

Nazwa kwalifikacji: **Programowanie, tworzenie i administrowanie stronami internetowymi i bazami danych**

Oznaczenie kwalifikacji: **INF.03**

Numer zadania: 01

Wypełnia zdający

Numer PESEL zdającego

--	--	--	--	--	--	--	--	--	--	--

Czas trwania egzaminu: 150 minut

**PRÓBNY EGZAMIN  
POTWIERDZAJĄCY KWALIFIKACJE W ZAWODZIE  
CZĘŚĆ PRAKTYCZNA**

**Instrukcja dla zdającego**

1. Na pierwszej stronie arkusza egzaminacyjnego wpisz w oznaczonym miejscu swój numer PESEL.
2. Na KARCIE OCENY w oznaczonym miejscu przyklej naklejkę z numerem PESEL oraz wpisz:
  - swój numer PESEL\*,
  - symbol cyfrowy zawodu,
  - oznaczenie kwalifikacji,
  - numer zadania,
  - numer stanowiska.
3. KARTĘ OCENY przekaz zespołowi nadzorującemu część praktyczną egzaminu.
4. Sprawdź, czy arkusz egzaminacyjny zawiera 7 stron i nie zawiera błędów. Ewentualny brak stron lub inne usterki zgłoś przez podniesienie ręki przewodniczącemu zespołu nadzorującego część praktyczną egzaminu.
5. Zapoznaj się z treścią zadania oraz stanowiskiem egzaminacyjnym. Masz na to 10 minut. Czas ten nie jest wliczany do czasu trwania egzaminu.
6. Czas rozpoczęcia i zakończenia pracy zapisze w widocznym miejscu przewodniczący zespołu nadzorującego.
7. Wykonaj samodzielnie zadanie egzaminacyjne. Przestrzegaj zasad bezpieczeństwa i organizacji pracy.
8. Jeżeli w zadaniu egzaminacyjnym występuje polecenie „zgłoś gotowość do oceny przez podniesienie ręki”, to zastosuj się do polecenia i poczekaj na decyzję przewodniczącego zespołu nadzorującego.
9. Po zakończeniu wykonania zadania pozostaw rezultaty oraz arkusz egzaminacyjny na swoim stanowisku lub w miejscu wskazanym przez przewodniczącego zespołu nadzorującego.
10. Po uzyskaniu zgody zespołu nadzorującego możesz opuścić salę/miejsce przeprowadzania egzaminu.

***Powodzenia!***

## Zadanie egzaminacyjne

Wykonaj aplikację internetową weryfikującą poprawność wpisywanych do bazy danych pracowników firmy. Wykorzystaj do tego celu edytor zaznaczający składnię oraz program XAMPP.

Aby wykonać zadanie, zaloguj się na konto Egzamin bez hasła. Na pulpicie znajdziesz archiwum o nazwie materiały.zip zabezpieczone hasłem: EgZ@Min

Na pulpicie konta Egzamin utwórz folder. Jako nazwy folderu użyj swojego numeru PESEL lub innego numeru, którym został podpisany arkusz. Umieść w tym folderze rozpakowane pliki. Po skończonej pracy wyniki zapisz również w tym folderze.

## Operacje na bazie danych

Do wykonania operacji na bazie należy wykorzystać przedstawioną na Obrazie 1 tabelę.

id	nazwisko	imie	PESEL
----	----------	------	-------

Obraz 1. Tabela bazy danych *firma*

Uruchom usługi MySQL i Apache za pomocą XAMPP Control Panel. Używając narzędzia phpMyAdmin wykonaj podane operacje na bazie danych:

- utwórz bazę danych o nazwie *firma*
- do bazy zaimportuj tabele z pliku *pracownicy.sql*. Wykonaj zrzut ekranu po imporcie. Zrzut zapisz w formacie JPEG o nazwie *import* w folderze opisanym numerem PESEL lub innym numerem, którym został podpisany arkusz. Nie kadruj zrzutu. Zrzut powinien obejmować cały ekran monitora, z widocznym paskiem zadań. Na zrzucie powinny być widoczne elementy wskazujące na poprawnie wykonany import tabel.
- zapisz i wykonaj zapytania SQL działające na bazie *firma*. Zapytania zapisz w pliku *zapytania.txt*, w folderze z numerem PESEL lub innym numerem, którym został podpisany arkusz. Wykonaj zrzuty ekranu przedstawiające wyniki działania kwerend. Zrzuty zapisz w formacie JPG i nadaj im nazwy *zap1*, *zap2*, *zap3*, *zap4*. Zrzuty powinny obejmować cały ekran monitora, z widocznym paskiem zadań.
  - Zapytanie 1: wybierające wszystkie dane pracowników z tabeli *pracownicy*.
  - Zapytanie 2: wybierające wyłącznie dane pracownika o nazwisku *Król* z tabeli *pracownicy*.
  - Zapytanie 3: zliczające i podające w kolumnie o nazwie *ilość pracowników* ilość rekordów z danymi znajdującymi się w tabeli *pracownicy*.
  - Zapytanie 4: wstawiające do tabeli *pracownicy* rekord z danymi pracownika: *Rak*, *Monika*, *00101843690*, gdzie odpowiednio danymi są nazwisko, imię, PESEL.

## Przygotowanie witryny internetowej

Projekt witryny powinien składać się ze strony głównej *index.php* oraz *formularz.php*

Strona główna *index.php*



Obraz 2. Strona główna aplikacji – plik *index.php*

### Cechy wspólne dla obu plikówstrony:

- zastosowany właściwy standard kodowania polskich znaków
- tytuł strony *Dane osobowe* widoczny w karcie przeglądarki
- podłączony plik z arkuszem stylów *style.css*
- podział stron na bloki: baner, poniżej blok lewy i prawy, na dole stopka. Podział dokonywany jest za pomocą znaczników sekcji. Rezultat powinien być podobny do strony z Obrazu 2.
- baner z nagłówkiem pierwszego stopnia o treści „*Dane osobowe pracowników*”
- w bloku lewym lista punktowana z odnośnikami:
  - *Wyświetl dane* powiązany z plikiem *index.php*
  - *Wpisz dane* powiązany z plikiem *formularz.php*
- stopka składa się z nagłówka piątego stopnia o treści: „*AUTOR strony:* ” i numeru PESEL zdającego lub innego numeru którym został podpisany arkusz

### Cechy indywidualne dla pliku *index.php*:

- w bloku prawym:
  - nagłówek drugiego stopnia o treści „*Dane osobowe*”
  - tabela generowana skryptem 1

Cechy indywidualne dla pliku *formularz.php* (jak na obrazie 3):

- w bloku prawym:
  - nagłówek drugiego stopnia o treści „Podaj dane”
  - formularz składający się z pól edycyjnych wraz z etykietami *Nazwisko*, *Imię* i *PESEL* oraz przycisku *Sprawdź i zapisz* wysyłającymi wpisane dane do skryptu 2
  - panel z wynikami błędów lub informacją o zapisie do bazy danych generowany skryptem 2

Dane osobowe pracowników

Wyświetl dane

Wpisz dane

Podaj dane

Nazwisko:

Imię:

PESEL:

Sprawdź i zapisz

brak nazwiska
brak imienia
brak numeru pesel

Autor strony: xxxxxxxxxxxx

Obraz3. Strona główna aplikacji – plik *formularz.php*

### Styl CSS projektowanej witryny:

Plik *style.css* zawiera formatowanie dla:

- całej strony: krój czcionki *Garamond*, wyśrodkowanie tekstu
- bloku banera i stopki: kolor tła #0451b0, kolor czcionki biały, wysokość 70 px, rozmiar czcionki 150 %
- bloku lewego: kolor tła #5ca4fd, wysokość 600 px, szerokość 35 %
- bloku prawego: kolor tła #8ec0fd, wysokość 600px, szerokość 65%
- znacznika tabeli: margines zewnętrzny auto, obramowanie w kolorze czarnym, ciągłe o grubości 1 px
- znacznika komórki tabeli: obramowanie w kolorze czarnym, ciągłe o grubości 1 px
- znacznika komórki nagłówkowej tabeli: obramowanie w kolorze czarnym, ciągłe o grubości 1 px
- listy punktowanej: brak symboli punktatorów, szerokość 180 px
- odnośników w liście wypunktowanej: kolor czcionki czarny, margines wewnętrzny górny i dolny 8 px, margines wewnętrzny prawy i lewy 16 px, margines zewnętrzny 10 px, brak podkreślenia

odnośnika, obramowanie w kolorze czarnym, ciągłe o grubości 1 px, zaokrąglenie rogów obramowania 10 px

- pseudoklasy odnośników: po najechaniu myszą na element jego tło zmienia się na #0451b0
- paneluz wynikami błędów lub informacją o zapisie do bazy danych: margines zewnętrzny górny 100 px, obramowanie w kolorze czarnym, ciągłe o grubości 1 px

## Skrypty:

### Wymagania dotyczące skryptu 1:

- wykonywany po stronie serwera
- łączy się z serwerem bazodanowym na *localhost*, użytkownik *root* bez hasła, baza danych o nazwie *firma*
- wysyła do bazy zapytanie 1 i pobiera z bazy danych dane pracowników (patrz „Operacje na bazie danych”).
- generuje tabelę z wierszem nagłówkowym z komórkami *Id*, *Nazwisko*, *Imię*, *PESEL* oraz wypełnia tabelę danymi pobranymi z bazy danych (jak na obrazie 2)

### Wymagania dotyczące skryptu 2:

- wykonywany po stronie serwera
- łączy się z serwerem bazodanowym na *localhost*, użytkownik *root* bez hasła, baza danych o nazwie *firma*
- w przypadku niewpisania w formularzu nazwiska wyświetla napis o treści *brak nazwiska*
- w przypadku niewpisania w formularzu imienia wyświetla napis o treści *brak imienia*
- w przypadku niewpisania w formularzu numeru PESEL wyświetla napis o treści *brak numeru PESEL*
- weryfikuje poprawność wpisanego numeru PESEL wyliczając sumę kontrolną - 11 cyfrę PESEL. (Algorytm wyliczania sumy kontrolnej znajduje się poniżej)
- po poprawnej weryfikacji wyświetla napis „Zapisuję do bazy <nazwisko> <imię> <PESEL>” (gdzie <nazwisko> <imię> <PESEL> są danymi pobranymi z formularza) i wysyła odpowiednio zmodyfikowane (uzupełnione danymi pobranymi z formularza) zapytanie 4 (patrz „Operacje na bazie danych”).

### **ALGORYTM WYLICZANIA SUMY KONTROLNEJ PESEL:**

PESEL składa się z 11 cyfr. Ostatnia 11 cyfra jest sumą kontrolną obliczaną według algorytmu.

Krok 1

Policz sumę iloczynów poszczególnych cyfr i odpowiadających im wag wg wzoru:

$$S = 1*cyfra1 + 3*cyfra2 + 7*cyfra3 + 9*cyfra4 + 1*cyfra5 + 3*cyfra6 + 7*cyfra7 + 9*cyfra8 + 1*cyfra9 + 3*cyfra10$$

Krok 2

Oblicz resztę z dzielenia przez 10 z otrzymanej sumy S (operacja modulo)

$$M = S \% 10$$

### Krok 3

Przypadek 1

Jeśli liczba M równa jest 0 to suma kontrolna (11 cyfra PESEL) równa jest 0

Przypadek 2

Jeśli liczba M jest różna od 0 to suma kontrolna jest wynikiem odejmowania  $10 - M$

#### **Przykład 1 dla PESEL 00101843690**

$$\begin{aligned} S &= 1*0 + 3*0 + 7*1 + 9*0 + 1*1 + 3*8 + 7*4 + 9*3 + 1*6 + 3*9 = \\ &= 0 + 0 + 7 + 0 + 1 + 24 + 28 + 27 + 6 + 27 = 120 \end{aligned}$$

$$M = 120 \% 10 = 0 \text{ (M równa 0 więc przypadek 1)}$$

11 cyfra PESEL = 0

#### **Przykład 2 dla PESEL 04061656044**

$$\begin{aligned} S &= 1*0 + 3*4 + 7*0 + 9*6 + 1*1 + 3*6 + 7*5 + 9*6 + 1*0 + 3*4 = \\ &= 0 + 12 + 0 + 54 + 1 + 18 + 35 + 54 + 0 + 12 = 186 \end{aligned}$$

$$M = 186 \% 10 = 6 \text{ (M różne od 0 więc przypadek 2)}$$

11 cyfra PESEL =  $10 - M = 10 - 6 = \underline{4}$

### **UWAGI:**

Po zakończeniu wykonywania zadania w folderze z numerem pesel lub innym numerem, którym został podpisany arkusz, powinny znajdować się:

- pliki graficzne: *import.jpg*, *zap1.jpg*, *zap2.jpg*, *zap3.jpg*, *zap4.jpg*
- plik tekstowy z zapytaniami SQL: *zapytania.txt*
- plik z arkuszem stylów: *styl.css*
- pliki z kodem witryny: *index.php* i *formularz.php*

**Tabela 1. Wybór funkcji języka PHP do obsługi bazy MySQL i MariaDB**

<b>Funkcje biblioteki MySQLi</b>	<b>Zwracana wartość</b>
<i>mysqli_connect</i> ( <i>serwer, użytkownik, hasło, nazwa_bazy</i> )	id połączenia lub FALSE, gdy niepowodzenie
<i>mysqli_select_db</i> ( <i>id_polaczenia, nazwa_bazy</i> )	TRUE/FALSE w zależności od stanu operacji
<i>mysqli_error</i> ( <i>id_polaczenia</i> )	Tekst komunikatu błędu
<i>mysqli_close</i> ( <i>id_polaczenia</i> )	TRUE/FALSE w zależności od stanu operacji
<i>mysqli_query</i> ( <i>id_polaczenia, zapytanie</i> )	Wynik zapytania
<i>mysqli_fetch_row</i> ( <i>wynik_zapytania</i> )	Tablica numeryczna odpowiadająca wierszowi zapytania
<i>mysqli_fetch_array</i> ( <i>wynik_zapytania</i> )	Tablica zawierająca kolejny wiersz z podanych w wyniku zapytania lub FALSE, jeżeli nie ma więcej wierszy w wyniku zapytania
<i>mysqli_num_rows</i> ( <i>wynik_zapytania</i> )	Liczba wierszy w podanym zapytaniu
<i>mysqli_num_fields</i> ( <i>wynik_zapytania</i> )	Liczba kolumn w podanym zapytaniu

**Tabela 2. Spis wartości atrybutu *type* znacznika <input>**

<b>Atrybut <i>type</i></b>	<b>Opis</b>
<i>button</i>	Przycisk
<i>checkbox</i>	Pole wielokrotnego wyboru
<i>color</i>	Paleta kolorów z możliwością wyboru koloru
<i>date</i>	Pole edycyjne do wpisywania daty
<i>datetime-local</i>	Pole edycyjne do wpisywania daty i czasu
<i>email</i>	Pole edycyjne do wpisywania adresu e-mail
<i>file</i>	Przycisk z możliwością wyboru pliku
<i>hidden</i>	Ukryte pole z możliwością przechowywania danych
<i>month</i>	Pole z możliwością wyboru miesiąca i roku
<i>number</i>	Pole edycyjne do wpisywania numerów
<i>password</i>	Pole edycyjne do wpisywania haseł
<i>radio</i>	Pole jednokrotnego wyboru
<i>reset</i>	Przycisk czyszczący formularz
<i>submit</i>	Przycisk wysyłający dane z formularza
<i>tel</i>	Pole edycyjne do wpisywania numerów telefonów
<i>text</i>	Pole edycyjne do wpisywania tekstów
<i>time</i>	Pole edycyjne do wpisywania czasu

