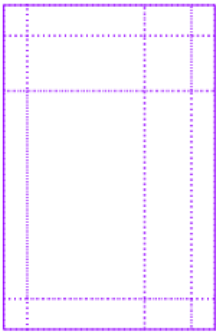


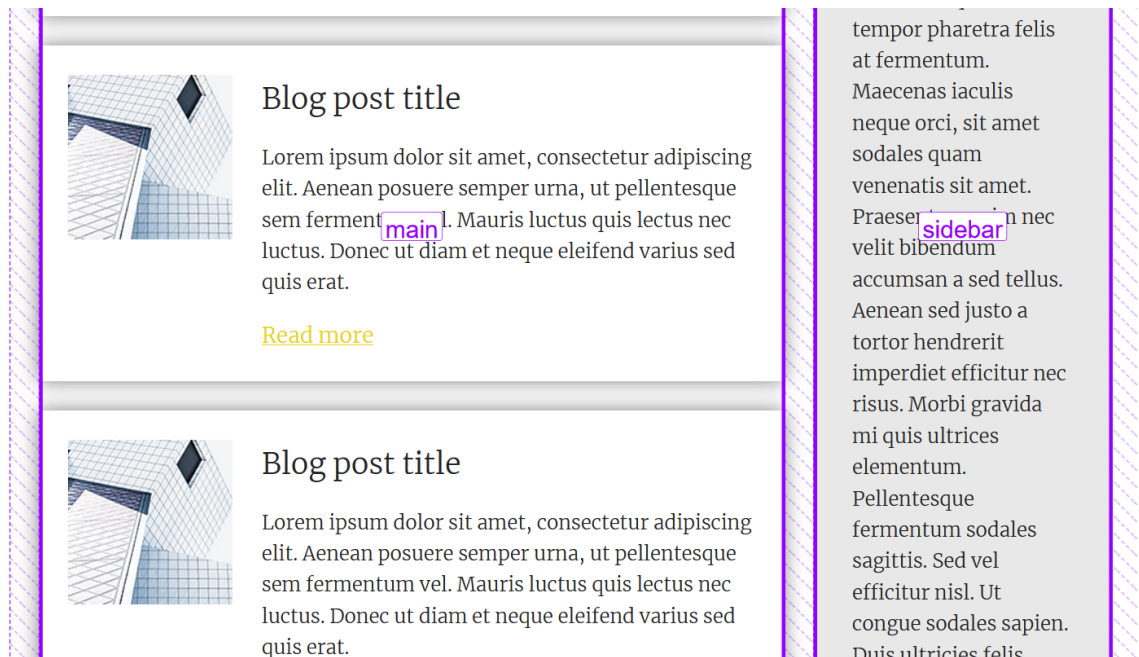
GRID-TEMPLATE-AREAS

En el body

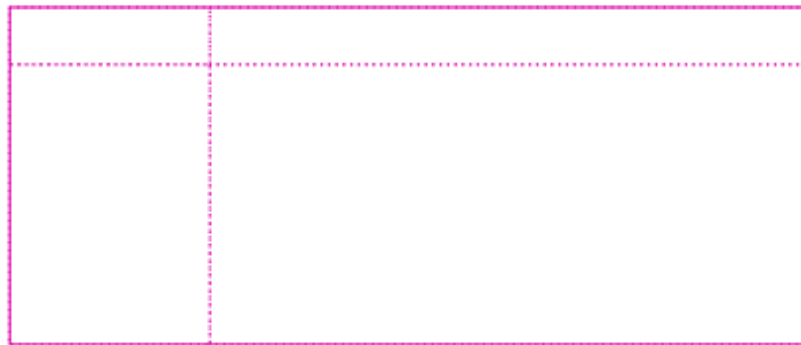
- ☒ body  
- ☐ div.card  
- ☐ div.card  
- ☐ div.card  

- ☒ Mostrar números de línea
- ☒ Mostrar nombres de áreas
- ☐ Extender líneas infinitamente





Cada card es también un grid con grid-template-areas



COSAS A TENER EN CUENTA

Tamaño de la letra en header, footer, hero debe ser responsive: recuerda el ejercicio de flex-box donde usábamos:

- Vw
- Calc (1rem + 0,6667vw)

Header, footer y hero son contenedor flex para facilitar centrar sus elementos

La fuente usada en header, footer y hero es diferente a la del resto del documento.

Grid del body, hay que definir

Grid-template-rows:

- 4 filas, En móviles/tablets son 5
- Header y footer tienen el mismo alto: aprox. 10rem
- La zona central (main y sidebar) tiene que ser auto
- Hero ocupa la mitad de la vista

Grid-template-columns

- Son 4
- La primera y la última son iguales y las expresamos con fr
- La segunda (la más ancha y el doble de la tercera) ocupa un mínimo de 30rem y un máximo de 5fr
- La tercera ocupará, por tanto, un mínimo de 15rem y máximo de 2rem

Grid-template-areas

Aquí definimos el mapa de mi grid. Recuerda que header, hero y footer ocupan todo el ancho, pero que la parte central (main y sidebar) NO, la primera y última columna no tienen nada.

Recuerda que cada item del grid tiene que tener especificado que zona le corresponde del mapa, ejemplo:

```
header {  
  grid-area: header;  
}
```

Grid del card

El elemento card es también un grid.

En este caso es de dos filas y dos columnas para colocar la imagen, título y contenido del card.

Hay que seguir los mismos pasos que en el body. Pero no te doy más pistas, pero ten en cuenta que el hueco para la imagen es más pequeño que el que ocupa el resto (título y contenido)