

Aplikacje WWW

Lista 2/2020

1. W nowym dokumencie `html` umieść następujący przycisk:

```
<button style="position:absolute"
onmouseover="this.style.left=''+(Math.random()*100)+'%';"
>Kliknij</button>
```

Wypróbuj i wyjaśnij jego działanie w paragrafie tekstu pod przyciskiem. Zauważ, że przycisk przemieszcza się w poziomie. Dodaj do dokumentu przycisk w kolorze zielonym przemieszczający w pionie, oraz przycisk czerwony przemieszczający się w poziomie i w pionie.

2. Do dokumentu dodaj przyciski, które będą zmieniać kolor tła dokumentu. Wskazówka: Kilka sposobów, jak to zrobić, było pokazane na wykładzie 1.
3. `<body><div></div><div></div><div></div><div></div></body>`

Do dokumentu o ciebie jak wyżej, dodaj taki `<style>` w nagłówku, aby dokument przedstawiał 4 kwadraty o boku 40px każdy, w czterech rogach okienka przeglądarki, każdy kwadrat w innym kolorze. Użyj `div {position:absolute}` oraz selektora `div:nth-child` lub `div:nth-of-type` aby ustawić atrybuty `width`, `height`, `left`, `right`, `top`, `bottom`, `background` osobno dla każdego elementu `div`. Nie możesz zmienić ciała dokumentu, wszystkie polecenia `css` powinny być umieszczone w znaczniku `<style>` w nagłówku.

Wskazówka: https://www.w3schools.com/cssref/css_selectors.asp

4. Przygotuj tabelę:

```
<table class="A">
<tr><th></th><th></th><th></th></tr>
<tr><td></td><td></td><td></td></tr>
<tr><td></td><td></td><td></td></tr>
<tr><td></td><td></td><td></td></tr>
</table>
```

oraz drugą, gdzie zamiast `class="A"` będzie `class="B"`. W nagłówku dokumentu zdefiniuj styl, który spowoduje, że komórki obu tabel będą się różniły, czcionką, wytłuszczeniem, justowaniem, kolorem czcionki, ramki i tła. Wskazówka: użyj selektorów `table.A {...}`, `table.A th {...}`, `table.A td {...}`, `table.A tr:nth-child(2n) td {...}` itp. by ustawić własności tabeli klasy A, jej nagłówków, komórek, komórek w parzystych wierszach itd. i analogicznie dla tabeli klasy B. Zawartości znaczników `<td>` i `<th>` wypełnij tekstem, ale **nie** dodawaj do powyższego kodu żadnych atrybutów (w szczególności `style`, `class`, `id`).

5. Sprawdź do czego służą atrybuty `rowspan` i `colspan`. Wymyśl dwa przykłady tabel, gdzie będą one miały sensowne zastosowanie. Zadbaj o estetyczny wygląd i sensowną treść obu tabel. Nie używaj lokalnych stylów (atrybutu `style`, tylko zastosuj technikę z poprzedniego zadania stosując do znaczników `table` różne klasy (np. `class='A'` oraz `class="B"`). Zastosuj następujące atrybuty CSS: `border`, `padding`, `color`, `background`, `text-align`, `vertical-align`, `margin`, `font-weight`.

Wskazówka: https://developer.mozilla.org/en-US/docs/Learn/CSS/Building_blocks/Styling_tables

6. Napisz tekst zawierający kilka elementów p. Zobacz co się stanie gdy do nagłówka dodasz `<style>p:hover {font-weight:bold;background:yellow}</style>`.
7. Utwórz dokument HTML zawierający w sekcji body pole tekstowe:


```
<input onchange="document.body.innerHTML
+= '<br>' + this.value + ' ' + eval(this.value)">
```

 oraz przycisk `<button onclick="location=location">Wyczyść</button>`. W przeglądarce wpisz do pola tekstowego $2+2$ i wciśnij Enter. Wypróbuj inne działania oraz funkcje. Nazwy funkcji trzeba poprzedzać słowem `Math`. np `Math.sin(Math.PI/2)`, lub `Math.sqrt(1+2*4)`, a potęgowanie to np. `Math.pow(2,5)`. Dodaj do dokumentu paragraf tekstu wyjaśniający zasadę działania tego “kalkulatora”.
8. W kalkulatorze z poprzedniego zadania wpisuj kolejno następujące działania: $x=5$, $y=2$, $x+y$, $x-y$, $x*y$, x/y i zatwierdzaj każde z nich klawiszem Enter. Następnie rozwiąż równanie kwadratowe $2x^2 + 3x - 5 = 0$ wpisując i zatwierdzając kolejno: $a=2$, $b=3$, $c=-5$, $\Delta=b^2-4*a*c$, $pierw=Math.sqrt(\Delta)$, $x1=(-b-pierw)/(2*a)$, $x2=(-b+pierw)/(2*a)$. Następnie w ten sam sposób rozwiąż inne równanie kwadratowe. Wykonanie zadania udokumentuj zrzutem ekranu o nazwie `zadanie8.png`.
9. Zmodyfikuj zadanie 7 tak, by wyniki pojawiały się w dodanej nad paragrafem wyjaśnień ramce `<div id="wynik"></div>`. Dopracuj estetykę dokumentu. Dodaj tytuł, ustaw marginesy, czcionki itd. Wskazówka: napis `document.body.innerHTML` zastąp napisem `document.getElementById('wynik').innerHTML`.
10.


```
<input>x<sup>2</sup>+<input>x+<input>=0 <hr>
<button onclick="solve()">Rozwiąż</button>
<div></div>
<script>
function solve(){
  let inp=document.getElementsByTagName("input");
  let a=inp[0].value, b=inp[1].value, c=inp[2].value;
  let delta=b*b-4*a*c;
  let x1=(-b-Math.sqrt(delta))/(2*a);
  let x2=(-b+Math.sqrt(delta))/(2*a);
  document.getElementsByTagName("div")[0].innerHTML=
    "x1="+x1+"<br>x2="+x2;
}
</script>
```

W nowym dokumencie powyższy kod wpisz jako zawartość znacznika `<body>`. Do sekcji `<head>` dodaj `<style>`, który ustawi szerokość pól tekstowych na 30px, a następnie wypróbuj działanie dokumentu. Zmień funkcję `solve()` tak, by w przypadku jeśli $\Delta < 0$ pojawiał się napis "Brak rozwiązań" a jeśli $\Delta \geq 0$ pojawiała się jedno rozwiązanie $x_0 = \dots$ (Patrz: https://www.w3schools.com/js/js_if_else.asp). W paragrafie na końcu dokumentu wyjaśnij, co robi każda linia poprawionej funkcji `solve()`.
11. Naucz się stosować w dokumencie google fonts (<https://fonts.google.com/>). Wykonaj prosty dokument, w którym zastosujesz ciekawą, nietypową czcionkę do nagłówków: H1, H2, H3, oraz inną czcionkę do body oraz paragrafów p. Styl czcionki powinien, w miarę możliwości, pasować do treści dokumentu.
12. Wykonaj logo “swojej firmy” rozmieszczając poszczególne litery zamknięte w znacznikach `span` wewnątrz ramki `div`. Litery mogą mieć różne wielkości, kolory i kroje. Ważne jest niekonwencjonalne pozycjonowanie lub przeplatanie się liter. Wskazówka: zastosuj styl `div {position:relative}` oraz `span {position:absolute}`, a w atrybucie `style` elementów `` ustaw różne wartości altrybutów CSS: `left`, `top`,

font-size, color, background, transform, text-shadow, border-radius, itd.. Transform pozwala obracać litery o pewien kąt. Możesz zastosować również google fonts z poprzedniego zadania

- Umieść w folderze `img` 10 zdjęć o nazwach od `a0.jpg` do `a9.jpg`. W ciele dokumentu wpisz:

```
<button onclick="f(-1)"> << </button>

<button onclick="f(1)"> >> </button>
<script>
  var i=0;
  var img=document.getElementsByTagName("img")[0];
  function f(x){
    i=i+x;
    img.src="img/a"+i+".jpg";
  }
</script>
```

Zastosuj style `img{vertical-align:middle;}` oraz `body {text-align:center;}` i ustal wysokość obrazka na 400px.

Zmienna `var i` pamięta numer zdjęcia (początkowo 0). Po kliknięciu lewego/prawego guzika ta zmienna jest zwiększana/zmniejszana o 1 i zmieniany jest atrybut `src` obrazka, co powoduje załadowanie odpowiedniej grafiki. Zadbaj, by nie było możliwe wyjście poza zakres tzn. po dodaniu `x`, jeśli `i==10`, to powinno być zmienione na 0, a jeśli `i<0`, to powinno być zmienione na 9. W przeglądarce chrome kliknij prawym przyciskiem myszy na obrazku i wybierz opcję `zbadaj`. Zaobserwuj jak atrybut `src` zmienia się w trakcie klikania na guziki `<<` oraz `>>`.

- Wykonaj druga wersję tego zadania, gdzie adresy zdjęć znajdują się w tablicy:
`var adr=["img/a0.jpg", "img/a1.jpg", ... , "img/a9.jpg"]`. a po zmianie wartości `i` wykonywana jest instrukcja `img.src=adr[i]`, czyli pobierany jest `i`-ty adres z tablicy `adr`. Następnie zastąp lokalne adresy w tablicy `adr`, odnośnikami do grafik w internecie, a zamiast liczby 10 zastosuj `adr.length` czyli długość tej tablicy. Zwiększ ilość adresów w tablicy i upewnij się, że galeria mimo tego działa poprawnie.
- Inny sposób wykonania galerii jest następujący. W dokumencie z zadania 13 zamiast jednego umieść 10 obrazków, ale 9 z nich ukryj stosując styl `display:none`. Zmodyfikuj funkcję `f()` dodając przed zmianą zmiennej `i` ukrycie `i`-tego obrazka: `document.images[i].style.display='none'` a po zmianie `i` pokazanie `i`-tego obrazka: `document.images[i].style.display='inline'`.

Zadbaj o to, by we wszystkich dokumentach były prawidłowo ustawione wartości `charset`. Każdy dokument powinien zawierać znacznik `<title>` z sensowną zawartością, czyli numerem lub tematem zadania, np. Kalkulator, Galeria, Logo, itp.