## Zadanie egzaminacyjne

*UWAGA: katalog z rezultatami pracy oraz płytę należy opisać numerem, którym został podpisany arkusz, czyli numerem PESEL lub w przypadku jego braku numerem paszportu.*

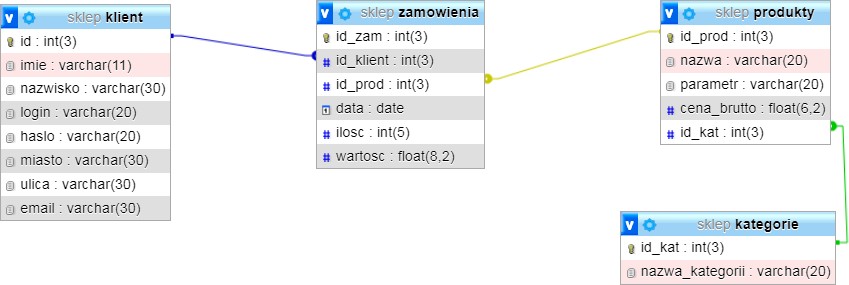
Wykonaj aplikację internetową pomocnika CSS, wykorzystując pakiet XAMPP oraz edytor zaznaczający składnię.

Aby wykonać zadanie, należy zalogować się na konto **Egzamin** bez hasła. Na pulpicie znajduje się archiwum o nazwie *zad2.zip* zabezpieczone hasłem: **Css\_$\_Css**

Archiwum należy rozpakować.

Na pulpicie konta **Egzamin** należy utworzyć folder. Jako nazwy folderu należy użyć numeru zdającego, którym został podpisany arkusz. Rozpakowane pliki należy umieścić w tym folderze. Po skończonej pracy wszystkie wyniki należy zapisać w tym folderze.

## Operacje na bazie danych

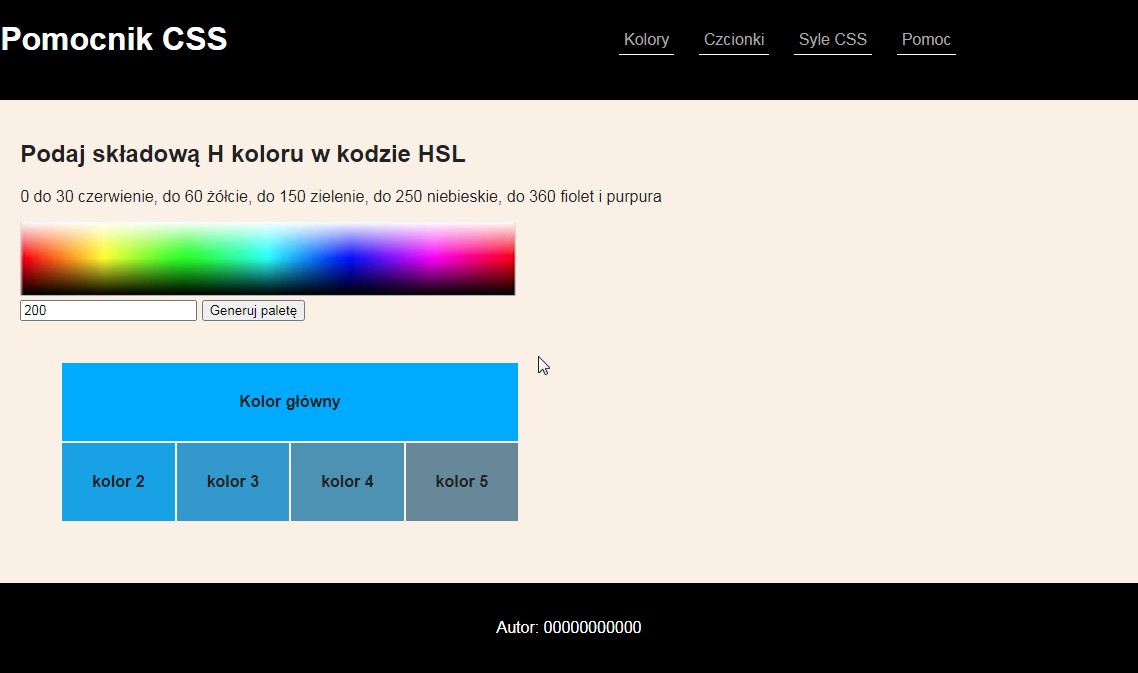
Do wykonania operacji na bazie należy wykorzystać tabele przedstawione na Obrazie 1.

## Obraz 1. Tabele bazy danych

Za pomocą narzędzia phpMyAdmin należy wykonać operacje na bazie danych:

* Utworzenie bazy danych o nazwie: *sklep* z zestawem polskich znaków (np. *utf8\_unicode\_ci*)
* Import do bazy *sklep* z pliku *sklep.sql*. Czynność należy udokumentować zrzutem ekranu, na którym powinny być widoczne elementy wskazujące na poprawnie wykonany import tabel. Zrzut należy zapisać w formacie JPEG, pod nazwą *import*. Powinien on obejmować cały ekran monitora, z widocznym paskiem zadań.
* Zapytania SQL na bazie *sklep* zapisane do pliku *kwerendy.txt*. Wykonanie kwerend należy udokumentować zrzutami w formacie PNG o nazwach *kw1*, *kw2*, *kw3*, *kw4*. Zrzuty powinny obejmować cały ekran monitora z widocznym paskiem zadań
  + Zapytanie 1: wybierające jedynie imię, nazwisko i miasto klientów z miast: Legionowo, Warszawa
  + Zapytanie 2: wybierające jedynie nazwę produktu dla produktów o cenie brutto wyższej od 500 zł
  + Zapytanie 3: wybierające nazwę kategorii z tabeli *kategorie* i odpowiadającą jej wartość średniej arytmetycznej cen brutto produktów tej kategorii z tabeli *produkty*. Należy posłużyć się relacją.
  + Zapytanie 4: wybierające id zamówienia, ilość i wartość z tabeli *zamowienia* oraz odpowiadającą im nazwę produktu z tabeli *produkty* dla klienta o id równym 2. Należy posłużyć się relacją

## Witryna internetowa

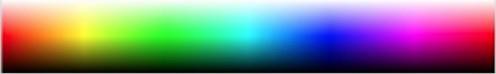


**Obraz 2. Witryna internetowa.**

**Zatwierdzono wartość 200, tabela wypełniona kolorami niebieskimi**

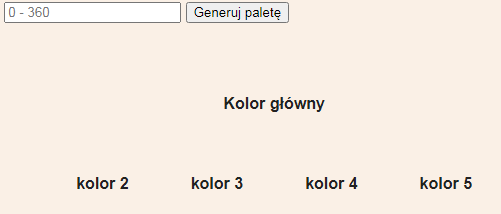
Cechy grafiki z pliku *kolory.jpg*:

* Wykadrowana tak, aby była widoczna jedynie przestrzeń barw zgodnie z obrazem 3
* Przeskalowana **bez zachowania** proporcji do szerokości 500 px, wysokość pozostaje bez zmian i wynosi około 75 px (przybliżony wymiar, zależny od kadrowania)



**Obraz 3. Grafika *kolory.jpg***

Cechy witryny:

* Składa się ze stron o nazwach *kolory.html*, *czcionki.html*, *style.html*
* Pliki *czcionki.html*, *style.html* zawierają jedynie tekst „strona w trakcie budowy”, pozostałe cechy dotyczą tylko pliku *kolory.html*
* Zapisana w języku HTML5
* Jawnie zastosowano właściwy standard kodowania polskich znaków
* Zadeklarowany język zawartości witryny: polski
* Tytuł strony widoczny na karcie przeglądarki: „Pomocnik CSS”
* Arkusz stylów w pliku o nazwie *styl2.css* prawidłowo połączony z kodem strony
* Podział strony na: kontener całej strony, a w nim obok siebie dwa bloki banera, blok główny, stopka. Podział zrealizowany za pomocą znaczników sekcji tak, aby wygląd układu bloków strony po uruchomieniu w przeglądarce, był zgodny z Obrazem 2
* Zawartość pierwszego bloku banera: nagłówek pierwszego stopnia o treści „Pomocnik CSS”
* Zawartość drugiego bloku banera: lista punktowana (nieuporządkowana) z elementami, które jednocześnie są odnośnikami:
  + element Kolory prowadzący do strony *kolory.html*
  + element Czcionki prowadzący do strony *czcionki.html*
  + element Style CSS prowadzący do strony *style.html*
  + element Pomoc prowadzący do strony <https://www.w3schools.com/css/>, odnośnik powinien otwierać się w nowej karcie przeglądarki
* Zawartość bloku głównego:
  + nagłówek drugiego stopnia o treści „Podaj składową H koloru w kodzie HSL”
  + paragraf treści: „0 do 30 czerwienie, do 60 żółcie, do 150 zielenie, do 250 niebieskie, do 360 fiolet i purpura”
  + obraz *kolory.jpg* z tekstem alternatywnym „Przestrzeń barw HSL”
  + Pole edycyjne typu numerycznego o maksymalnej wartości 360 i podpowiedzi o treści „0 – 360”
  + Przycisk „Generuj paletę”, którego wciśnięcie uruchamia skrypt
  + Tabela o czterech kolumnach i dwóch wierszach, z czego pierwszy wiersz ma wszystkie komórki połączone, wypełniona zgodnie z Obrazem 2
* Zawartość bloku stopki: paragraf o treści „Autor: ”, dalej wstawiony numer zdającego

## Obraz 4. Fragment bloku głównego - stan początkowy aplikacji Styl CSS witryny internetowej

Cechy formatowania CSS:

* Styl CSS zdefiniowany w całości w zewnętrznym pliku o nazwie *styl2.css*
* Dla kontenera całej strony: krój czcionki Helvetica, w przypadku braku sans-serif, kolor tła Linen, kolor czcionki #222, marginesy zewnętrzne automatycznie wyliczane przez przeglądarkę, szerokość 90%
* Wspólne dla obu bloków banera: czarny kolor tła, biały kolor czcionki, szerokość 50%, wysokość 100 px
* Dla bloku głównego: marginesy wewnętrzne 20 px
* Dla stopki: czarny kolor tła, biały kolor czcionki, marginesy wewnętrzne 20 px, wyrównanie tekstu do środka
* Dla selektora tabeli: marginesy zewnętrzne 40 px
* Dla selektora komórki tabeli: marginesy wewnętrzne 30 px, wyrównanie tekstu do środka, czcionka pogrubiona
* Dla odnośnika: kolor czcionki #aaa bez podkreślenia
* W momencie, gdy kursor znajdzie się na odnośniku kolor czcionki tego odnośnika zmienia się na biały
* Dla selektora elementu listy: sposobem wyświetlenia jest kontener liniowo-blokowy (patrz Tabela 3), marginesy wewnętrzne 5 px, zewnętrzne 10 px, dolna linia obramowania ciągła o szerokości 1 px i kolorze białym

*Uwaga: style CSS dla tabeli, komórki tabeli i elementu listy należy zdefiniować wyłącznie przy pomocy selektora dla znaczników tabela, komórka tabeli i element listy. Jest to uwarunkowane projektem późniejszej rozbudowy witryny.*

## Skrypt

W Tabeli 1 podano Wybrane pola i metody modelu DOM języka JavaScript. Wymagania dotyczące skryptu:

* Napisany w języku JavaScript
* Składa się z funkcji wywoływanej po wciśnięciu przycisku
* Pobiera wartość wpisaną w polu edycyjnym
* Ustala tło w formacie HSL dla komórek tabeli. Składowa H odpowiada wartości pobranej z pola edycyjnego, wartość L jest stała i wynosi zawsze 50%, wartość S jest różna dla poszczególnych komórek tabeli i wynosi:
  + dla pierwszego wiersza 100%
  + dla drugiego wiersza, kolejno w czterech komórkach: 80%, 60%, 40% i 20%

Przykład:

W polu edycyjnym wpisano 200, tła HSL komórek tabeli rozkładają się następująco:

|  |  |  |  |
| --- | --- | --- | --- |
| 200, 100%, 50% | | | |
| 200, 80%, 50% | 200, 60%, 50% | 200, 40%, 50% | 200, 20%, 50% |

## Tabela 1. Wybrane pola i metody modelu DOM języka JavaScript

|  |  |
| --- | --- |
| **Wyszukiwanie elementów** | **Zmiana elementów** element.innerHTML = "*nowa zawartość*" element.*attribute\_name* = "*nowa wartość*"  element.setAttribute(*atrybut*, *wartosc*) element.style.*property\_name* = "nowa wartość" |
| document.getElementById(*id*) document.getElementsByTagName(*TagName*)  document.getElementsByClassName(*ClassName*) document.getElementsByName(*ElementName*) |

|  |  |
| --- | --- |
| **Operacje na elementach dokumentu** | **Wybrane właściwości style (property\_name)** |
| document.createElement(*element*) | backgroundColor |
| document.removeChild(*element*) | color |
| document.appendChild(*element*) | fontSize |
| document.replaceChild(*element*) | fontStyle = "normal | italic | oblique | initial | inherit" |
| document.write(*text*) | fontWeight = "normal | lighter | bold | bolder | value | |
|  | initial | inherit" |
|  | listStyleType = "circle| decimal | disc | none | square | |
|  | initial..." |

Wybrane zdarzenia HTML

|  |  |  |
| --- | --- | --- |
| **Zdarzenia myszy** | **Zdarzenia klawiatury** | **Zdarzenia obiektów** |
| onclick ondblclick  onmouseover onmouseout | onkeydown onkeypress onkeyup | onload onresize  onfocusin onfocusout |

## Tabela 2. Sposoby reprezentacji koloru w stylach CSS

|  |  |  |
| --- | --- | --- |
| **Sposób** | **Składnia** | **Przykład JS** |
| Nazwa | Nazwa\_koloru | *element*.style.backgroundColor = 'SteelBlue'; |
| Heksadecymalnie | #rrggbb (wartości hex 00-FF) | *element*.style.color = '#FFFFFF'; |
| RGB | rgb(wartR, wartG, wartB) | *element*.style.backgroundColor = 'rgb(10, 20, 30)'; |
| HSL | hsl(wartH, wartS%, wartL%) | *element*.style.color = 'hsl(30, 100%, 50%)'; |

**Tabela 3. Wybrane fragmenty pomocy ze strony *kodilla.com* na temat właściwości CSS display Opis:** CSS display definiuje sposób wyświetlania elementu.

**Składnia:** display: value;

## Spis wartości:

|  |  |
| --- | --- |
| **Wartość (value)** | **Opis** |
| inline | Wyświetla element w linii (np. <span>) |
| block | Wyświetla element w bloku (np. <p>) |
| inline-block | Wyświetla element jako kontener liniowo-blokowy |