding) dla bloku głównego i selektora <code>img</code> na 20px
R.4.8	Ustawiono rozmiar czcionki banera na 50%
R.4.9	Ustawiono przynajmniej jeden pasek przewijania (przesuwania) dla bloku głównego, np. <code>overflow: scroll;</code> (lub <code>auto</code> )
R.4.10	Ustawiono zakrzywienie obramowania 100% dla obrazu, np. <code>border-radius: 100%;</code> poprawne również dla wartości z przedziału <code>&lt;50%, 100%</code> )
<b>R.5</b>	<b>Rezultat 5: Skrypt</b>
	<i>Uwaga: cały rezultat sprawdzić w wersji XAMPP 7.4.8. Jeśli witryna nie uruchamia się z powodu błędów PHP - sprawdzić 5.1 - 5.4 wg kryteriów w kodzie źródłowym strony.</i> <i>Kryteria uznajemy za spełnione jeżeli zastosowano biblioteki MySQLi lub PDO</i>
R.5.1	Skrypt realizuje połączenie z serwerem bazy danych i wybór bazy <i>dane</i>
R.5.2	Ostatnią operacją na bazie jest jej zamknięcie (zgodne ze składnią)
R.5.3	Skrypt wysłał do bazy danych zapytanie 1
R.5.4	Dane zwróconego zapytaniem wiersza są podzielone na bloki zawierające: zdjęcie i opis obok siebie, pod spodem linia. W przypadku linii dopuszcza się użycie jej bez znacznika bloku. Zapis jest poprawny składniowo.
R.5.5	Wyświetlono obraz o nazwie pobranej z bazy danych oraz tekście alternatywnym „przyjaciel”
R.5.6	Wyświetlono nagłówek trzeciego stopnia z imieniem i nazwiskiem pobranym z bazy danych oraz paragraf (akapit) o treści: „Ostatni wpis: <opis>”, gdzie pole <opis> jest pobrane z bazy danych
R.5.7	Wyświetlono wszystkie wiersze zwrócone zapytaniem w oparciu o pętlę