

2.1 Layout

Vi ska nu i alla uppgifter 2.* stilsätta den webbplats som man har gjort i uppgifterna 1.*.

Uppgiften introducerar till layout med CSS och eftersom användare har olika enheter som:

- desktop
- plattor
- mobiler

som har olika storlekar på skärmarna så måste layouten vara dynamisk och anpassa sig efter dessa för att vara användarvänlig. Denna dynamiska anpassning kallas för **responsiv layout**.

Uppgift

När man positionerar ut saker på en HTML-sida med CSS så kan man antingen göra layouten:

- Fixerad: Man använder `px` som måttenhet och allt får samma storlek oberoende av webbläsarfönstrets storlek
- Flödande: Man använder `%` som måttenhet och allt får olika storlek beroende av webbläsarfönstrets storlek

Vi ska använda en kombination av båda i denna uppgift. Designen ska vara flödande så att den följer med fönstrets storlek inom intervaller men vara fixerad utanför vissa gränser:

1. Desktop:

- Fönsterbredd över ett visst värde (exempelvis `1200px`): designen ska vara fixerad
- Fönsterbredd mellan: designen ska vara flödande
- Fönsterbredd under ett visst värde (exempelvis `1024px`): designen ska gå över till en annan stil anpassat för plattor

2. Plattor:

- Fönsterbredd över ett visst värde (exempelvis `1024px`): designen ska gå över till en annan stil anpassat för desktop
- Fönsterbredd mellan: designen ska vara flödande
- Fönsterbredd under ett visst värde (exempelvis `480px`): designen ska gå över till en annan stil anpassat för mobiler

3. Mobiler:

- Fönsterbredd över ett visst värde (exempelvis `480px`): designen ska gå över till en annan stil anpassat för plattor
- Fönsterbredd mellan: designen ska vara flödande
- Fönsterbredd under ett visst värde (exempelvis `100px`): designen ska vara fixerad

Skrivare ska ha en egen stilsättning som är enkel och tydlig (exempelvis genom att inte stilsätta dokumentet alls i detta fall) så att den passar för utskrift.

Notera att:

- Delarna i huvudstrukturen i uppgift [1.3 Huvudstruktur](#) ska positioneras ut på ett passande sätt för varje enhet (olika sätt för varje enhet)
- För större enheter ska man använda flera kolumner för texten så raderna inte blir för breda (och därmed svårlästa) och för allt mindre enheter ska allt färre kolumner för texten användas (för att vara läsbart), för mobiler endast en kolumn

Stilsätt alla HTML-dokument (utom `index.html`) så att de får en responsiv layout enligt ovan. Varje CSS-dokument ska följa kraven under sidan [Krav](#).

Exempel

Följande fyra exempel är minimala och enbart stilsatta för att visa en viss funktion. De har fula färger för att visa olika områden och så vidare och det är inte tänkt att ni ska stilsätta er webbplats så här fult :). Studera koden för att se hur enkelt det fungerar. Testa exemplen genom att förändra bredden på webbläsaren. Notera att länken till den externa CSS-filen har följande format:

```
<link rel="stylesheet" type="text/css" href="test1.css?rnd=12345"/>
```

Den extra texten `?rnd=12345` finns eftersom vissa webblärare (som exempelvis Safari) cashar och använder tidigare CSS-filer med ett visst namn och genom att ändra dessa siffror tvingar man webbläsaren att ladda in en ny fil.

Följande är ett exempel på hur en grundstruktur (utan innehåll) utan responsiv layout kan se ut för desktop:

- [HTML](#)
- [CSS](#)

Följande är ett exempel på responsiv layout (där enbart en färg ändras) för olika fönsterbredder:

- [HTML](#)
- [CSS](#)

Följande är ett exempel på hur en grundstruktur (utan innehåll) med responsiv layout kan se ut för mobil, plattor och desktop (här har gränsen satts till 600px för mobiler för att kunna synas i en "smal" webbläsare på desktop):

- [HTML](#)
- [CSS](#)

Senaste exemplet utbyggt med:

- `I header`: Variabel storlek på texten "Header"
- `I section 1`: Texten placerad i olika antal kolumner för ökad tydlighet (bred text är svår att läsa)
- `I section 2`: Bilden har tre olika bildkällor
- [HTML](#)
- [CSS](#)

Följande är ett exempel på responsiv layout (där enbart en färg ändras) för olika behållarbredder:

- [HTML](#)
- [CSS](#)

Detta behöver bara användas i avancerade layouter, se:

- [YouTube: Container Queries are going to change how we make layouts»](#)
- [Developer Mozilla: CSS Container Queries»](#)
- [IShadeed: Say Hello To CSS Container Queries»](#)

Tips

Genom att använda samma CSS-dokument för hela webbplatsen så kan man enkelt uppdatera och förändra designen för alla HTML-sidor på ett enda ställe.

Använd `display` med `grid`:

- `grid`, `grid-auto-rows`, `grid-auto-columns`, `grid-auto-flow`
- `grid-template`, `grid-template-rows`, `grid-template-columns`, `grid-template-areas`

- grid-area, (grid-row, grid-row-start, grid-row-end), (grid-column, grid-column-start, grid-column-end)
- grid-gap, grid-row-gap, grid-column-gap
- gap, row-gap, column-gap

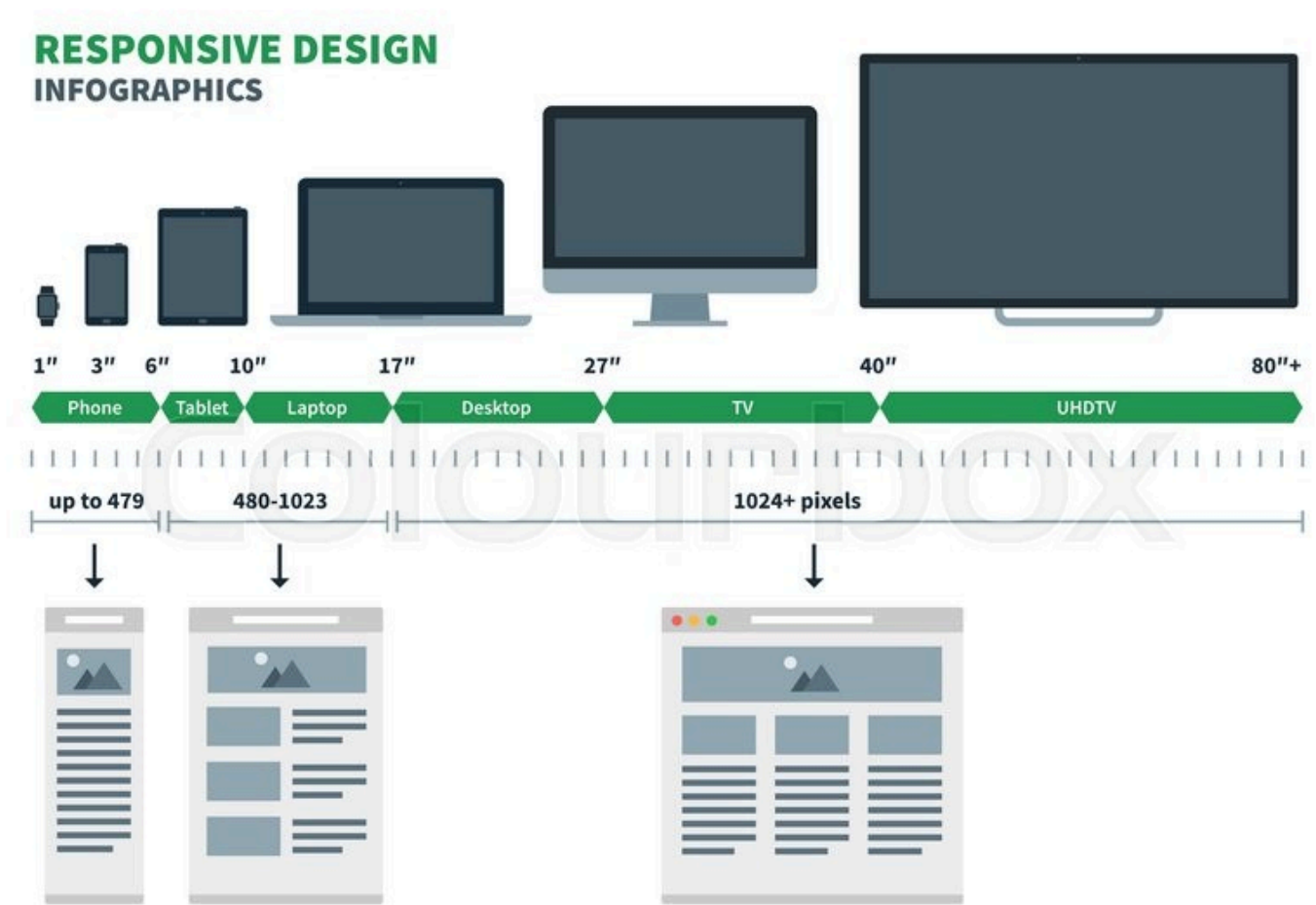
Se följande tips:

- [CSS-Tricks: A Complete Guide to CSS Grid»](#)
- [CSS-Tricks: An Introduction to the `fr` CSS unit»](#)
- [Grid by Example: Get Started with Grid Layout»](#)
- [Youtube: CSS Grid Changes EVERYTHING Amazing Presentation»](#)
- [Youtube: CSS Grid Layout Crash Course»](#)
- [Youtube: CSS Grid Tutorial | Responsive Crash Course»](#)

Använd "media query":

- @media

Följande bild visar gränserna för responsiv layout för olika enheter:



Se även:

- [W3Schools: Responsive Web Design: Media Queries»](#)
- [W3Schools: CSS Media Queries: Examples»](#)

Designa för mobil enhet först och lägg till speciell layout för plattor och desktop efter som specialfall. Då fungerar webbplatsen även för äldre webbläsare som inte stöder CSS Grid.

Man behöver möjligtvis även använda ett urval av följande egenskaper:

- **Position:** position, top, left bottom, right, z-index, float, clear

- **Storlek:** width, height, max-width, max-height, min-width, min-height, vertical-align
- **Marginaler, kantlinjer och paddning:**
 - **Marginaler:** margin, margin-top, margin-right, margin-bottom, margin-left
 - **Kantlinjer:**
 - border, border-top, border-right, border-bottom, border-left
 - border-color, border-top-color, border-right-color, border-bottom-color, border-left-color
 - border-style, border-top-style, border-right-style, border-bottom-style, border-left-style
 - border-width, border-top-width, border-right-width, border-bottom-width, border-left-width
 - **Paddning:** padding, padding-top, padding-right, padding-bottom, padding-left
 - **Outlines:** outline, outline-color, outline-style, outline-width
- **Synlighet:** visibility, clip-path, display, overflow

Man kan även titta på hur man kan få en box:

- Som är transparent till en viss grad (se `rgba` och `opacity`)
- Som har rundade hörn (se `border-radius`)
- Som har multippla bakgrundsbilder
- Som har skuggor (se `box-shadow`)
- Som har gradienter (se `linear-gradient` och `radial-gradient`)

Om man använder `em` och `rem` som måttenheter (som beror på fontstorleken) så får allt olika storlek proportionerligt beroende på fontstorleken:

```
.article {
  width: 5em;
  height: 2em;
  border-radius: .5em;
  border: 1px solid gold;
}
```

Sen kan man enkelt ändra storleken proportionerligt på hela konstruktionen genom att ändra fontstorleken:

```
font-size: 95%;
```

Se vidare:

- [Simurai: Sizing \(Web\) components»](#)
- [Sitepoint: The Power of em Units in CSS»](#)

Man får text att flöda mellan flera kolumner med `column-count` och `column-width`.

Använd:

```
<meta name="viewport" content="width=device-width, initial-scale=1.0">
```

För utskrift kan man (frivilligt) titta på de speciella CSS-konstruktioner som finns för skrivare:

- Sidbaserad regel: `@page`
- Sidbaserade egenskaper: `marks`, `page-brake-after`, `page-brake-before`, `page-break-inside`, `size`, `orphans`, `widows`

Man kan testa sin design för skrivare genom att "skriva ut till PDF" och sen se hur den resulterande PDF-filen ser ut.

Hjälp

Om man fastnar ordentligt så kan man få hjälp via epost:

[Klicka här för hjälp](#)

Om man ska sända med kod:

- Sänd endast med de delar som är relevanta för frågan
- Koden ska vara välkommenterad och välstrukturerad

Det är **ingen** handledning på kunskap som ingår i förkunskapskraven (kompileringsfel med mera). Epost sänds beroende på ämne till en specifik handledare som försöker svara snabbt men eftersom det ibland är väldigt många frågor så kan svaren dröja. Handledarresurserna fördelas jämt över studenterna så om man ställer många frågor så får man vänta längre tid på svar. Notera att handledarna enbart arbetar inom angivna handledningstider.

Den här sidan fungerar på både stationära och mobila klienter, har hög tillgänglighet för funktionshindrade och använder webbsäkra fonter. Den följer W3C:s strikta standard HTML5 och CSS 3. Den är genererad från textfiler som använder ett egendefinerat minimalt WIKI-baserat språk och en databas. Systemet och allt material är gjort av [Pierre A. I. Wijkman](#)» och är skyddat av lagen om upphovsrätt.