Graad 8

Les 1: Inleiding tot CSS

Lesdoelwitte

• Verstaan ​​wat CSS (Cascading Style Sheets) is en hoe dit HTML verbeter

• Leer hoe om CSS-style op HTML-elemente toe te pas

• Oefen die stilering van teks, agtergronde en uitleg met behulp van CSS

Wat is CSS?

CSS (Cascading Style Sheets) is 'n taal wat gebruik word om die uitleg van HTML-elemente op 'n webblad te stileer en te formateer. Dit laat jou toe om die voorkoms van die inhoud te beheer, soos kleure, lettertipes, spasiëring, posisionering en meer.

Terwyl HTML die struktuur van 'n webblad verskaf, word CSS gebruik om dit te stileer en dit visueel aantreklik te maak.

CSS-sintaksis

'n Basiese CSS-reël bestaan ​​uit:

• Selector: Die HTML-element wat jy wil stileer (bv. h1, p, .class, #id)

• Property: Die styl wat jy wil toepas (bv. color, font-size, marge)

• Value: Die waarde van die eienskap (bv. red, 16px, 10px)

selector {

property: value;

}

Voeg CSS by HTML

Daar is drie hoofmaniere om CSS by 'n HTML-dokument te voeg:

1. Inline CSS: Gebruik die style-attribuut binne 'n HTML-etiket

2. Interne CSS: Plaas CSS-reëls binne 'n <style>-etiket in die <head>-afdeling

3. Eksterne CSS: Skakel na 'n eksterne CSS-lêer met behulp van die <link>-etiket

Voorbeeld 1: Inline CSS

Inline CSS word direk by die HTML-etiket gevoeg met behulp van die style-attribuut.

<!DOCTYPE html>

<html>

<head>

<title>Inlyn CSS Voorbeeld</title>

</head>

<body>

<h1 style="color: blue;">Welkom by My Webwerf</h1>

<p style="font-size: 18px; color: green;">Hierdie is 'n inlyn CSS voorbeeld.</p>

</body>

</html>

Voorbeeld 2: Interne CSS

Interne CSS word binne die <style>-etiket in die <head>-afdeling van die HTML-dokument geplaas.

<!DOCTYPE html>

<html>

<head>

<title>Interne CSS Voorbeeld</title>

<style>

h1 {

color: purple;

}

p {

font-size: 20px;

color: darkblue;

}

</style>

</head>

<body>

<h1>Welkom by my webwerf</h1>

<p>Hierdie is 'n interne CSS-voorbeeld.</p>

</body>

</html>

Voorbeeld 3: Eksterne CSS

Eksterne CSS word in 'n aparte .css-lêer geplaas, wat aan die HTML-lêer gekoppel is.

• CSS-lêer (styles.css):

h1 {

kleur: rooi;

}

p {

lettergrootte: 18px;

kleur: swart;

}

• HTML-lêer (index.html):

<!DOCTYPE html>

<html>

<head>

<title>Eksterne CSS-voorbeeld</title>

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

</head>

<body>

<h1>Welkom by my webwerf</h1>

<p>Hierdie is 'n eksterne CSS-voorbeeld.</p>

</body>

</html>

CSS-eienskappe

Hier is 'n paar algemene CSS-eienskappe wat gebruik word om teks en elemente te stileer:

• Kleur: Verander die kleur van teks of agtergrond

p {

kleur: blou;

}

• Lettergrootte: Pas die grootte van teks aan

p {

lettergrootte: 16px;

}

• Agtergrondkleur: Verander die agtergrondkleur van 'n element

body {

agtergrondkleur: liggrys;

}

• Marge: Voeg spasie rondom elemente by

p {

marge: 20px;

}

• Opvulling: Voeg spasie binne 'n element by (rondom die inhoud)

p {

opvulling: 10px;

}

• Teksbelyning: Rig teks horisontaal (links, middel, regs)

h1 {

teksbelyning: middel;

}

Oefeninge

1. Stileer 'n opskrif

Skep 'n opskrif (<h1>) en stileer dit om 'n ander kleur en grootte te hê.

2. Stileer 'n paragraaf

Voeg 'n paragraaf (<p>) by en verander die lettergrootte, kleur en voeg 'n bietjie opvulling by.

3. Skep 'n bladsyagtergrond

Verander die agtergrondkleur van jou bladsy met behulp van CSS.

Uitdagingstaak

Skep 'n webblad met die volgende style:

• 'n Opskrif (<h1>) wat gesentreer is en 'n rooi kleur het

• 'n Paragraaf (<p>) wat 'n blou lettertipekleur, 18px lettergrootte en 10px opvulling het

• 'n Agtergrondkleur van jou keuse vir die webblad

Opsomming

• CSS word gebruik om HTML-elemente te stileer en die voorkoms van 'n webblad te verbeter

• Selektors word gebruik om HTML-elemente te teiken, en eienskappe word toegepas om hulle te stileer

• CSS kan op drie maniere bygevoeg word: inlyn, intern of ekstern

• Algemene CSS-eienskappe sluit in kleur, lettertipegrootte, agtergrondkleur, marge, opvulling en teksbelyning

Vasvrae

1. Hoe voeg jy CSS by 'n HTML-dokument?

a) Met die <style>-etiket in die <body>

b) Met die style-attribuut binne die HTML-etiket

c) Met die <link>-etiket in die <head>

d) Al die bogenoemde

2. Watter van die volgende is 'n geldige CSS-eienskap vir die verandering van die tekskleur?

a) lettergrootte

b) kleur

c) agtergrondkleur

d) rand

3. Hoe skakel jy 'n eksterne CSS-lêer aan 'n HTML-dokument?

a) <style src="styles.css">

b) <link rel="stylesheet" href="styles.css">

c) <css src="styles.css">

d) <link href="styles.css" rel="style">

4. Wat doen die padding-eienskap in CSS?

a) Voeg spasie rondom 'n element by

b) Verander die lettergrootte

c) Verander die agtergrondkleur

d) Voeg spasie binne 'n element by

5. Watter van die volgende word gebruik om teks in CSS te sentreer?

a) text-align: center;

b) font-align: center;

c) center-text: true;

d) align-text: center;

Antwoordsleutel:

1 - d, 2 - b, 3 - b, 4 - d, 5 – a

**Les 2**: CSS Boksmodel

Lesdoelwitte

• Verstaan ​​die CSS-boksmodel en hoe elke HTML-element as 'n boks behandel word

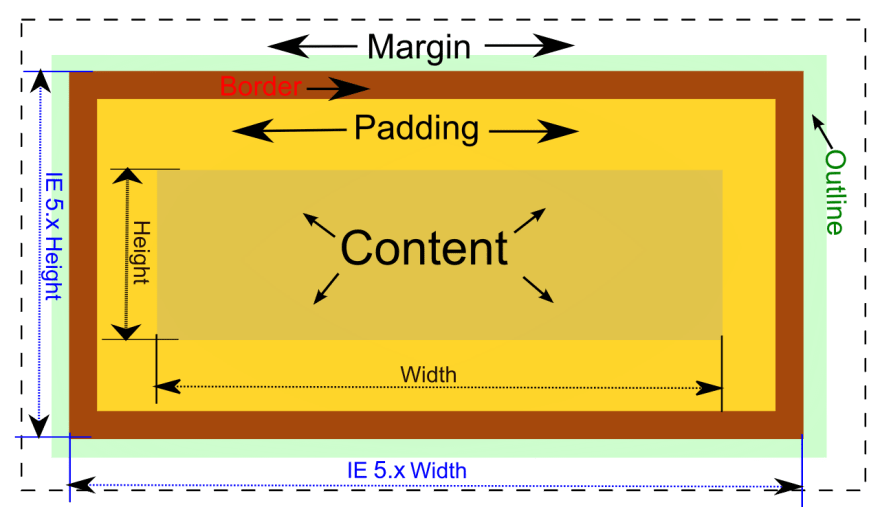
• Leer die vier dele van die boksmodel: inhoud, opvulling, rand en kantlyn

• Oefen die gebruik van CSS om spasiëring en uitleg te beheer

Wat is die CSS Boksmodel?

In CSS word elke element op 'n webblad as 'n boks behandel. Die boksmodel beskryf die ruimte wat elke HTML-element omring en help jou om uitleg en spasiëring te beheer.

Die CSS Boksmodel sluit in:



Boksmodeldele

| **Deel** | **Beskrywing** |
| --- | --- |
| Inhoud | Die werklike teks of beeld binne die element |
| Opvulling | Ruimte tussen die inhoud en die rand |
| Rand | Die lyn wat die opvulling en inhoud omvou |
| Kantlyn | Die spasie buite die rand, wat elemente skei |

Voorbeeld

<!DOCTYPE html>

<html>

<head>

<title>Boksmodelvoorbeeld</title>

<style>

.box {

agtergrondkleur: ligblou;

opvulling: 20px;

rand: 5px soliede vlootblou;

kantlyn: 30px;

}

</style>

</head>

<body>

<h1>CSS Boksmodel</h1>

<div class="box">

Hierdie is 'n boks met opvulling, rand en kantlyn.

</div>

</body>

</html>

In hierdie voorbeeld:

• Die inhoud is die teks binne die boks

• Die opvulling skep spasie binne die boks

• Die rand is 'n dik blou lyn

• Die kantlyn skep spasie rondom die boks

Hoe die Boksmodel die Uitleg Beïnvloed

Die totale breedte van 'n boks is:

Totale Breedte = inhoud + opvulling + rand + kantlyn

Dieselfde geld vir hoogte.

As jou inhoud 200px breed is, die opvulling 10px, die rand 5px en die kantlyn 20px, dan:

Totale Breedte = 200 + 102 + 52 + 20\*2 = 270px

Oefeninge

1. Skep 'n Boks

Gebruik 'n <div> met klasboks wat die volgende insluit:

• 'n Agtergrondkleur

• 15px opvulling

• 3px soliede rand

• 25px kantlyn

2. Voeg Verskeie Bokse by

Maak twee bokse met verskillende agtergrondkleure en spasiëring. Let op hoe die kantlyn hulle skei.

3. Pas Opvulling en Kantlyn aan

Eksperimenteer deur die opvulling- en kantlynwaardes te verander en kyk hoe die boks beweeg en van grootte verander.

Uitdagingstaak

Skep drie blokkies op 'n bladsy:

• Eerste blokkie: blou agtergrond, 10px-opvulling, 5px-marge

• Tweede blokkie: groen agtergrond, 20px-opvulling, 15px-marge

• Derde blokkie: oranje agtergrond, 30px-opvulling, 25px-marge

Stil elkeen met 'n rand en maak seker dat hulle duidelik geskei is met behulp van marges.

Opsomming

• Die blokkiemodel is die fondament van CSS-uitleg

• Elke element het: Inhoud → Opvulling → Rand → Marge

• Gebruik opvulling om spasie binne die blokkie te beheer

• Gebruik marge om spasie tussen blokkies te skep

• Rande kan gestileer word met breedte, kleur en tipe (solied, gestippel, ens.)

Vasvra

1. Watter deel van die blokkiemodel is die naaste aan die teks of inhoud?

a) Rand

b) Marge

c) Opvulling

d) Inhoud

2. Watter eienskap voeg spasie binne die rand by, maar buite die inhoud?

a) Marge

b) Opvulling

c) Rand

d) Breedte

3. Wat doen die marge-eienskap?

a) Voeg spasie binne die element by

b) Voeg spasie tussen elemente by

c) Voeg 'n rand by

d) Stel die tekskleur

4. As 'n element 10px-opvulling, 2px-rand en 5px-marge het, hoeveel spasie word rondom die inhoud bygevoeg?

a) 17px

b) 10px

c) 5px

d) 30px

5. Watter CSS-reël stel opvulling korrek op 20px?

a) opvulling: 20;

b) opvulling: "20px";

c) opvulling: 20px;

d) opvulling = 20px;

Antwoordsleutel:

1 - d, 2 - b, 3 - b, 4 - a, 5 – c

**Les 3**: CSS Teksstilering

Lesdoelwitte

• Leer hoe om die voorkoms van teks met CSS te verander

• Tekskleur, grootte, spasiëring, belyning, versiering en transformasie stileer

• Oefen die toepassing van teksstyle op verskillende HTML-elemente

Waarom teks met CSS stileer?

HTML-teks lyk standaard eenvoudig. Met CSS kan jy:

• Teks meer leesbaar maak

• Visuele belangstelling byvoeg

• 'n Ontwerptema pas

Algemene CSS Tekseienskappe

Eienskap Wat dit doen Voorbeeld

kleur Verander die tekskleur kleur: blou;

font-size Verander die grootte van die teks font-size: 20px;

text-align Rig teks (links, middel, regs) text-align: center;

font-family Verander die fontstyl font-family: Arial;

font-weight Maak teks vetdruk font-weight: bold;

text-decoration Onderstreep of verwyder onderstreep text-decoration: none;

text-transform Maak teks hoofletters of kleinletters text-transform: uppercase;

letter-spasiering Voeg spasie tussen letters by letter-spasiering: 2px;

line-height Stel die spasiëring tussen teksreëls line-height: 1.5;

Voorbeeld: Teksstilering

<!DOCTYPE html>

<html>

<head>

<title>Voorbeeld van teksstilering</title>

<style>

h1 {

color: donkerrooi;

font-size: 32px;

text-align: center;

}

p {

font-family: Georgia, serif;

font-size: 18px;

line-height: 1.6;

color: donkerslategrys;

text-transform: capitalize;

}

.highlight {

font-weight: bold;

text-decoration: underline;

color: oranje;

}

</style>

</head>

<body>

<h1>Welkom by My Bladsy</h1>

<p>Hierdie is 'n voorbeeld van gestileerde teks met behulp van CSS.</p>

<p class="highlight">Hierdie teks word met 'n klas uitgelig.</p>

</body>

</html>

Oefeninge

1. Stileer 'n Opskrif en Paragraaf

Verander die kleur en belyning van 'n opskrif, en die lettertipe van 'n paragraaf.

2. Voeg 'n Teksklas by

Skep 'n CSS-klas wat teks vetdruk, onderstreep en blou maak. Pas dit toe op 'n <p>- of <h2>-etiket.

3. Gebruik Tekstransformasie

Maak 'n paragraaf wat teks in hoofletters wys met behulp van tekstransformasie.

Uitdagingstaak

Skep 'n kort webblad wat die volgende insluit:

• 'n Gesentreerde titel in 'n deftige lettertipe

• 'n Paragraaf met 'n persoonlike lettergrootte en lynhoogte

• 'n Uitgeligte aanhaling met vetgedrukte, kursiewe en gekleurde teks

•

Opsomming

• CSS laat teks beter lyk en makliker leesbaar wees

• Gebruik eienskappe soos kleur, lettergrootte, teksbelyning en lettertipefamilie om voorkoms te verander

• Jy kan CSS-klasse skep om style op spesifieke dele van teks toe te pas

• Gebruik tekstransformasie, letterspasiëring en lynhoogte om leesbaarheid te verfyn

Vasvra

1. Watter eienskap verander die tekskleur?

a) lettertipestyl

b) kleur

c) teksstyl

d) lettertipekleur

2. Wat doen teksbelyning: middelpunt;?

a) Maak die teks vetgedruk

b) Plaas die teks aan die regterkant

c) Sentreer die teks

d) Verander tekskleur na grys

3. Hoe maak jy teks alles hoofletters?

a) tekstransformasie: hoofletter;

b) tekshoofletter: waar;

c) teks-transform: hoofletters;

d) lettertipe-geval: hoofletters;

4. Watter eienskap verander die spasiëring tussen lyne?

a) lynhoogte

b) letterspasiëring

c) lettertipe-gaping

d) marge

5. Watter CSS-reël verwyder onderstreep uit teks?

a) teks-versiering: geen;

b) teks-onderstreep: vals;

c) lettertipe-styl: normaal;

d) teks-geen-onderstreep: ja;

Antwoordsleutel:

1 - b, 2 - c, 3 - c, 4 - a, 5 – a

**Graad 9**

**Les 1**: CSS-posisionering

Lesdoelwitte

• Verstaan ​​die verskillende CSS-posisioneringskemas

• Leer hoe om elemente te posisioneer deur staties, relatief, absoluut, vas en klewerig te gebruik

• Pas posisionering toe om dinamiese en responsiewe uitlegte te skep

Wat is CSS-posisionering?

CSS-posisionering laat jou toe om die uitleg en plasing van elemente op 'n webblad te beheer. Deur die posisie-eienskap te verander, kan jy elemente relatief tot hul normale posisie, hul ouer of die viewport skuif.

Tipes posisionering

1. Staties (Standaard)

o Elemente word geposisioneer volgens die normale vloei van die dokument.

o Dit is die standaardwaarde; elemente word nie beïnvloed deur bo-, onder-, links- of regs-eienskappe nie.

2. Relatief

o Die element word relatief tot sy normale posisie geposisioneer.

o Jy kan bo, regs, onder en links gebruik om die element te skuif.

3. Absoluut

o Die element word relatief tot die naaste geposisioneerde voorouer geposisioneer (nie staties nie).

o As daar geen sodanige voorouer is nie, gebruik dit die dokumentliggaam.

4. Vas

o Die element word relatief tot die blaaiervenster geposisioneer.

o Dit bly in plek selfs wanneer die bladsy geskrol word.

5. Kleef

o Die element wissel tussen relatief en vas, afhangende van die skrolposisie.

o Dit bly by 'n gegewe posisie terwyl jy verby dit skrol.

Voorbeeld: Posisionering van Elemente

<!DOCTYPE html>

<html>

<head>

<title>CSS Posisionering Voorbeeld</title>

<style>

.relative-box {

position: relative;

top: 20px;

left: 30px;

background-color: lightgreen;

width: 150px;

height: 100px;

}

.absolute-box {

position: absolute;

top: 50px;

left: 50px;

background-color: lightcoral;

width: 150px;

height: 100px;

}

.vaste-boks {

posisie: vas;

onder: 10px;

regs: 10px;

agtergrondkleur: ligblou;

breedte: 150px;

hoogte: 100px;

}

.klewerige-boks {

posisie: klewerig;

bo: 0;

agtergrondkleur: liggeel;

opvulling: 10px;

}

</style>

</head>

<body>

<div class="sticky-box">Kleefkop</div>

<div class="relative-box">Relatiewe blokkie</div>

<div class="absolute-box">Absolute blokkie</div>

<div class="fixed-box">Vaste blokkie</div>

<p style="margin-top: 200px;">Blaai af om te sien hoe die vaste blokkie in plek bly.</p>

</body>

</html>

Oefeninge

1. Skep 'n relatiewe element

o Ontwerp 'n <div> met posisie: relatief; en skuif dit 20px af en 30px na regs.

2. Absolute posisionering

o Plaas 'n kind <div> binne 'n ouer <div>. Stel die ouer na posisie: relatief; en die kind na posisie: absolute; met spesifieke boonste en linker waardes.

3. Vaste navigasiebalk

o Skep 'n navigasiebalk wat bo-aan die bladsy bly met behulp van posisie: fixed;

4. Kleefsybalk

o Ontwerp 'n sybalk wat bo-aan die kykvenster vassit wanneer jy blaai met behulp van position: sticky;

Uitdagingstaak

Bou 'n webbladuitleg met die volgende:

• 'n Vaste koptekst wat bo-aan bly

• 'n Kleefnavigasiekieslys wat onder die koptekst vassit wanneer jy blaai

• 'n Absoluut geposisioneerde beeld binne 'n relatiewe houer

• 'n Vaste voetskrif wat onderaan die kykvenster bly

Opsomming

• CSS-posisionering laat presiese beheer oor elementplasing toe.

• Verskillende posisioneringskemas (staties, relatief, absoluut, vas, kleef) dien verskeie uitlegbehoeftes.

• Die kombinasie van posisionering met ander CSS-eienskappe maak komplekse en responsiewe ontwerpe moontlik.

Vasvrae

1. Watter posisioneringswaarde is die verstekwaarde vir HTML-elemente?

a) relatief

b) absoluut

c) staties

d) vas

Wat doen position: absolute;? a) Posisioneer die element relatief tot sy normale posisie

b) Posisioneer die element relatief tot die naaste geposisioneerde voorouer

c) Posisioneer die element relatief tot die kykvenster

d) Hou die element in plek tydens blaai

Watter posisioneringswaarde laat 'n element toe om in plek te bly, selfs wanneer die bladsy geblaai word?

a) relatief

b) absoluut

c) vas

d) klewerig

Wat word vereis vir posisie: klewerig; om behoorlik te werk?

a) Die element moet 'n gedefinieerde boonste, linker-, regter- of onderwaarde hê

b) Die element moet binne 'n vaste houer wees

c) Die element moet 'n z-indeks hê

d) Die element moet absoluut geposisioneer wees

As 'n ouerelement posisie: relatief; het en 'n kind posisie: absoluut; het, waar is die kind geposisioneer?

a) Relatief tot die kykvenster

b) Relatief tot die ouerelement

c) Relatief tot die dokumentliggaam

d) Relatief tot sy normale posisie

Antwoordsleutel: 1 - c, 2 - b, 3 - c, 4 - a, 5 – b

**Les 2**: CSS z-indeks

Lesdoelwitte

• Verstaan ​​die doel van die z-indeks-eienskap in CSS

• Leer hoe om die stapelvolgorde van oorvleuelende elemente te beheer

• Pas z-indeks in praktiese scenario's toe om elementlaag te bestuur

Wat is z-indeks?

In CSS bepaal die z-indeks-eienskap die stapelvolgorde van elemente langs die z-as (wat loodreg op die skerm strek). Wanneer elemente oorvleuel, spesifiseer z-indeks watter een bo-op verskyn.

Sleutelpunte

• z-indeks werk slegs op elemente met 'n ander posisiewaarde as staties (d.w.s. relatief, absoluut, vas of klewerig)

• Elemente met hoër z-indekswaardes verskyn voor dié met laer waardes.

• z-indeks kan positiewe, negatiewe of nulwaardes aanneem.

• Die standaard z-indeks is outomaties, wat die volgorde van elemente in die HTML volg.

Sintaksis

element {

posisie: relatief;

z-indeks: 1;

}

Voorbeeld: Oorvleuelende Bokse

<!DOCTYPE html>

<html>

<head>

<title>z-indeks Voorbeeld</title>

<style>

.box {

position: absolute;

width: 150px;

height: 150px;

accuracy: 0.8;

}

.box1 {

background-color: red;

top: 50px;

left: 50px;

z-index: 1;

}

.box2 {

background-color: blue;

top: 100px;

left: 100px;

z-index: 2;

}

</style>

</head>

<body>

<div class="box box1">Blokkie 1</div>

<div class="box box2">Blokkie 2</div>

</body>

</html>

Negatiewe z-indeks

Jy kan negatiewe waardes aan die z-indeks toeken om elemente agter ander te plaas.

.element {

position: relative;

z-indeks: -1;

}

Stapelkontekste

'n Stapelkonteks is 'n driedimensionele konseptualisering van HTML-elemente langs die z-as. Elke stapelkonteks is selfstandig; elemente daarbinne word gestapel volgens hul z-indekswaardes, maar hierdie stapeling beïnvloed nie elemente buite die konteks nie.

'n Nuwe stapelkonteks word geskep wanneer 'n element geposisioneer word (relatief, absoluut, vas of klewerig) en 'n ander z-indekswaarde as outo het.

Oefeninge

1. Skep Oorvleuelende Elemente

o Ontwerp drie <div> elemente met verskillende agtergrondkleure.

o Posisioneer hulle om te oorvleuel en ken verskillende z-indekswaardes toe om te beheer watter een bo-op verskyn.

2. Eksperimenteer met Negatiewe z-indeks

o Skep 'n agtergrondbeeld en 'n inhoudsboks.

o Gebruik 'n negatiewe z-indeks om die agtergrondbeeld agter die inhoud te plaas.

3. Verstaan ​​Stapelkontekste

o Skep 'n ouer <div> met 'n z-indeks van 1.

o Plaas twee kind <div>-elemente met verskillende z-indekswaardes daarbinne.

o Neem waar hoe die stapelvolgorde binne die ouerkonteks beïnvloed word.

Uitdagingstaak

Ontwerp 'n modale opspringvenster wat bo-op 'n semi-deursigtige oorlegsel verskyn. Maak seker dat:

• Die oorlegsel die hele bladsyinhoud dek.

• Die modale venster gesentreerd en bo die oorlegsel verskyn.

• Gebruik toepaslike z-indekswaardes om die verlangde lae te bereik.

Opsomming

• Die z-indeks-eienskap beheer die vertikale stapelvolgorde van elemente.

• Dit is slegs van toepassing op geposisioneerde elemente.

• Hoër z-indekswaardes bring elemente na vore.

• Om stapelkontekste te verstaan ​​is noodsaaklik vir die bestuur van komplekse lae.

Vasvrae

Watter CSS-eienskap beheer die stapelvolgorde van elemente?

a) posisie

b) z-indeks

c) vertoon

d) float

Wat is die standaardwaarde van z-indeks?

a) 0

b) outomaties

c) 1

d) -1

Kan z-indeks toegepas word op elemente met posisie: staties?

a) Ja

b) Nee

As twee oorvleuelende elemente dieselfde z-indeks het, watter een verskyn bo-aan?

a) Die een wat eerste in HTML verklaar word

b) Die een wat laaste in HTML verklaar word

c) Die een met hoër ondeursigtigheid

d) Die een met groter afmetings

Wat gebeur as jy 'n negatiewe z-indeks aan 'n element toeken?

a) Dit beweeg voor ander elemente

b) Dit beweeg agter ander elemente

c) Dit word onsigbaar

d) Dit veroorsaak 'n fout

Antwoordsleutel: 1 - b, 2 - b, 3 - b, 4 - b, 5 – b

**Les 3**: CSS Inlynblok

Lesdoelwitte

• Verstaan ​​die display: inlynblok-eienskap in CSS

• Leer hoe inlynblok verskil van inlyn- en blok-vertoonwaardes

• Pas inlynblok toe om horisontale uitlegte sonder drywers te skep

Wat is display: inlynblok?

Die display: inlynblok-eienskap laat 'n element toe om soos 'n inlynelement op te tree terwyl dit die vermoë behou om breedte en hoogte te stel, soortgelyk aan 'n blokelement. Dit beteken dat elemente horisontaal langs mekaar kan sit en steeds gedefinieerde afmetings kan hê.

Vergelyking met ander vertoonwaardes

• display: inlyn:

o Elemente begin nie op 'n nuwe lyn nie.

o Breedte- en hoogte-eienskappe het geen effek nie.

o Boonste en onderste marges en opvullings word nie gerespekteer nie.

• display: blok:

o Elemente begin op 'n nuwe lyn en neem die volle beskikbare breedte op.

o Breedte- en hoogte-eienskappe word gerespekteer.

• display: inlynblok:

o Elemente begin nie op 'n nuwe lyn nie.

o Breedte- en hoogte-eienskappe word gerespekteer. o Boonste en onderste marges en opvullings word gerespekteer.

Dit maak inlynblok 'n veelsydige opsie vir die skep van uitlegte waar elemente horisontaal met spesifieke afmetings in lyn gebring moet word.

Sintaksis

element {

display: inlynblok;

}

Voorbeeld: Horisontale Navigasiekieslys

<!DOCTYPE html>

<html>

<kop>

<styl>

.nav {

agtergrondkleur: #f2f2f2;

opvulling: 10px;

teksbelyning: middelpunt;

}

.nav li {

display: inlynblok;

marge: 0 15px;

opvulling: 10px 20px;

agtergrondkleur: #ddd;

lysstyltipe: geen;

}

</style>

</head>

<body>

<ul class="nav">

<li>Tuisblad</li>

<li>Oor</li>

<li>Dienste</li>

<li>Kontak</li>

</ul>

</body>

</html>

Algemene Gebruiksgevalle

• Skep horisontale navigasie-kieslyste

• Beelde en teks langs mekaar in lyn bring

• Roosteruitlegte bou sonder om drywers te gebruik

• Responsiewe uitlegte ontwerp wat soos nodig omvou

Oefeninge

1. Skep 'n Horisontale Lys

o Ontwerp 'n ongeordende lys waar elke lysitem horisontaal vertoon word met behulp van inlynblok.

2. Beeld- en Onderskrifbelyning

o Plaas 'n beeld en sy onderskrif langs mekaar met behulp van inlynblok, en maak seker dat beide gedefinieerde breedtes en hoogtes het.

3. Responsiewe Kaartuitleg

o Bou 'n stel inhoudskaarte wat horisontaal in lyn kom en na die volgende lyn op kleiner skerms omvou, met behulp van inlynblok.

Uitdagingstaak

Ontwerp 'n responsiewe fotogalery waar elke beeld as 'n miniatuur vertoon word. Gebruik inline-block om die miniatuurprente in rye te rangskik wat omvou gebaseer op die skermgrootte. Voeg sweef-effekte by om die beelde effens te vergroot wanneer dit oor die sweef gehou word.

Opsomming

• display: inline-block kombineer die vloei van inline-elemente met die boks-eienskappe van blok-elemente.

• Dit maak voorsiening vir horisontale belyning van elemente met gedefinieerde afmetings.

• Nuttig vir die skep van uitlegte sonder die kompleksiteite van drywende elemente of flexbox.

Vasvrae

1. Wat laat display: inline-block jou toe om te doen wat display: inline nie doen nie?

a) Stel breedte en hoogte

b) Rig elemente vertikaal

c) Sweef elemente

d) Geen van die bogenoemde nie

Watter vertoonwaarde veroorsaak dat elemente op 'n nuwe lyn begin en die volle breedte opneem?

a) inline

b) inline-block

c) block

d) none

In watter scenario is inline-block die nuttigste?

a) Skep vertikale lyste

b) Rig elemente horisontaal met spesifieke afmetings

c) Versteek elemente

d) Pas animasies toe

Respekteer inline-block boonste en onderste marges en opvullings?

a) Ja

b) Nee

Kan inlynblok-elemente na die volgende lyn vou as daar nie genoeg horisontale spasie is nie?

a) Ja

b) Nee

Antwoordsleutel: 1 - a, 2 - c, 3 - b, 4 - a, 5 – a

**Graad 10**

**Les 1**: CSS-belyning

Lesdoelwitte

• Verstaan ​​hoe om elemente met CSS te belyn

• Leer verskillende belyningstegnieke vir teks- en blokelemente

• Pas belyningseienskappe toe om visueel aantreklike uitlegte te skep

Teksbelyning

Die teks-belyning-eienskap word gebruik om die horisontale belyning van teks binne 'n element te stel.

Sintaksis:

element {

teks-belyning: links | regs | middelpunt | regverdig;

}

• links: Belyn teks na links.

• regs: Belyn teks na regs.

• middelpunt: Sentreer die teks.

• regverdig: Rek die lyne sodat elke lyn gelyke breedte het (soos in koerante en tydskrifte).

Voorbeeld:

<p style="text-align: middelpunt;">Hierdie teks is gesentreer.</p>

Vertikale belyning

Vertikale belyning kan met behulp van verskeie metodes bereik word, afhangende van die konteks.

1. Gebruik van vertikaal-belyning vir inlyn-elemente

Die vertikaal-belyning-eienskap belyn inlyn- of inlyn-blok-elemente vertikaal.

Sintaksis:

element {

vertical-align: basislyn | bo | middel | onder;

}

Voorbeeld:

<span style="vertical-align: middle;">Belynde teks</span>

2. Gebruik van Flexbox vir vertikale belyning

Flexbox bied 'n kragtige manier om items horisontaal en vertikaal te belyn.

Voorbeeld:

<div style="display: flex; align-items: center; justify-content: center; height: 200px;">

<p>Vertikaal en horisontaal gesentreer</p>

</div>

• align-items: center; sentreer items vertikaal.

• justify-content: center; sentreer items horisontaal.

Sentreer blokelemente

Om 'n blokelement te sentreer (soos 'n <div>), kan jy marge auto gebruik.

Voorbeeld:

<div style="width: 50%; margin: 0 auto;">

<p>Hierdie div is horisontaal gesentreer.</p>

</div>

Oefeninge

1. Sentreer teks

o Skep 'n paragraaf en sentreer die teks met behulp van teks-align.

2. Vertikale belyning van inlyn-elemente

o Plaas 'n beeld en teks langs mekaar en belyn hulle vertikaal met behulp van vertikale-align.

3. Sentreer 'n div met behulp van Flexbox

o Skep 'n houer met 'n vaste hoogte en sentreer 'n kind-element beide vertikaal en horisontaal met behulp van Flexbox.

Uitdagingstaak

Ontwerp 'n kaartkomponent wat 'n beeld bo-aan het, gevolg deur 'n titel en beskrywing. Sentreer al die inhoud horisontaal en maak seker dat die kaart in die viewport beide vertikaal en horisontaal gesentreer is met behulp van Flexbox.

Opsomming

• teks-align word gebruik vir horisontale teksbelyning.

• vertikale-align belyn inlyn-elemente vertikaal.

• Flexbox bied 'n buigsame manier om elemente beide vertikaal en horisontaal te belyn.

• Sentrering van blokelemente kan bereik word met behulp van margin: 0 auto;.

Vasvrae

Watter eienskap word gebruik om teks in die middel te belyn?

a) vertikaal-belyn

b) teks-belyn

c) items-belyn

d) inhoud-regverdig

Wat doen margin: 0 auto; aan 'n blokelement?

a) Belyn dit links

b) Belyn dit regs

c) Sentreer dit horisontaal

d) Sentreer dit vertikaal

Watter Flexbox-eienskap sentreer items vertikaal?

a) inhoud-regverdig

b) items-belyn

c) teks-belyn

d) vertikaal-belyn

Kan vertikale-belyn gebruik word om blokelemente vertikaal te belyn?

a) Ja

b) Nee

Watter metode is die beste vir die sentrering van beide horisontaal en vertikaal?

a) Gebruik teks-belyn en vertikale-belyn

b) Gebruik margin: 0 auto;

c) Gebruik Flexbox met items-belyn en inhoud-regverdig

d) Gebruik posisie: absoluut;

Antwoordsleutel: 1 - b, 2 - c, 3 - b, 4 - b, 5 – c

**Les 2:** CSS-kombinators

Lesdoelwitte

• Verstaan ​​wat CSS-kombinators is en hoe hulle verhoudings tussen selekteerders definieer.

• Leer die vier primêre CSS-kombinators: afstammeling, kind, aangrensende broer/suster en algemene broer/suster.

• Pas kombinators toe om presiese en doeltreffende CSS-selekteerders te skep.

Wat is CSS-kombinators?

CSS-kombinators is simbole wat verhoudings tussen selekteerders definieer, wat jou toelaat om elemente te teiken gebaseer op hul posisie relatief tot ander elemente in die HTML-struktuur.

Tipes CSS-kombinators

1. Afstammeling-kombinator (spasie)

Selekteer alle elemente wat afstammelinge (kinders, kleinkinders, ens.) van 'n gespesifiseerde element is.

Sintaksis:

ouer afstammeling {

/\* style \*/

}

Voorbeeld:

div p {

kleur: blou;

}

Dit selekteer alle <p>-elemente binne <div>-elemente, ongeag hoe diep geneste hulle is.

2. Kind-kombinator ( > )

Selekteer alle elemente wat direkte kinders van 'n gespesifiseerde element is.

Sintaksis:

ouer > kind {

/\* style \*/

}

Voorbeeld:

ul > li {

lys-styl-tipe: vierkant;

}

Dit kies alle <li> elemente wat direkte kinders van 'n <ul> element is.

3. Aangrensende Sibling Kombinator ( + )

Kies die element wat direk na 'n gespesifiseerde element is, en beide deel dieselfde ouer.

Sintaksis:

element1 + element2 {

/\* style \*/

}

Voorbeeld:

h2 + p {

lettertipe-styl: kursief;

}

Dit kies die eerste <p> element wat direk op 'n <h2> element volg.

4. Algemene Sibling Kombinator ( ~ )

Kies alle elemente wat sibbe van 'n gespesifiseerde element is en daarna kom.

Sintaksis:

element1 ~ element2 {

/\* style \*/

}

Voorbeeld:

h2 ~ p {

kleur: grys;

}

Dit kies alle <p>-elemente wat broers en susters van 'n <h2>-element is en daarna verskyn.

Oefeninge

1. Oefening vir Afstammelinge-selektor

o Skep 'n <div> wat veelvuldige geneste elemente bevat, insluitend <p>-etikette. Gebruik die afstammelinge-kombinator om alle <p>-elemente binne die <div> te stileer.

2. Oefening vir Kinderselektor

o Bou 'n navigasiekieslys met behulp van <ul> en <li>. Gebruik die kinderkombinator om slegs die direkte <li>-kinders van die <ul> te stileer.

3. Oefening vir Aangrensende Sibling-selektor

o Ontwerp 'n afdeling met opskrifte en paragrawe. Gebruik die aangrensende sibling-kombinator om die eerste <p> te stileer wat direk na elke <h2> kom.

4. Oefening vir Algemene Sibling-selektor

o Skep 'n reeks opskrifte en paragrawe. Gebruik die algemene sibling-kombinator om alle <p>-elemente wat op 'n <h2> volg, te stileer.

Uitdagingstaak

Ontwerp 'n blogplasing-uitleg waar:

• Elke <h2>-opskrif gevolg word deur veelvuldige <p>-paragrawe.

• Die eerste <p> na elke <h2> word anders gestileer (bv. vetdruk of kursief).

• Alle daaropvolgende <p> elemente na die eerste word met 'n ander kleur gestileer.

Gebruik 'n kombinasie van aangrensende en algemene sibling-kombinators om hierdie uitleg te bereik.

Opsomming

• CSS-kombinators definieer verhoudings tussen selektors, wat presiese teikenstelling van elemente moontlik maak.

• Die vier hoofkombinators is:

• Afstammeling (spasie): Kies alle afstammelinge van 'n element.

• Kind ( > ): Kies direkte kinders van 'n element.

• Aangrensende Sibling ( + ): Kies die volgende sibling-element.

Algemene Sibling ( ~ ): Kies alle sibling-elemente wat volg.

Vasvrae

1. Watter kombinator kies alle elemente wat afstammelinge van 'n spesifieke element is?

a) Kind ( > )

b) ​​Afstammeling (spasie)

c) Aangrensende Broer/Suster ( + )

d) Algemene Broer/Suster ( ~ )

2. Wat kies die kindkombinator ( > )?

a) Alle afstammelingelemente

b) Slegs direkte kindelemente

c) Alle broer/susterelemente

d) Elemente met dieselfde klas

3. Watter kombinator sou jy gebruik om die eerste <p>-element onmiddellik na 'n <h2> te kies?

a) Afstammeling (spasie)

b) Kind ( > )

c) Aangrensende Broer/Suster ( + )

d) Algemene Broer/Suster ( ~ )

4. Wat is die verskil tussen die aangrensende broer/suster ( + ) en algemene broer/suster ( ~ ) kombinators?

a) Geen verskil nie; hulle funksioneer dieselfde.

b) Aangrensende broer/suster kies slegs die volgende broer/suster; algemene broer/suster kies alle volgende broer/suster.

c) Aangrensende broer/suster kies alle vorige broer/suster; algemene broer/suster kies alle volgende broer/suster.

d) Aangrensende broer/suster kies alle broer/suster; 'n Algemene sibbe kies slegs die volgende sibbe.

5. Kan kombinators gekombineer word om meer spesifieke selekteerders te skep?

a) Ja

b) Nee

Antwoordsleutel: 1 - b, 2 - b, 3 - c, 4 - b, 5 – a

**Les 3:** CSS Pseudo-klasse

Lesdoelwitte

• Verstaan ​​wat CSS pseudo-klasse is en hoe hulle funksioneer.

• Leer algemene pseudo-klasse en hul toepassings.

• Pas pseudo-klasse toe om interaktiwiteit en stilering in webblaaie te verbeter.

Wat is CSS Pseudo-klasse?

'n CSS pseudo-klas is 'n sleutelwoord wat by 'n selektor gevoeg word wat 'n spesiale toestand van die gekose element(e) spesifiseer. Pseudo-klasse laat jou toe om style toe te pas gebaseer op gebruikersinteraksie of die element se posisie in die dokumentboom. citeturn0search0

Sintaksis:

selector:pseudo-class {

property: value;

}

Algemene CSS Pseudo-klasse

1. :hover

Word toegepas wanneer die gebruiker 'n element aanwys (met 'n wysertoestel), maar dit nie aktiveer nie. Byvoorbeeld, wanneer 'n gebruiker oor 'n knoppie beweeg.

Voorbeeld:

button:hover {

background-color: lightblue;

}

2. :active

Word toegepas terwyl 'n element deur die gebruiker geaktiveer word. Byvoorbeeld, tussen die tye wat die gebruiker die muisknoppie druk en dit loslaat.

Voorbeeld:

a:active {

color: red;

}

3. :focus

Word toegepas wanneer 'n element fokus ontvang het, óf deurdat die gebruiker dit met die muis kies óf deur daarna te navigeer met behulp van die sleutelbord.

Voorbeeld:

input:focus {

border-color: green;

}

4. :first-child

Word toegepas op 'n element wat die eerste kind van sy ouer is.

Voorbeeld:

p:first-child {

font-weight: bold;

}

5. :nth-child(n)

Van toepassing op die nde kind van 'n ouer-element.

Voorbeeld:

li:nth-child(2) {

color: blue;

}

6. :not(selector)

Van toepassing op elke element wat nie ooreenstem met die gegewe selektor nie.

Voorbeeld:

p:not(.intro) {

color: grys;

}

Oefeninge

1. Sweef-effek

o Skep 'n knoppie wat sy agtergrondkleur verander wanneer dit oor die sweef gehou word met behulp van die :hover pseudo-klas.

2. Fokus Stilering

o Ontwerp 'n invoerveld wat sy randkleur verander wanneer dit gefokus word met behulp van die :focus pseudo-klas.

3. Eerste Kind Stilering

o Stileer die eerste paragraaf binne 'n <div> om vet teks te hê met behulp van die :first-child pseudo-klas.

4. Nde Kind Stilering

o Verander die tekskleur van elke tweede item in 'n lys van items met behulp van die :nth-child(2n) pseudo-klas.

Uitdagingstaak

Ontwerp 'n navigasiekieslys waar:

• Elke skakel van kleur verander wanneer dit oorgehou word.

• Die aktiewe skakel (die een wat tans gekies is) 'n ander agtergrondkleur het.

• Die eerste skakel het vetgedrukte teks.

Gebruik gepaste pseudo-klasse om hierdie stilering te bereik.

Opsomming

• CSS pseudo-klasse laat jou toe om style toe te pas gebaseer op die status van 'n element.

• Algemene pseudo-klasse sluit in: hover, :active, :focus, :first-child, :nth-child(n), en :not().

• Hulle verbeter interaktiwiteit en bied meer beheer oor elementstilering sonder bykomende klasse of JavaScript.

Kwis

1. Wat doen die :hover pseudo-klas?

a) Stileer 'n element wanneer daarop geklik word.

b) Stileer 'n element wanneer die muiswyser daaroor is.

c) Stileer die eerste kind van 'n element.

d) Stileer 'n element wanneer dit fokus kry.

2. Watter pseudo-klas teiken die eerste kind van 'n ouer-element?

a) :first-child

b) :nth-child(1)

c) :first

d) :child(1)

3. Hoe sou jy elke derde lysitem in 'n ongeordende lys stileer?

a) li:nth-child(3)

b) li:nth-child(3n)

c) li:nth-of-type(3)

d) li:nth(3)

4. Wat doen die :not() pseudo-klas?

a) Kies elemente wat ooreenstem met die selektor binne :not().

b) Kies elemente wat nie ooreenstem met die selektor binne :not().

c) Kies alle elemente.

d) Kies geen elemente nie.

5. Watter pseudo-klas is van toepassing wanneer 'n invoerveld deur die gebruiker gekies word?

a) :sweef

b) :aktief

c) :fokus

d) :gekontroleer

Antwoordsleutel: 1 - b, 2 - a, 3 - b, 4 - b, 5 - c