

Nazwa kwalifikacji: **Tworzenie i administrowanie stronami i aplikacjami internetowymi oraz bazami danych**
Symbol kwalifikacji: **INF.03**
Numer zadania: **05**
Wersja arkusza: **SG**

Wypełnia zdający

Numer PESEL zdającego*

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

Miejsce na naklejkę z numerem
PESEL i z kodem ośrodka

Czas trwania egzaminu: **150** minut.

INF.03-05-25.06-SG

EGZAMIN ZAWODOWY

Rok 2025

CZĘŚĆ PRAKTYCZNA

**PODSTAWA PROGRAMOWA
2019**

Instrukcja dla zdającego

1. Na pierwszej stronie arkusza egzaminacyjnego wpisz w oznaczonym miejscu swój numer PESEL i naklej naklejkę z numerem PESEL i z kodem ośrodka.
2. Na KARCIE OCENY w oznaczonym miejscu przyklej naklejkę z numerem PESEL oraz wpisz:
 - swój numer PESEL*,
 - oznaczenie kwalifikacji,
 - numer zadania,
 - numer stanowiska.
3. Sprawdź, czy arkusz egzaminacyjny zawiera 6 stron i nie zawiera błędów. Ewentualny brak stron lub inne usterki zgłoś przez podniesienie ręki przewodniczącemu zespołu nadzorującego.
4. Zapoznaj się z treścią zadania oraz stanowiskiem egzaminacyjnym. Masz na to 10 minut. Czas ten nie jest wliczany do czasu trwania egzaminu.
5. Czas rozpoczęcia i zakończenia pracy zapisze w widocznym miejscu przewodniczący zespołu nadzorującego.
6. Wykonaj samodzielnie zadanie egzaminacyjne. Przestrzegaj zasad bezpieczeństwa i organizacji pracy.
7. Po zakończeniu wykonania zadania pozostaw arkusz egzaminacyjny z rezultatami oraz KARTĘ OCENY na swoim stanowisku lub w miejscu wskazanym przez przewodniczącego zespołu nadzorującego.
8. Po uzyskaniu zgody zespołu nadzorującego możesz opuścić salę/miejsce przeprowadzania egzaminu.

Powodzenia!

* w przypadku braku numeru PESEL – seria i numer paszportu lub innego dokumentu potwierdzającego tożsamość



Zadanie egzaminacyjne

UWAGA: numer, którym został podpisany arkusz egzaminacyjny (PESEL lub w przypadku jego braku numer paszportu) jest w zadaniu nazywany numerem zdającego.

Wykonaj aplikację internetową salonu fotograficznego, wykorzystując edytor grafiki rastrowej, pakiet XAMPP oraz edytor zaznaczający składnię.

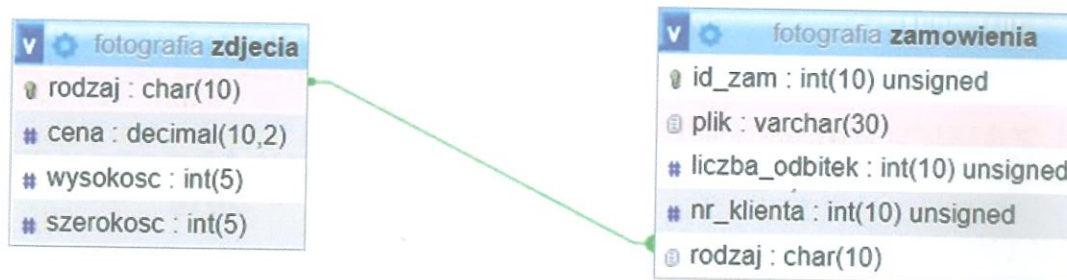
Aby wykonać zadanie, należy zalogować się na konto **Egzamin** bez hasła. Na pulpicie znajduje się archiwum 7z o nazwie *pliki6* zabezpieczone hasłem: **#Foto+Foto&**.

Archiwum należy rozpakować.

Na pulpicie konta **Egzamin** należy utworzyć folder. Jako nazwy folderu należy użyć numeru zdającego. Wszystkie wyniki należy zapisać w tym folderze.

Operacje na bazie danych

Do wykonania operacji na bazie należy wykorzystać przedstawione na ilustracji 1 tabele. Kluczem obcym dla relacji jest pole rodzaj w tabeli *zamowienia*.

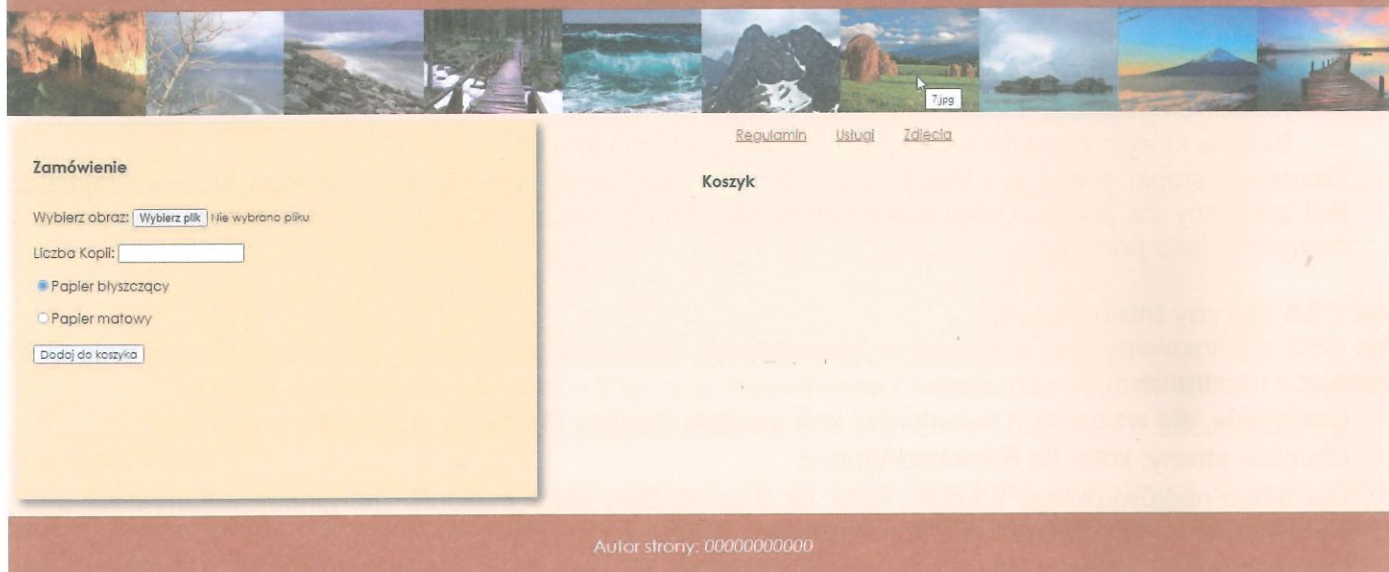


Ilustracja 1. Baza danych

Za pomocą narzędzia phpMyAdmin wykonaj następujące operacje na bazie danych:

- Utwórz bazę danych o nazwie *fotografia*, z zestawem polskich znaków (np. utf8_unicode_ci)
- Z rozpakowanego archiwum zaimportuj tabele z pliku *fotografia.sql*
- Wykonaj zrzut ekranu po imporcie. Zapisz zrzut w formacie PNG pod nazwą *import*. Nie kadruj zrzutu. Powinien on obejmować cały ekran monitora, z widocznym paskiem zadań. Na zrzucie powinny być widoczne elementy wskazujące na poprawnie wykonany import tabel
- Wykonaj zapytania SQL działające na bazie *fotografia*. Zapytania zapisz w pliku *kwerendy.txt*. Wykonaj zrzuty ekranu przedstawiające wyniki działania kwerend. Zrzuty zapisz w formacie PNG i nadaj im nazwy *kw1*, *kw2*, *kw3*, *kw4*. Zrzuty powinny obejmować cały ekran monitora z widocznym paskiem zadań
 - Zapytanie 1: wybierające jedynie pola plik, nr_klienta, liczba_odbitek dla zamówień na papierze matowym, z liczbą odbitek większą od 100
 - Zapytanie 2: wybierające dla klienta o identyfikatorze 3 wartości jego zamówień, obliczane na podstawie liczby odbitek i ceny, nazwa kolumny (alias) „Do zapłaty”. Należy posłużyć się relacją
 - Zapytanie 3: zliczające sumę wszystkich odbitek na papierze błyszczącym
 - Zapytanie 4: tworzące konto użytkownika **Anna** na *localhost* z hasłem **@NNa**

Fotografia artystyczna



Ilustracja 2. Wygląd strony internetowej w przeglądarce Chrome, widać dymek dla obrazu 7.jpg

Cechy grafiki 10.jpg:

- Grafikę należy przeskalować z zachowaniem proporcji do wysokości 200 px

Cechy witryny:

- Składa się z podstron o nazwach *zamowienie.html*, *regulamin.html*, *uslugi.html*
- W pliku *regulamin.html* należy zapisać jedynie: „Regulamin”
- W pliku *uslugi.html* należy zapisać jedynie: „Usługi”

Cechy strony *zamowienie.html*:

- Zapisana w języku HTML5
- Zadeklarowany polski język zawartości witryny
- Jawnie zastosowany właściwy standard kodowania polskich znaków
- Tytuł strony widoczny na karcie przeglądarki: „Fotografia”
- Arkusz stylów w pliku o nazwie *style.css* prawidłowo połączony z kodem strony
- Podział strony na bloki zrealizowany za pomocą semantycznych znaczników bloków języka HTML5 tak, aby po uruchomieniu w przeglądarce układ bloków na stronie był zgodny z ilustracją 3
- Zawartość bloku nagłówkowego: nagłówek pierwszego stopnia o treści „Fotografia artystyczna”
- Zawartość bloku sekcji obrazów: 10 obrazów od *1.jpg* do *10.jpg*. Wszystkie obrazy mają tekst alternatywny „obrazy do sprzedaży” oraz tekst dymka podpowiedzi taki sam jak nazwa pliku, np. *1.jpg*
- Zawartość bloku głównego: bloki sekcji lewej, nawigacyjny i sekcji prawej
- Zawartość sekcji lewej:
 - Nagłówek trzeciego stopnia o treści „Zamówienie”
 - Etykieta „Wybierz obraz:”, powiązana z polem edycyjnym znajdującym się obok
 - Pole edycyjne przeznaczone do wyboru pliku. Akceptuje tylko obrazy w formacie JPEG
 - Poniżej etykieta „Liczba Kopii:”, powiązana z polem edycyjnym znajdującym się obok
 - Pole edycyjne przeznaczone do wpisywania jedynie liczb
 - Poniżej pole opcji „Papier błyszczący”, domyślnie zaznaczone
 - Poniżej pole opcji „Papier matowy”
 - W jednym momencie można zaznaczyć tylko jedno pole opcji
 - Przycisk „Dodaj do koszyka”, którego kliknięcie wywołuje skrypt



Ilustracja 3. Układ bloków

- Zawartość bloku nawigacyjnego:
 - Lista punktowana (nieuporządkowana) z elementami:
 - Odnośnik prowadzący do podstrony *regulamin.html* o treści „Regulamin”
 - Odnośnik prowadzący do podstrony *uslugi.html* o treści „Usługi”
 - Odnośnik prowadzący do strony internetowej *https://pixabay.com* o treści „Zdjęcia”
- Zawartość sekcji prawej:
 - Nagłówek trzeciego stopnia o treści „Koszyk”
 - Blok, w którym zostanie umieszczony efekt działania skryptu
- Zawartość stopki: paragraf o treści „Autor strony: ”, dalej wstawiony numer zdającego. Numer zdającego jest zapisany za pomocą znacznika semantycznego oznaczającego tekst uwypuklony, formatowany domyślnie jako pochylony

Styl CSS witryny internetowej

Styl CSS zdefiniowany jest w całości w zewnętrznym pliku o nazwie *style.css*. Cechy formatowania CSS działające na stronie:

- Domyślnie, dla wszystkich selektorów: krój czcionki Century Gothic, w przypadku braku Arial
- Dla ciała strony: kolor tła BlancheAlmond
- Dla bloku nagłówkowego i stopki: kolor tła Sienna, biały kolor czcionki, marginesy wewnętrzne 5 px, wyrównanie tekstu do środka, rozmiar czcionki 130%
- Dla sekcji lewej: kolor tła NavajoWhite, marginesy zewnętrzne 10 px, wewnętrzne 20 px, szerokość 35%, wysokość 400 px, cień o przesunięciu 5 px w obu osiach, rozmyciu 10 px i kolorze DimGray
- Dla bloku nawigacyjnego: szerokość 50%, wysokość 50 px
- Dla sekcji prawej: szerokość 50%, wysokość 420 px, paski przewijania pojawiające się tylko w przypadku przepełnienia bloku
- Dla pola edycyjnego i przycisku: jedynie zewnętrzny margines górny 20 px
- Dla selektora elementu listy: sposób wyświetlania liniowo-blokowy
- Dla selektora obrazu: szerokość 10%, opływanie po prawej stronie (obraz po lewej stronie)
- Dla selektora odnośnika: kolor czcionki Sienna, jedynie zewnętrzny margines prawy 30 px
- W momencie, gdy kursor znajdzie się na odnośniku jego kolor czcionki zamienia się na #4C1900

Uwaga: styl CSS elementu listy, obrazu i odnośnika należy zdefiniować wyłącznie przy pomocy selektora tych znaczników. Jest to uwarunkowane projektem późniejszej rozbudowy witryny.

Skrypt

W tabeli 1 zamieszczono wybrane pola i metody modelu DOM w języku JavaScript. Wymagania dotyczące skryptu:

- Wykonywany po stronie klienta, po kliknięciu przycisku
- Należy stosować znaczące nazewnictwo zmiennych i funkcji w języku polskim lub angielskim
- Pobiera dane z kontrolki
- Oblicza cenę na podstawie liczby kopii i rodzaju papieru. Dla papieru błyszczącego cena jednostkowa wynosi 1,5 zł, dla papieru matowego – 2 zł
- Ustala nazwę pliku z wartości pobranej z pierwszego pola edycyjnego
- Tworzy elementy i dodaje je do bloku z sekcji prawej (ilustracja 4):
 - Element DOM dla obrazu z ustaloną nazwą pliku
 - Element DOM dla paragrafu z treścią „Liczba kopii: <kopie>”, gdzie pole <> jest pobrane z kontrolki
 - Element DOM dla paragrafu z treścią „Cena: <cena>”, gdzie pole <> jest wyliczoną ceną

Zamówienie

Wybierz obraz: 7.jpg

Liczba Kopii:

☒ Papier błyszczący

☐ Papier matowy

[Regulamin](#) [Usługi](#)

Koszyk

Liczba kopii: 10

Cena: 20

Liczba kopii: 7

Cena: 14

Liczba kopii: 20

Cena: 30

Ilustracja 4. Działanie skryptu – trzy razy wypełniono i zatwierdzono formularz

File Input Type

`<input>` elements with `type="file"` let the user choose one or more files from their device storage. Once chosen, the files can be uploaded to a server using form submission, or manipulated using JavaScript code and the File API. Example:

```
<input type="file" id="plik" accept="image/png, image/jpeg" />
```

Tabela 1. Wybrane pola i metody modelu DOM języka JavaScript

Wyszukiwanie elementów	Zmiana elementów
<code>document.getElementById(<i>id</i>)</code>	<code>element.innerHTML = "wartość"</code>
<code>document.getElementsByTagName(<i>TagName</i>)</code>	<code>element.attribute_name = "wartość"</code>
<code>document.getElementsByClassName(<i>ClassName</i>)</code>	<code>element.setAttribute(<i>atrybut</i>, <i>wartosc</i>)</code>
<code>document.getElementsByName(<i>ElementName</i>)</code>	<code>element.style.property_name = "wartość"</code>
<code>document.querySelector(<i>CSSselector</i>)</code>	
<code>document.querySelectorAll(<i>CSSselector</i>)</code>	

Operacje na elementach dokumentu	Wybrane właściwości obiektu style
<code>document.createElement(<i>element</i>)</code>	<code>backgroundColor</code>
<code>document.removeChild(<i>element</i>)</code>	<code>color</code>
<code>document.appendChild(<i>element</i>)</code>	<code>fontSize</code>
<code>document.replaceChild(<i>element</i>)</code>	<code>fontStyle = "normal italic oblique initial inherit"</code>
<code>document.write(<i>text</i>)</code>	<code>fontWeight = "normal lighter bold bolder value initial inherit"</code>
	<code>listStyleType = "circle decimal disc none square initial..."</code>

Wybrane zdarzenia HTML

Zdarzenia myszy	Zdarzenia klawiatury	Zdarzenia obiektów
<code>onclick</code>	<code>onkeydown</code>	<code>onload</code>
<code>ondblclick</code>	<code>onkeypress</code>	<code>onresize</code>
<code>onmouseover</code>	<code>onkeyup</code>	<code>onfocusin</code>
<code>onmouseout</code>		<code>onfocusout</code>

Elementy formularzy	Metody i pola obiektu string (JS)
Ważniejsze typy pola input: button, checkbox, number, password, radio, text, range, file	<code>Length</code>
Inne elementy: select, textarea	<code>indexOf(<i>text</i>)</code>
	<code>search(<i>text</i>)</code>
	<code>lastIndexOf()</code>
	<code>substr(<i>startIndex</i>, <i>endIndex</i>)</code>
	<code>replace(<i>textToReplace</i>, <i>newText</i>)</code>
	<code>toUpperCase()</code>
	<code>toLowerCase()</code>

Tabela 2. Tworzenie elementów w JavaScript**Example**

Create a <p> element and append it to the document:

```
const para = document.createElement("p");  
para.innerText = "This is a paragraph";  
para.className = "nazwaKlasyCSS";  
document.body.appendChild(para);
```

Example

Append an item to a list:

```
const node = document.createElement("li");  
/* add text, classes and attributes */  
document.getElementById("idListy").appendChild(node);
```

Tabela 3. Semantic Elements in HTML

Tag	Description
<article>	Defines independent, self-contained content
<aside>	Defines content aside from the page content
<details>	Defines additional details that the user can view or hide
<figcaption>	Defines a caption for a <figure> element
<figure>	Specifies self-contained content, like illustrations, diagrams, photos, code listings, etc.
<footer>	Defines a footer for a document or section
<header>	Specifies a header for a document or section
<main>	Specifies the main content of a document
<mark>	Defines marked/highlighted text
<nav>	Defines navigation links
<section>	Defines a section in a document
<summary>	Defines a visible heading for a <details> element
<time>	Defines a date/time

UWAGA: po zakończeniu pracy utwórz plik tekstowy o nazwie przeglądarka.txt. Zapisz w nim nazwę przeglądarki internetowej, w której weryfikowana była poprawność działania witryny. Umieść go w folderze z numerem zdającego.

Nagraj płytę z rezultatami pracy. W folderze z numerem zdającego, powinny znajdować się pliki: 10 obrazów o nazwach 1.jpg – 10.jpg, import.png, kw1.png, kw2.png, kw3.png, kw4.png, kwerendy.txt, przeglądarka.txt, regulamin.html, style.css, usługi.html, zamowienie.html, ewentualnie inne przygotowane pliki. Po nagraniu płyty sprawdź poprawność jej odczytu. Opisz płytę numerem zdającego i pozostaw zapakowaną w pudełku na stanowisku wraz z arkuszem egzaminacyjnym.

Czas przeznaczony na wykonanie zadania wynosi 150 minut.

Ocenie podlegać będzie 5 rezultatów:

- operacje na bazie danych,
- zawartość witryny internetowej,
- działanie witryny internetowej,
- styl CSS witryny internetowej,
- skrypt.