

Ćwiczenie 4

Wprowadzenie do kaskadowych arkuszy stylów. Organizacja podstawowych informacji na stronie internetowej przy pomocy CSS – nagłówki, listy, tekst.

1. WSTĘP

Kaskadowe arkusze stylu (ang. *Cascading Style Sheets*) **CSS** – język służący do opisu formy prezentacji (wyświetlania) zawartości stron internetowych w przeglądarce. Przy pomocy CSS możemy kontrolować wygląd i sposób działania stron internetowych. Jeżeli wygląd niektórych elementów już został zdefiniowany w HTML to przy użyciu CSS można przeglądarce nakazać jego wyświetlenie w inny niż standardowy sposób.

Style CSS oferują elastyczność w zakresie zmiany dowolnego typu elementu, a same style mogą być stosowane na różne sposoby. Na przykład wszystkie style mogą być zdefiniowane w oddzielnym pliku, do którego odwołanie znajdzie się na stronie internetowej. A zatem jeśli będzie trzeba zmienić wygląd całej witryny, wystarczy po prostu przeprowadzić edycję pliku CSS, a wprowadzone zmiany zostaną odzwierciedlone na wszystkich stronach zawierających odwołanie do danego arkusza stylów. Jeżeli preferowane jest inne podejście, style możemy umieścić na początku strony w HTML i wówczas mają zastosowanie jedynie na tej stronie. Arkusze stylów wpływają na wygląd całej strony. Jednak za pomocą atrybutu **<style>** można je zastosować tylko dla wskazanego znacznika. Istnieje również możliwość umieszczenia stylów wewnątrz samego znacznika za pomocą atrybutu **<style>**. Poniżej przedstawiono przykłady wykorzystania każdego ze sposobów:

- **wewnątrz znacznika** (tzw. *inline*). W hierarchii sposobów definiowania stylów CSS ten sposób ma najwyższy priorytet. Składnia definicji wygląda następująco:

```
<nazwa_znacznika style="właściwość1:wartość1; właściwość2:wartość2;">
```

Przykład wykorzystania:

```
<body style="background-color:grey;">
  <h2 style="font-family:Verdana; font-size:20px; color:white;">
    Style CSS </h2>
</body>
```

Powyższy kod ustawia kolor tła sekcji **body** na szary oraz definiuje krój czcionki, jej rozmiar i kolor nagłówka **h2**.

- **za pomocą znacznika <style>**. Definicja stylów zazwyczaj jest umieszczana wewnątrz sekcji **head**. Jest to drugi za priorytetem sposób definiowania stylu CSS. Składnia takiej definicji wygląda następująco:

```
<style>
  nazwa_elementu
  {
    właściwość1: wartość1;
    właściwość2: wartość2;
  }
</style>
```

Przykład wykorzystania:

```
<head>
  <style>
    h1{
      text-align: center;
      font-size:40px;
    }
  </style>
</head>
```

```

        background-color: green;
    }
    p{
        margin-left: 20px;
        color: white;
    }
</style>
</head>
<body>
    <h1> Style CSS </h1>
    <p> Zawartość akapitu #1 </p>
    <p> Zawartość akapitu #2 </p>
</body>

```

Powyższy kod pokazuje, że nagłówek **h1** oraz akapity **p** zostały utworzone w sekcji **body** natomiast ich style są definiowane w sekcji **head** za pomocą znacznika **style**. Wyrównanie tekstu **h1** ustawiono do środka, rozmiar czcionki to 40 px oraz kolor tła ustawiono na zielony. Określono parametry akapitów: lewy margines na 20 px oraz kolor tekstu – biały.

- za pomocą zewnętrznego pliku **.css**. Definicja stylów CSS zostaje zawarta w dodatkowym pliku z rozszerzeniem **.css**. Posiada trzeci po priorytetowości poziom. Sposób opisu stylów elementów jest taki sam jak i w poprzednim przypadku. Żeby móc korzystać z tej opcji należy w pliku **.html** wskazać na zewnętrzny arkusz CSS:

```
<link rel="stylesheet" href="nazwa.css" type="text/css"/>
```

Atrybut **type** wskazuje na tekstowy typ pliku **.css** oraz nie jest wymagany w standardzie HTML5. Przykład wykorzystania:

```

1 <!DOCTYPE HTML>
2 <html lang="pl">
3 <head>
4   <link rel="stylesheet" href="style.css"/>
5 </head>
6 <body>
7   <h1> Wprowadzenie do CSS </h1>
8   
9   <p> CSS to język służący do opisu formy
10  prezentacji (wyświetlania) zawartości
11  stron internetowych w przeglądarce. </p>
12  <p> Przy pomocy CSS możemy kontrolować
13  wygląd i sposób działania stron
14  internetowych. </p>
15 </body>
16 </html>

```

```

1 body {
2   background-color: grey;
3   font-family: Verdana;
4   font-size: 20px;
5 }
6 h1 {
7   text-align: center;
8   font-size: 40px;
9 }
10 img {
11  width: 250px;
12  height: 270px;
13 }

```

Powyższy przykład pokazuje, że do pliku **start.html** dołączono arkusz styli **style.css** zawierający definicji właściwości sekcji **body**, nagłówka **h1** oraz obrazu **img**.

Priorytetowość sposobu definicji styli można zmieniać przy pomocy deklaracji **!important**. Oznacza to, że np. właściwość elementu, zdefiniowana w zewnętrznym pliku **.css** oraz posiadająca deklarację **!important** uzyskuje wyższy priorytet niż definicja stylu w sposób *inline*:

```

1 <!DOCTYPE HTML>
2 <html lang="pl">
3 <head>
4   <link rel="stylesheet" href="style.css"/>
5 </head>
6 <body>
7   <h2 style="color:white"/> Priorytety w CSS
8 </body>
9 </html>

```

```

1 h2{
2   color: red !important;
3 }

```

W tym przypadku kolor tekstu nagłówku **h2** będzie czerwony.

2. WYKONANIE ĆWICZENIA

Zadanie 1 – Stylizacja zawartości strony przy użyciu CSS

Zapoznaj się z informacjami dotyczącymi możliwości formatowania tekstu, nagłówków i list przy użyciu stylu CSS. Wybrane właściwości zostaną wykorzystane podczas wykonania ćwiczenia.


Tabela 1. Właściwości zbioru „CSS Text”

<i>Nazwa podzbioru</i>	<i>Nazwa właściwości, składnia</i>	<i>Opis właściwości</i>
Text Color	color: wartość;	służy do ustawienia koloru tekstu. Kolor określa się przy użyciu: <ul style="list-style-type: none"> • nazwy koloru – np. "red"; • odpowiedniej wartości HEX – np. "#ff0000"; • odpowiednich wartości RGB – np. "rgb(255,0,0)".
	background-color: wartość;	służy do ustawienia koloru tła tekstu. Definiuje się w sposób podobny do color.
Text Alignment	text-align: wartość;	służy do ustawiania wyrównania tekstu w poziomie: <ul style="list-style-type: none"> • right – do prawej; • centered – wyśrodkowany; • left – do lewej; • justify – wyjustowany.
	text-direction: wartość;	służy do zmiany kierunku tekstu elementu: <ul style="list-style-type: none"> • rtl – dla hebrajskiego lub arabskiego tekstu; • ltr – dla innych pism.
	vertical-align: wartość;	ustawia pionowe wyrównanie elementu: <ul style="list-style-type: none"> • top – do góry; • middle – do środka; • bottom – do dołu.
Text Decoration	text-decoration: wartość;	służy do ustawiania lub usuwania dekoracji z tekstu: <ul style="list-style-type: none"> • none – jest często używany do usuwania podkreśleń z linków; • overline – linia nad tekstem; • line-through – zakreślenie tekstu; • underline – podkreślenie tekstu.
Text Transformations	text-transform: wartość;	służy do określania wielkich i małych liter w tekście: <ul style="list-style-type: none"> • uppercase – wszystkie litery są przekształcane na wielkie; • lowercase – wszystkie litery są przekształcane na małe; • capitalize – pierwsza litera każdego słowa jest zmieniana na wielką, inne zostają bez zmian; • none – żadne zmiany wielkości liter nie są dokonywane.
Text Spacing	text-indent: wartość px;	służy do określenia wcięcia pierwszej linii tekstu.
	letter-spacing: wartość px;	służy do określenia odstępu między znakami w tekście.
	line-height: wartość;	służy do określenia odstępu między wierszami.
	word-spacing: wartość px;	służy do określenia odstępu między słowami w tekście.

Kontynuacja Tabeli 1.

<i>Nazwa podzbioru</i>	<i>Nazwa właściwości, składnia</i>	<i>Opis właściwości</i>
Text Spacing	white-space: wartość px;	określa sposób obsługi białych znaków wewnątrz elementu: <ul style="list-style-type: none"> • normal – wszystkie sekwencje białych znaków i zakończenia linii będą skracane. Zakończenia linii nie likwidują zawijania zawartości; • pre – wszystkie białe znaki w źródle będą zachowane, linie są przełamane tylko na znakach nowej linii w źródle; • nowrap – wszystkie sekwencje białych znaków i zakończenia linii będą skracane, również przełamania linii (zawijanie tekstu) są zlikwidowane wewnątrz zawartości; • pre-wrap – wszystkie białe znaki w źródle będą zachowane, linie są przełamane na znakach nowej linii i w razie konieczności zawijają zawartość; • pre-line – wszystkie sekwencje białych znaków będą skrócone, linie są przełamane na znakach nowej linii i również w razie potrzeby zawartość jest zawijana; • inherit – dziedziczy po rodzicu.
Text Shadow	text-shadow: wartość1px wartość2px;	dodaje cień do tekstu: Text shadow effect!
	text-shadow: wartość1px wartość2px kolor;	dodaje kolor do cieniu: Text shadow effect!
	text-shadow: wartość1px wartość2px wartość 3px kolor;	dodaje blur-efekt (wartość3px) do cieniu: Text shadow effect!

Tabela 2. Właściwości zbioru „CSS Fonts”

<i>Nazwa podzbioru</i>	<i>Nazwa właściwości, składnia</i>	<i>Opis właściwości</i>
Font Family	font-family: krój;	określona rodzinę czcionek, np. Times New Roman
	generic-family: typ;	grupa rodzin czcionek o podobnym wyglądzie – serif, sans-serif, monospace: 
Font Style	font-style: styl;	umożliwia określenie stylów dla określonych czcionek: <ul style="list-style-type: none"> • normal – wybiera normalną wersję rodziny dla wybranych czcionek; • italic – określa, że font-face jest zapisaną kursywą; • oblique – krój czcionki jest sztucznie nachyloną. Może być również określony kąt nachylenia.

Kontynuacja Tabeli 2.

<i>Nazwa podzbioru</i>	<i>Nazwa właściwości, składnia</i>	<i>Opis właściwości</i>
Font Style	font-weight: styl lub wartość;	określa wagę lub grubość czcionki: <ul style="list-style-type: none"> • normal – normalna waga czcionki. Odpowiada wartości 400; • bold – pogrubiona czcionka. Odpowiada wartości 700; • lighter – waga czcionki lżejsza o jedną wartość niż w elemencie rodzica; • bolder – waga czcionki cięższa o jedną wartość niż w elemencie rodzica; • 100, 200, 300, 400, 500, 600, 700, 800, 900 – liczbowe wartość wagi czcionek dla czcionek, które mają więcej niż tylko waga normalna i pogrubiona;
	font-variant: styl;	wybiera z rodziny czcionek wygląd normalny lub wielkie litery: <ul style="list-style-type: none"> • normal – definiuje normalny krój czcionki; • small-caps – definiuje czcionkę, która jest wyróżniona dużymi literami.
Font Size	font-size: rozmiar;	określa rozmiar czcionki.

Tabela 3. Właściwości zbioru „CSS Lists”

<i>Nazwa podzbioru</i>	<i>Nazwa właściwości, składnia</i>	<i>Opis właściwości</i>
List Style	list-style-type: typ;	określa typ znacznika elementu listy – circle, square, upper-roman, lower-alpha i td. Wartość none jest wybierana w przypadku gdy nie chcemy korzystać z markerów.
	list-style-image: obraz;	określa obraz jako znacznik elementu listy – np.: url('picture.jpg').
	list-style-position: pozycja;	określa położenie znaczników pozycji listy: <ul style="list-style-type: none"> • outside – markery znajdują się poza elementem listy; • inside – markery znajdują się wewnątrz elementu listy.
	list-style: właściwości	służy do ustawienia wszystkich właściwości listy w jednej deklaracji.

Dodatkowe opis właściwości innych zbiorów styli CSS można znaleźć na stronach:

<https://www.w3schools.com/css/> (w języku angielskim)

<https://developer.mozilla.org/pl/docs/Web/CSS> (w języku polskim)

Zadanie 2 – Stylizacja zawartości strony przy użyciu CSS

Utwórz dwa nowe pliki i nazwij je jako **priorytety.html** oraz **style.css**. W pliku **.html** wykonaj jego następujące modyfikacje:

- 1) zdefiniuj tytuł strony jako „Wprowadzenie do CSS”;
- 2) wprowadź opis zawartości strony jako „Strona zawierająca różne sposoby definicji stylów CSS. Tu znajdziesz trzy sposoby na tworzenie styli oraz deklarację „łamania” priorytetów!”;
- 3) dołącz plik **style.css** do **priorytety.html**;
- 4) zdefiniuj styl sekcji body w następujący sposób: sposób tworzenia stylu CSS – **inline**, krój czcionki – Arial, rozmiar czcionki – 20px, kolor – niebieski;
- 5) wstaw logo CSS do utworzonej strony internetowej (obraz z logiem udostępnia prowadzący).
Wymagania do obrazu: źródło – dysk twardy komputera, sposób tworzenia stylu CSS – **inline**, szerokość – 455px, wysokość – 850px; margines lewy – 50px;
- 6) utwórz nagłówek #1 o treści „Krótka historia CSS”. Określ jego **właściwości następująco:** sposób tworzenia stylu CSS – za pomocą znacznika `<style>`, wyrównanie – do środka strony, krój czcionki – Times New Roman, rozmiar czcionki 40px, tło – szare, kolor – czerwony;
- 7) utwórz akapit z tekstem zawierającym krótką historię CSS. Tekst skopiuj z:

https://pl.wikipedia.org/wiki/Kaskadowe_arkusze_styl%C3%B3w (rozdział **Historia**)

Właściwości akapitu: sposób tworzenia stylu CSS – za pomocą zewnętrznego pliku, wyrównanie – wyjustowany, margines lewy – 20 px, margines prawy – 20 px, wcięcie – 30 px;

- 8) utwórz nagłówek #2 o treści „Postawione cele”. Określ jego **właściwości następująco:** sposób tworzenia stylu CSS – za pomocą znacznika `<style>`, wyrównanie – do środka strony, krój czcionki – Verdana, rozmiar czcionki 55px, tło – szare, kolor – żółty;
- 9) utwórz akapit #2 z tekstem opisującym cele tworzenia CSS. Tekst skopiuj z:

https://pl.wikipedia.org/wiki/Kaskadowe_arkusze_styl%C3%B3w (rozdział **Cele**)

Zmień kolor tekstu akapitu na zielony. Zastanów się jak to zrobić, żeby kolor tekstu w akapicie #1 pozostał bez zmian.

- 10) opracuj wnioski na temat stylizacji zawartości stron internetowych przy użyciu CSS. Jakie wady/zalety ma stylizacja CSS w porównaniu do HTML?

Po wykonaniu wszystkich wytycznych uzyskasz następujący widok strony:



Krótką historia CSS

Pierwotnie HTML był językiem wyłącznie do opisu struktury dokumentu. Jednak z czasem zrodziła się potrzeba ożywienia wyglądu takich dokumentów. Powoli dodawano nowe znaczniki do HTML, pozwalające kontrolować kolory, typografię, dodawać nowe media (np. obrazki). Te niestandardowe rozszerzenia realizowane były przez najpopularniejszych producentów przeglądarek bez porozumienia z drugim. Doprowadziło to do zaimplementowania nowych znaczników działających w konkretnej grupie przeglądarek i nie działających w innych przeglądarkach. Projektanci zostali zmuszeni do wysyłania do klienta różnych wersji tej samej witryny w zależności od użytej przeglądarki, uzyskanie identycznego wyglądu w różnych przeglądarkach było praktycznie niemożliwe. Håkon Wium Lie jako pierwszy zaproponował CHSS (Cascading HTML Style Sheets) w październiku 1994 roku. Później Lie i Bert Bos pracowali wspólnie nad standardem CSS (literka H została usunięta ze względu na możliwość stosowania stylów do innych podobnych do HTML języków). W tym czasie została utworzona organizacja World Wide Web Consortium, która z Lie'em i Bossem na czele przejęła prace nad CSS. Pod koniec 1996 roku wydano oficjalną dokumentację CSS, Kaskadowe arkusze stylów, poziom 1. W3C zatwierdziło dwa oficjalne standardy CSS: CSS 1 i CSS 2, a także dnia 7 czerwca 2011 roku standard CSS 2.1. Trwają również prace nad CSS3. Wersja ta w stosunku do poprzedników wzbogaci się o wiele selektorów oraz właściwości, nowością jest także modułowy charakter języka – nie będzie to już jednolita rekomendacja, lecz kilkadziesiąt osobnych dokumentów, co pozwoli na włączanie lub wyłączenie odpowiednich modułów w przeglądarkach w zależności od chwilowych potrzeb.

Postawione cele

Przed pojawieniem się CSS wszystkie informacje dotyczące wyglądu dokumentów HTML (między innymi rodzaj i kolor czcionki, ułożenie, marginesy) zawarte były w znacznikach HTML. Język CSS umożliwił przeniesienie tych informacji do osobnego pliku. Skutkuje to uproszczeniem i zwiększeniem przejrzystości samego dokumentu HTML. Bez użycia CSS w przypadku definiowania stylu dla nagłówka (h1) lub podtytułu (h2) jego definicja musiałaby zostać powtórzona w każdym miejscu, w którym pojawia się dana struktura. Efektem takiego działania byłoby zmniejszenie czytelności dokumentu i jego odporności na błędy oraz trudność w utrzymaniu. Dodatkowo, zmiany stylu w jednym z miejsc wiązałyby się z koniecznością wprowadzania zmian w każdym miejscu wystąpienia. CSS pozwala na rozgraniczenie warstwy prezentacji od struktury. Język ten umożliwia definicję kolorów, czcionek, układu, rozmiarów, marginesów oraz wielu innych cech związanych z warstwą prezentacji.

Zadanie 3 – Priorytety stylów CSS

Wykonaj następujące modyfikację strony internetowej:

- 1) wyśrodkuj logo CSS. Jest to możliwe poprzez użycie następujących właściwości:

```
margin-left: auto;
margin-right: auto;
display: block;
```

deklaracja **display: block** – przekształca obraz na element blokowy strony. Zmień rozmiar logotypu na 680 (długość)×364 (szerokość). Dokonaj niezbędnych zmian z poziomu pliku *style.css*.

- 2) zastosuj jednolitą stylizację nagłówków na stronie internetowej: krój czcionki – Times New Roman, rozmiar – 30px; styl – pochylony, kolor – pomarańczowy, tło – szare, wyrównanie – do środka. Dokonaj niezbędnych zmian z poziomu pliku *style.css*.
- 3) zmień rozmiar czcionki tekstu w akapitach na 18px. Zmień marginesy akapitów lewy i prawy na 40 px; ustaw wcięcie na 25px. Zmian dokonaj modyfikując kod znacznika `<style>` pliku *.html*;
- 4) zmień kolor akapitu #1 na brązowy. Zmiany dokonaj wewnątrz znacznika akapitu;
- 5) dodaj kolejny nagłówek (#3) o nazwie „CSS Frameworks” – stylizacja nagłówka jak i w poprzednich przypadkach. Umieść definicję nagłówka w pliku *.css*;
- 6) utwórz akapit #3 z tekstem opisującym biblioteki CSS. Tekst skopiuj z: https://pl.wikipedia.org/wiki/Kaskadowe_arkusze_styl%C3%B3w (rozdział **CSS Frameworks**)
Zastosuj dowolny kolor tekstu – zdefiniuj go wewnątrz znacznika.
- 7) opracuj wnioski na temat hierarchii sposobów stylizacji przy użyciu CSS. W jaki sposób można zmieniać priorytety w tych sposobach?

Po wykonaniu wszystkich wytycznych uzyskasz następujący widok strony:



Krótką historią CSS

Pierwotnie HTML był językiem wyłącznie do opisu struktury dokumentu. Jednak z czasem zrodziła się potrzeba ożywienia wyglądu takich dokumentów. Powoli dodawano nowe znaczniki do HTML pozwalające kontrolować kolory, typografię, dodawać nowe media (np. obrazy). Te niestandardowe rozszerzenia realizowane były przez najpopularniejszych producentów przeglądarek bez porozumienia z drugim. Doprowadziło to do zaimplementowania nowych znaczników działających w konkretnej grupie przeglądarek i nie działających w innych przeglądarkach. Projektanci zostali zmuszeni do wysyłania do klienta różnych wersji tej samej witryny w zależności od użytej przeglądarki, uzyskanie identycznego wyglądu w różnych przeglądarkach było praktycznie niemożliwe. Hakon Wium Lie jako pierwszy zaproponował CHSS (Cascading HTML Style Sheets) w październiku 1994 roku. Później Lie i Bert Bos pracowali wspólnie nad standardem CSS (literka H została usunięta ze względu na możliwość stosowania stylów do innych podobnych do HTML języków). W tym czasie została utworzona organizacja World Wide Web Consortium, która z Lie'em i Bossem na czele przejęła prace nad CSS. Pod koniec 1996 roku wydano oficjalną dokumentację CSS, Kaskadowe arkusze stylów, poziom 1. W3C zatwierdziło dwa oficjalne standardy CSS: CSS 1 i CSS 2, a także dnia 7 czerwca 2011 roku standard CSS 2.1. Trwają również prace nad CSS3. Wersja ta w stosunku do poprzedników wzbogaciła się o wiele selektorów oraz właściwości, nowością jest także modułowy charakter języka – nie będzie to już jednolita rekomendacja, lecz kilkadziesiąt osobnych dokumentów, co pozwoli na włączanie lub wyłączanie odpowiednich modułów w przeglądarkach w zależności od chwilowych potrzeb.

Postawione cele

Przed pojawieniem się CSS wszystkie informacje dotyczące wyglądu dokumentów HTML (między innymi rodzaj i kolor czcionki, ułożenie, marginesy) zawarte były w znacznikach HTML. Język CSS umożliwił przeniesienie tych informacji do osobnego pliku. Skutkuje to uproszczeniem i zwiększeniem przejrzystości samego dokumentu HTML. Bez użycia CSS w przypadku definiowania stylu dla nagłówka (h1) lub podtytułu (h2) jego definicja musiałaby zostać powtórzona w każdym miejscu, w którym pojawia się dana struktura. Efektem takiego działania byłoby zmniejszenie czytelności dokumentu i jego odporności na błędy oraz trudność w utrzymaniu. Dodatkowo, zmiany stylu w jednym z miejsc wiązałyby się z koniecznością wprowadzania zmian w każdym miejscu wystąpienia. CSS pozwala na rozgraniczenie warstwy prezentacji od struktury. Język ten umożliwił definicję kolorów, czcionek, układu, rozmiarów, marginesów oraz wielu innych cech związanych z warstwą prezentacji.

CSS Frameworks

CSS Frameworks to specjalnie stworzone biblioteki, których zadaniem jest ułatwienie definicji stylu dla stron internetowych z wykorzystaniem kaskadowych arkuszy stylów. Przykładem takich bibliotek są: Foundation, Blueprint, Bootstrap oraz Cascade Framework. Podobnie jak w przypadku programowania i bibliotek, CSS Frameworks są zazwyczaj dołączane jako zewnętrzne pliki *.css* w znaczniku HTML nad. Zapewniają one gotowe rozwiązania do projektowania stron.

Zadanie 4 – Listy

Zmodyfikuj stronę internetową utworzoną w poprzednim zadaniu. Wprowadź następujące zmiany:

- 1) W akapicie #1 należy utworzyć listę nienumerowaną z kwadratowymi markerami jak na rysunku poniżej:

Krótką historia CSS

Pierwotnie HTML był językiem wyłącznie do opisu struktury dokumentu. Jednak z czasem zrodziła się potrzeba ożywienia wyglądu takich dokumentów. Powoli dodawano nowe znaczniki do HTML pozwalające kontrolować kolory, typografię, dodawać nowe media (np. obrazki). Te niestandardowe rozszerzenia realizowane były przez najpopularniejszych producentów przeglądarek bez porozumienia z drugim. Doprowadziło to do zaimplementowania nowych znaczników działających w konkretnej grupie przeglądarek i nie działających w innych przeglądarkach. Projektanci zostali zmuszeni do wysyłania do klienta różnych wersji tej samej witryny w zależności od użytej przeglądarki, uzyskanie identycznego wyglądu w różnych przeglądarkach było praktycznie niemożliwe.

Chronologia etapów rozwoju języka CSS:

- Håkon Wium Lie jako pierwszy zaproponował CHSS (Cascading HTML Style Sheets) w październiku 1994 roku. Później Lie i Bert Bos pracowali wspólnie nad standardem CSS (literka H została usunięta ze względu na możliwość stosowania stylów do innych podobnych do HTML języków).
- W tym czasie została utworzona organizacja World Wide Web Consortium, która z Lie'em i Bossem na czele przejęła prace nad CSS. Pod koniec 1996 roku wydano oficjalną dokumentację CSS, Kaskadowe arkusze stylów, poziom 1.
- W3C zatwierdziło dwa oficjalne standardy CSS: CSS 1 i CSS 2, a także dnia 7 czerwca 2011 roku standard CSS 2.1.
- Trwają również prace nad CSS3. Wersja ta w stosunku do poprzedników wzbogaci się o wiele selektorów oraz właściwości, nowością jest także modułowy charakter języka – nie będzie to już jednolita rekomendacja, lecz kilkadziesiąt osobnych dokumentów, co pozwoli na włączanie lub wyłączanie odpowiednich modułów w przeglądarkach w zależności od chwilowych potrzeb.

Więcej na temat historii CSS można dowiedzieć się [TUTAJ](#).

Tekst rozpoczynający listę (Chronologia ...:) ma być pochylony i podkreślony. Margines prawy punktów listy należy ustawić na poziomie 60px. Do tekstu podpisu (pod listą) należy zastosować blur-efekt o parametrach np. 2px 2px 5px red. W podpisie należy umieścić odsyłacz do strony internetowej (po kliknięciu na TUTAJ):

<https://www.w3.org/Style/CSS20/history.html>

- 2) W akapicie #2 należy utworzyć listę numerowaną główną oraz dodatkową ze sposobem numeracji jak na rysunku poniżej:

Postawione cele

Przed pojawieniem się CSS wszystkie informacje dotyczące wyglądu dokumentów HTML (między innymi rodzaj i kolor czcionki, ułożenie, marginesy) zawarte były w znacznikach HTML.

Poniżej przedstawiono główne zalety kaskadowych arkuszy stylów:

1. Język CSS umożliwia przeniesienie tych informacji do osobnego pliku. Skutkuje to uproszczeniem i zwiększeniem przejrzystości samego dokumentu HTML;
2. CSS pozwala na rozgraniczenie warstwy prezentacji od struktury;
3. Język ten umożliwia definicję:
 - a. kolorów,
 - b. czcionek,
 - c. układu,
 - d. rozmiarów,
 - e. marginesów oraz wielu innych cech związanych z warstwą prezentacji.

Jeżeli chcesz dowiedzieć się więcej o zaletach CSS kliknij w [LINK](#).

Tekst rozpoczynający listę (Poniżej przedstawiono ...:) ma być pogrubiony i podkreślony. Margines prawy punktów listy należy ustawić na poziomie 60px. Do tekstu podpisu (pod listą) należy zastosować efekt cienia o parametrach np. 2px 2px yellow. W podpisie należy umieścić odsyłacz do strony internetowej (po kliknięciu na LINK):

<https://vanseodesign.com/css/benefits-of-cascading-style-sheets/>

* Żeby zamieścić kilka list o różnych konfiguracjach w jednym pliku HTML można im nadawać różne id="nazwa" po czym w pliku .css definiować ich właściwości:

```
<ol id="Lista2">Zawartość listy</ol> - w pliku HTML
#Lista2>li{właściwości} - w pliku .css
```


3) W akapicie #3 należy utworzyć listę numerowaną jak na rysunku poniżej:

CSS Frameworks

CSS Frameworks to specjalnie stworzone biblioteki, których zadaniem jest ułatwienie definicji stylu dla stron internetowych z wykorzystaniem kaskadowych arkuszy stylów. Przykładem takich bibliotek są:

- I. [Foundation](#),
- II. [Blueprint](#),
- III. [Bootstrap](#),
- IV. [Cascade Framework](#).

Podobnie jak w przypadku programowania i bibliotek, CSS Frameworks są zazwyczaj dołączane jako zewnętrzne pliki .css w znaczniku HTML head. Zapewniają one gotowe rozwiązania do projektowania stron.

Dostosuj swój kod w taki sposób żeby uzyskać podobny efekt. Podaj właściwy link do każdej biblioteki (Frameworków) CSS. Jeżeli masz problem ze znalezieniem takowych możesz skorzystać z następujących:

[https://en.wikipedia.org/wiki/Foundation_\(framework\)](https://en.wikipedia.org/wiki/Foundation_(framework)) – *Foundation*;

<http://www.blueprintcss.org/> – *Blueprint*;

<https://getbootstrap.com/docs/3.4/css/> – *Bootstrap*;

https://en.wikipedia.org/wiki/Cascade_Framework – *Cascade*.

4) opracuj wnioski na temat tworzenia i stylizacji list przy użyciu CSS.