

Genom att använda mig av Grid och lite av flex, så har majoriteten av anpassningen för olika storlekar lösts automatiskt.

Men jag har behövt använda lite @mediaqueries för att få sidan/sidorna att se ut som jag vill och för den att bli komplett anpassad för mobil, desktop och tablet.

1. Jag ville ändra min Nav till att ha hamburger-format när jag går under 1040px i max-width.

Detta sker då i min main.css fil som alla HTML-sidor är länkade till.

Min Nav är från början en flex-box och ser originellt ut såhär:

```
nav {  
  grid-area: nav;  
  display: flex;  
  gap: 20px;  
  justify-content: space-evenly;  
}
```



Med @mediaquery på 1040px max-width så ändrar jag min nav till en grid och ger varje knapp en row på 1fr storlek. Vilket ger hamburger-format enligt nedan:

```
@media (max-width: 1040px) {  
  nav {  
    display: grid;  
    grid-template-rows: repeat(3, 1fr);  
    gap: 1px;  
    justify-content: stretch;  
  }  
}
```



2. Vid 1430px max-width ska big-container i min grid separera sig från boxes och de ska hamna på var sin rad. Jag placerar också dessa i mitten av skärmen för att det ska se bra ut. Det går då från:

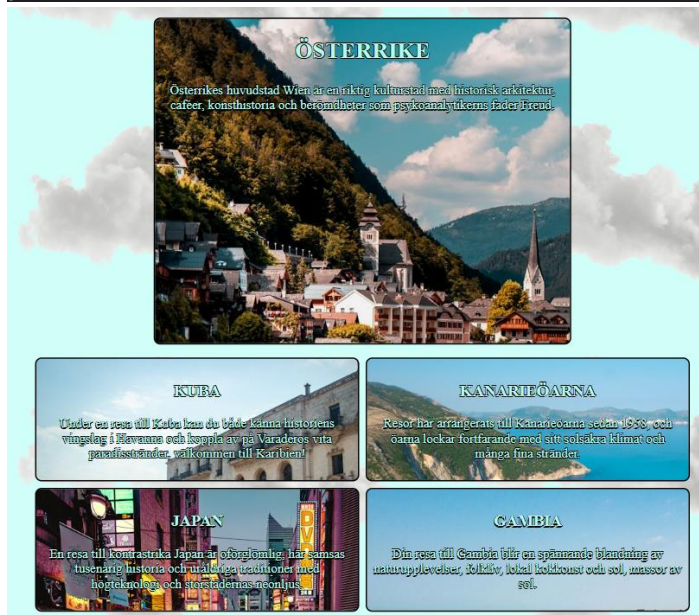
```
.grid-container {
  display: grid;
  grid-template-areas:
    "header header header header header header"
    "nav nav nav nav nav nav"
    "big-container big-container big-container boxes boxes boxes"
    "footer footer footer footer footer footer";
  gap: 20px;
  padding: 50px;
  border-radius: 10px;

  color: rgba(190, 255, 240, 0.98);
}
```



Till att hamna under varandra:

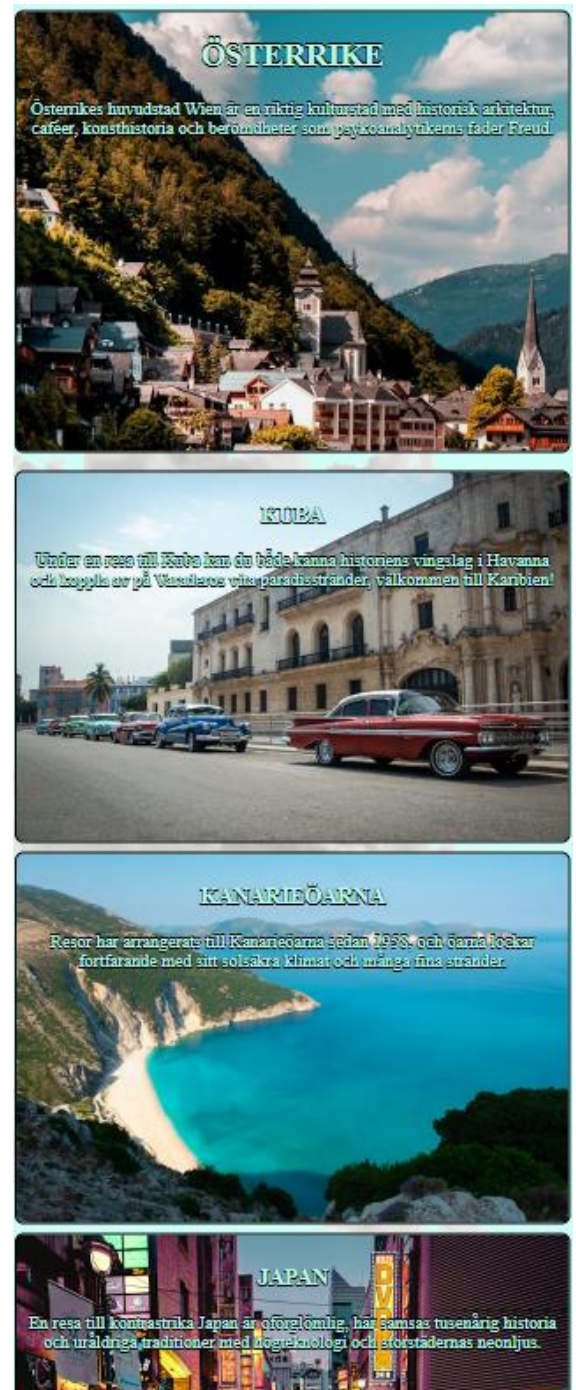
```
@media (max-width: 1430px){
  .grid-container{
    grid-template-areas:
      "header header header header header header"
      "nav nav nav nav nav nav"
      "big-container big-container big-container big-container big-container big-container"
      "boxes boxes boxes boxes boxes boxes"
      "footer footer footer footer footer footer";
  }
  .big-container{
    justify-self: center;
    align-self: center;
  }
  .boxes {
    justify-self: center;
  }
}
```



3. Vid 767px max-width så är sidan anpassad att .boxes går från att vara flera på samma rad, till att de ska komma under varandra. För att göra det så ändrar jag hur .boxes grid ser ut. Från att de ska vara två i varje kolumn till att var ska sin egna rad. Jag ger också att varje .box ska ha en minimum höjd.

```
.boxes {  
  grid-area: boxes;  
  display: grid;  
  grid-template-columns: 1fr 1fr;  
  gap: 10px;  
}
```

```
@media(max-width: 767px) {  
  .boxes {  
    grid-template-columns: none;  
    grid-template-rows: repeat(4, 1fr);  
    justify-content: stretch;  
  }  
  .box {  
    min-height: 400px;  
  }  
}
```



4. Vid max-width 767px på mina innersidor för de olika destinationerna så anpassas också font-storleken för artiklarna. Detta så att texten faktiskt passar där den ska vara och man ska slippa skrolla på själva artiklarna. Testade det först, men tyckte inte det passade så bra på mobil.

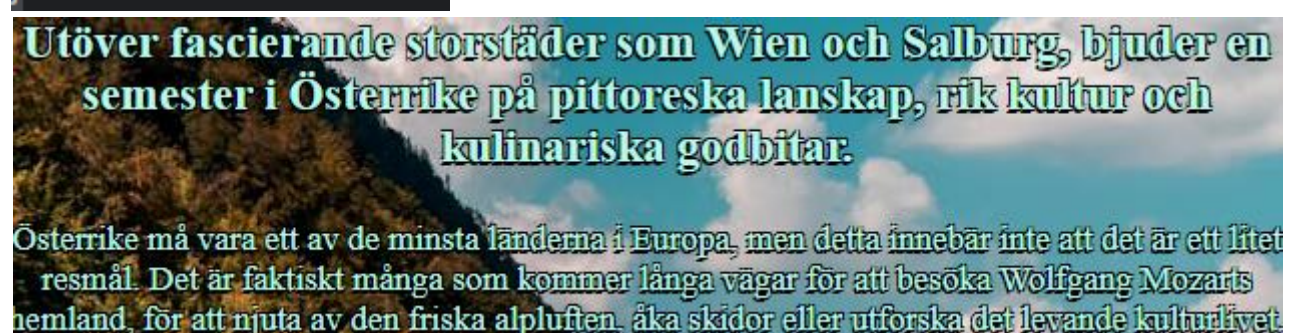
Från:

```
h2 {  
  font-size: 35px;  
}  
  
h3 {  
  font-size: 25px;  
}  
  
p {  
  font-size: 20px;  
}
```



Till:

```
@media (max-width: 767px) {  
  h2 {  
    font-size: 25px;  
  }  
  h3 {  
    font-size: 23px;  
  }  
  p {  
    font-size: 17px;  
  }  
}
```



5. För de minsta skärmarna på max-width 500px så anpassar jag också marginalerna för min grid-container för att kunna utnyttja större del av skärmen. Detta är gjort på både huvudsidan, men också alla innersidor.

```
.grid-container {  
  display: grid;  
  grid-template-areas:  
    "header header"  
    "nav nav nav"  
    "big-container"  
    "footer footer"  
  gap: 20px;  
  padding: 50px;  
  border-radius: 10px;  
  color: rgba(190, 255,  
}
```

```
@media (max-width: 500px) {  
  .grid-container {  
    padding: 10px;  
  }  
}
```

