Nazwa kwalifikacji:	**Programowanie, tworzenie i administrowanie stronami internetowymi i bazami danych**
Oznaczenie kwalifikacji:	**EE.09**
Numer zadania:	**02**
Kod arkusza:	**EE.09-02-24.01-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 *egzamin*, czynność udokumentowano plikiem o nazwie *import* w formacie PNG
R.1.2	Zapisano plik tekstowy o nazwie *kwerendy* z zapytaniami SQL zawierający co najmniej jedno zapytanie SQL, wynikające z treści zadania
R.1.3	Wykonano zapytanie 1 wybierające jedynie pola czas, kierunek, nr_rejsu i status_lotu z tabeli *przyloty* posortowane rosnąco według czasu. W pliku z kwerendami lub na zrzucie istnieje zapis np.: `SELECT czas, kierunek, nr_rejsu, status_lotu FROM przyloty ORDER BY czas;` (też gdy zapisano `ASC`) oraz wynik działania zapytania udokumentowano zrzutem, na którym jest widocznych dokładnie 8 rekordów, jedynie pola czas, kierunek, nr_rejsu, status_lotu. Rekordy są posortowane rosnąco wg. czasu
R.1.4	Wykonano zapytanie 2 zliczające przyloty, które odbyły się w godzinach 10:00 – 10:59. W pliku z kwerendami lub na zrzucie istnieje zapis np.: `SELECT COUNT(*) FROM przyloty WHERE czas LIKE "10%";` możliwe inne warianty, np. z BETWEEN oraz wynik działania zapytania udokumentowano zrzutem, na którym jest widoczna wartość 7, wynikająca z zadanego zapytania
R.1.5	Wykonano zapytanie 3 wybierające jedynie pola nr_rejsu z tabeli *przyloty* oraz typ z tabeli *samoloty*, dla lotów których kierunkiem jest Warszawa. W pliku z kwerendami lub na zrzucie istnieje zapis np.: `SELECT nr_rejsu, samoloty.typ FROM przyloty JOIN samoloty ON przyloty.samoloty_id = samoloty.id WHERE kierunek = "Warszawa";` możliwe INNER JOIN lub bez JOIN (porównanie kluczy w warunku) oraz wynik działania zapytania udokumentowano zrzutem, na którym widoczny jest dokładnie jeden rekord jedynie z danymi: W63454, Boening 737
R.1.6	Wykonano zapytanie 4 aktualizujące w tabeli *odloty* pole status_lotu na: „opoznienie 20 min” dla lotu o numerze rejsu LX5673. W pliku z kwerendami lub na zrzucie istnieje zapis np.: `UPDATE odloty SET status_lotu="opoznienie 20 min" WHERE nr_rejsu="LX5673";` oraz wynik działania zapytania udokumentowano zrzutem, dokumentującym poprawne wykonanie zapytania 4
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 *zad5.png* przeskalowano z zachowaniem proporcji do wysokości 200 px, odpowiadająca temu szerokość wynosi 305 px (± 5 px); przezroczystość została zachowana
R.2.2	Witrynę internetową zapisano w pliku PHP o nazwie *lotnisko* oraz jawnie zastosowano właściwy standard kodowania polskich znaków
R.2.3	Nadano tytuł strony: "Port Lotniczy"
R.2.4	Wygląd strony zdefiniowano dzieląc ją na bloki: trzy bloki banera, poniżej blok główny, następnie dwa bloki stopki. Zastosowano znaczniki sekcji
R.2.5	W drugim bloku banera zapisano nagłówek h1: "Przyloty", w trzecim bloku banera nagłówek h3: "przydatne linki" oraz w pierwszym bloku stopki zapisano paragraf (w kodzie HTML lub skrypcie)
R.2.6	W pierwszym bloku banera umieszczono obraz *zad5.png* z tekstem alternatywnym: "logo lotnisko"
R.2.7	W bloku głównym umieszczono tabelę o czterech kolumnach, zastosowano znaczniki <table>, <tr>, <td>, tabela generowana skryptem lub HTML
R.2.8	Umieszczono w tabeli wiersz nagłówkowy, zastosowano cztery znaczniki <th> wewnątrz pierwszego wiersza tabeli
R.2.9	W pierwszym bloku stopki zastosowano znaczniki lub do pochylenia oraz , lub do pogrubienia tekstu (w skrypcie lub w kodzie HTML). Możliwe alternatywne rozwiązanie za pomocą stylów CSS
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 w pełni zgodny z obrazem 2 w arkuszu egzaminacyjnym (prawidłowo zastosowano właściwości CSS układające bloki na stronie, np. float i clear albo display flex albo grid albo position)
R.3.2	Na stronie umieszczono odnośnik, którego wybranie prowadzi do pliku *kwerendy.txt*
R.3.3	Odnośnik otwiera się w nowym oknie
R.3.4	Strona zawiera działające połączenie z zewnętrznym arkuszem stylów o nazwie *styl5.css* , formatowanie CSS 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 3.4, kryteria w rezultacie R.4 należy ocenić w kodzie CSS (w pliku CSS lub znaczniku <style>). Składnia musi być zgodna ze specyfikacją CSS *rgb(30, 144,255) = #1E90FF; rgb(0, 0, 128) = #000080; rgb(192, 192, 192) = #C0C0C0*
R.4.1	Ustawiono krój czcionki Garamond dla body
R.4.2	Ustawiono kolor tła rgb(30, 144, 255) dla body, oraz kolor tła rgb(0, 0, 128) dla bloku głównego
R.4.3	Ustawiono biały kolor czcionki dla body
R.4.4	Ustawiono wysokość wszystkich bloków banera na 200px
R.4.5	Ustawiono szerokość pierwszego i trzeciego bloku banera na 25%, drugiego bloku banera oraz obu bloków stopki na 50% oraz szerokość tabeli na 100%
R.4.6	Ustawiono wyrównanie tekstu do środka dla drugiego bloku banera, pierwszego bloku stopki oraz wyrównanie do prawej strony dla drugiego bloku stopki
R.4.7	Ustawiono marginesy wewnętrzne (padding) na: 60px dla bloku głównego, 20px dla paragrafu, 10px dla komórki tabeli i komórki nagłówkowej oraz jedynie margines wewnętrzny górny (padding-top) 100px dla drugiego bloku stopki
R.4.8	Ustawiono obramowanie selektora p, td, th: 1px linią ciągłą rgb(192, 192, 192) oraz połączone obramowanie dla selektora tabeli (border-collapse: collapse;)
R.4.9	W momencie najechania kursorem na paragraf, jego kolor tła zmienia się na rgb(192, 192, 192)
R.4.10	Ustawiono rozmiar czcionki 250% dla drugiego bloku banera, 200% dla paragrafu p oraz tekst wielkimi literami (text-transform: uppercase;) dla bloku głównego

Arkusz pobrany z zawodowe.edu.pl

R.5	**Rezultat 5: Skrypt połączenia z bazą**
	Uwaga: rezultat R.5 sprawdzić w wersji XAMPP 8.1.2. Jeśli witryna nie uruchamia się z powodu błędów PHP - sprawdzić 5.1 - 5.4, 5.6, 5.7 wg kryteriów w kodzie źródłowym strony. 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 napisano w języku PHP oraz zawiera przynajmniej jedną poprawną składniowo instrukcję, wynikającą z treści zadania
R.5.2	Skrypt 1 realizuje połączenie z serwerem bazy danych i wybór bazy *egzamin*
R.5.3	Po wykonaniu operacji na bazie zastosowano funkcję zamknięcia bazy
R.5.4	Skrypt 1 wysyła do bazy zapytanie nr 1
R.5.5	Dane otrzymane z bazy są umieszczane w kolejnych wierszach tabeli, w odpowiednich komórkach, tak jak na Obrazie 2
R.5.6	Strona wykorzystuje mechanizm ciasteczek, zastosowano funkcję PHP setcookie, sprawdzono czy ciasteczko jest ustawione (np. isset, empty) oraz zastosowano tablicę $_COOKIE, zapis musi być poprawny składniowo i logicznie
R.5.7	Gdy odwiedzający jest na stronie pierwszy raz zakładane jest ciasteczko z czasem trwania 2 godziny od utworzenia
R.5.8	Gdy odwiedzający jest na stronie pierwszy raz wyświetlany jest komunikat "Dzień dobry! Strona lotniska używa ciasteczek” w przeciwnym wypadku wyświetlany jest komunikat „Witaj ponownie na stronie lotniska"

Darmowe egzaminy zawodowe: zawodowe.edu.pl