

**Bloggie-Blog-> A blog developed with HTML, CSS and JavaScript with frontend development skills, including various features of header, search bar, featured articles, posts, popular tags, newsletter and media queries, and so on.**

#### **Functionalities/Demo:**

- A responsive blog website from scratch.
- A header section that holds the navigation and all of its items.
- A theme toggle button that transitions the color scheme into the light scheme.
- It uses JavaScript Store, so the theme selection remains active when closing the webpage.
- A search button with a popping up activated search bar when clicking the button.
- Signin and Signup buttons with neon effect.
- A main section with a nested grid that is split into two sides with creative layouts.
- A right-hand-side sidebar contains a quick read section utilizing JavaScript library **Swiper**, which creates a responsive carousel, to achieve arrows in pagination work.
- Older Posts section and Popular Tags section that are composed of six cards.
- A Newsletter section prompting users to subscribe to the newsletter if they want to.

#### **Developing Tools:**

JavaScript Swiper <https://swiperjs.com/>

Vscode 1.72 [https://code.visualstudio.com/updates/v1\\_72](https://code.visualstudio.com/updates/v1_72)

Live Server Vscode Extension v5.7.9 <https://www.vsixhub.com/vsix/1950/>

#### **Prerequisites & Setup:**

##### Set up Header:

In index.html under Starter:

Make Header contains a navigation bar contains a menu of lists and their list-items:

```
<!-- Header -->
<header class = "header" id = "header">
  <nav class = "navbar container">
    <a href = "./index.html">
      <h2 class = "logo">BloggieBlog</h2>
    </a>
    <div class = "menu" id = "menu">
      <ul class = "list">
        <li class="list-item">
          <a href = "#" class = "list-link
current">Home</a>
        </li>
        <li class="list-item">
          <a href = "#" class =
"list-link">Categories</a>
        </li>
        <li class="list-item">
          <a href = "#" class = "list-link">Reviews</a>
        </li>
```

```

        <li class="list-item">
            <a href = "#" class = "list-link">News</a>
        </li>
        <li class="list-item">
            <a href = "#" class =
"list-link">Membership</a>
        </li>
        <li class="list-item">
            <a href = "#" class = "list-link">Contact</a>
        </li>
        <li class="list-item">
            <a href = "#" class = "list-link">Sign in</a>
        </li>
        <li class="list-item">
            <a href = "#" class = "list-link">Sign up</a>
        </li>
    </ul>
</div>
</nav>
</header>

```

Click Go Live, and the 8 list-items shows on localhost as:

### 8 list-items.PNG

Adding a base button as sun and moon icons underneath the list items:

```

<div class = "list list-right">
    <button class = "btn place-items-center" id = "theme-toggle-btn">
        <i class="ri-sun-line sun-icon"> </i>
        <i class="ri-moon-line moon-icon"> </i>
    </button>

```

### sun and moon icons.PNG

Adding a search button and a closed-menu button:

```

    <button class = "btn place-items-center" id = "search-icon">
        <i class="ri-search-line"> </i>
    </button>

    <button class = "btn place-items-center screen-lg-hidden
menu-toggle-icon" id = "menu-toggle-icon">
        <i class="ri-menu-3-line open-menu-icon"> </i>
        <i class="ri-close-line close close-menu-icon"> </i>
    </button>

```

### search button and closed-menu button.PNG

Hide specific signin and signup items into small screen:

```
<a href = "#" class = "list-link screen-sm-hidden">Sign in</a>
<a href = "#" class = "list-link screen-sm-hidden btn sign-up-btn
fancy-border">
  <span>Sign up</span></a>
```

### **Synchronous Developing Notes:**

#### **Base - CSS:**

In main.css under Starter:

For the base styles:

```
*{
  margin: 0;
  padding: 0;
  box-sizing: border-box;
}
html{
  /* font-size: 10px;*/
  font-size: 62.5%;
}
body{
  font-family: var( --font-family);
  font-size: var(--font-size-sm);
  color: var(--light-color-alt);
  background-color: var(--primary-background-color);
  letter-spacing: 1px;
  transition: background-color .25s, color .25s;
}
a{
  text-decoration: none;
  color: inherit;
}
img{
  max-width: 100%;
  display: block;
}
ul{
  list-style: none;
}
input,
button{
  font: inherit;
  color: inherit;
  border: none;
```

```

        background-color: transparent;
        outline: none;
    }
    i{
        font-size: var(--font-size-md);
    }

```

For the theme color:

```

body.light-theme{
    --light-color: #3d3d3d;
    --light-color-alt: rgba(0,0,0,.6);
    --primary-background-color: #fff;
    --secondary-background-color: #f1f1f1;
    --hover-light-color: #fff;
    --transparent-light-color: #252830;
    --transparent-dark-color: rgba(0,0,0,.1); }

```

For Reusable classes:

```

.container{
    max-width: 160rem;
    margin: 0 auto;
    padding: 0 1.5rem;
}

.place-items-center{
    display: inline-flex;
    align-items: center;
    justify-content: center;
}

```

And set the items as hidden into a small screen at the end:

```

.screen-sm-hidden{
    display: none; }

```

**after setting up CSS Base.PNG**

Header - CSS:

```

.header{
    background-color: var(--secondary-background-color);
    position: fixed;
    top: 0;
    left: 0;
    width: 100%;
    z-index: 999;
}

.navbar{
    display: flex;
    align-items: center;
}

```

```
        justify-content: space-between;
        padding-block: 1.5rem;
    }
    .logo{
        font-size: var(--font-size-md);
        color: var(--light-color);
    }
    .menu{
        position: absolute;
        top: 8.5rem;
        right: 1.5rem;
        width: 23rem;
        padding: 1.5rem;
        background-color: var(--secondary-background-color);
        /*opacity: 0;
        transform: scale(0);*/
        transition: opacity .25s ease-in;
    }
    .list{
        display: flex;
        align-items: center;
        gap: var(--gap);
    }
    .menu > .list{
        flex-direction: column;
    }
    .list-link.current{
        color: var(--light-color);
    }
    .close-menu-icon{
        display: none;
    }
    .btn{
        cursor: pointer;
    }
    .list-link:hover,
    .btn:hover,
    .btn:hover span{
        color: var(--light-color);
    }
    .moon-icon{
        display: none;
    }
```

```

}
.light-theme .sun-icon{
  display: none;
}
.light-theme .moon-icon{
  display: block; }

```

### Header JavaScript Styles:

Have the header and the menu activated:

```

.header.activated{
  box-shadow: 0 1px .5rem var( --transparent-light-color);}
.menu.activated{
  box-shadow: 1px 1px 1rem var( --transparent-light-color);
  opacity: 1;
  transform: scale(1); }

```

Switch the icons:

When it's activated, target it to the open menu icon:

```

.menu-toggle-icon.activated .open-menu-icon{
  display: none; }
.menu-toggle-icon.activated .close-menu-icon{
  display: block; }

```

### JavaScript Events Management:

Go to main.js.

To Grab Element:

Select element, if sth goes wrong, throw an error and specify the selector:

```

const selectElement = selector => {
  const element = document.querySelector(selector)
  if(element) return element;
  throw new Error('Something went wrong, make sure that ${selector}
exists or is typed correctly.');
```

Call it and pass the navigation bar to test if above function works:

```

console.log(selectElement('.navbar'));
```

Navigate the styles on scroll:

```

const scrollHeader = () => {
  const headerElement = selectElement('#header');
  if(this.scrollY >= 15){
    headerElement.classList.add('activated');
  } else{
    headerElement.classList.remove('activated');
  }
};

```

Make Scroll header visible by calling it in add event listener:

```

window.addEventListener('scroll', scrollHeader);
```

Test it by setting the transparent-light-color to red in main.css:

```
.header.activated{
  box-shadow: 0 1px .5rem var( --transparent-light-color);
  background-color: red;
}
```

Also add Search division in index.html:

```
<div class = "search"></div>
```

So when we scroll down the webpage, the navigation bar turns to red:

**nav bar turns into red.PNG**

Open menu & search pop-up:

```
const menuToggleIcon = selectElement('#menu-toggle-icon');
const toggleMenu = () => {
  const mobileMenu = selectElement('#menu');
  mobileMenu.classList.toggle('activated');
  menuToggleIcon.classList.toggle('activated');
};
```

Then call the menu toggle icon with add event listener:

```
menuToggleIcon.addEventListener('click', toggleMenu);
```

Error: Menu item not showing when clicking the menu toggle icon. DEBUGGING: Change opacity from 0 to 1 to make the menu list display.

Now click the menu toggle icon on the right top bar we got:

**menu toggle list items showing.PNG**

Switch theme/add to local storage:

```
const bodyElement = document.body;
const themeToggleBtn = selectElement(`#theme-toggle-icon`);
```

Add the click operation to switch light/dark themes:

```
themeToggleBtn.addEventListener('click', () => {
  bodyElement.classList.toggle('light-theme'); ...});
```

To store the selected theme whenever refresh webpage by adding localStorage:

```
if(bodyElement.classList.contains('light-theme')){
  localStorage.setItem('currentTheme', 'themeActive');
}else{
  localStorage.removeItem('currentTheme');
}
});
```

Also add the localStorage onto getItem:

```
const currentTheme = localStorage.getItem('currentTheme');
```

To retain the selected brightness theme when refreshing the page:

```
if (currentTheme){
  bodyElement.classList.add('light-theme'); }
```

So now when we selected certain theme, dark or bright, and when we refresh the webpage:

**theme retains bright.PNG**

**theme retains dark.PNG**

### Search Bar:

In index.html:

```
<div class = "search-form-container container"
id="search-form-container">
    <div class = "form-container-inner">
        <form action = "" class="form">
            <input type="text" class="form-input"
placeholder="What are you looking for?">
            <button class="btn form-btn" type="submit">
                <i class="ri-search-line"></i>
            </button>
        </form>
        <span class = "form-note">Or Press ESC to Close</span>
    </div>
    <button class="btn form-close-btn place-items-center"
id="form-close-btn">
        <i class = "ri-close-line"></i>
    </button>
</div>
```

### Search Bar -CSS:

Start with the Search form container:

```
.search-form-container{
    width: 100%;
    max-width: 100%;
    height: 100vh;
    background-color: var(--primary-background-color);
    position: fixed;
    top: 0;
    left: 0;
    z-index: 9999;
    display: flex;
    /*opacity: 0;
    transform: scale(0);*/
    transition: opacity .5s;}
```

Insignificant Error: Search Bar prompt not showing. DEBUGGING: Change display to flex in main.css. And set opacity: 0; transform: scale(0); into comments.

Add some styles for the search bar in main.css:

```
.search-form{
    width: 100%;
    max-width: 100%;
    height: 100vh;
    background-color: var(--primary-background-color);
```



```

    position: fixed;
    top: 0;
    left: 0;
    z-index: 9999;
    display: flex;
    /*opacity: 0;
    transform: scale(0);*/
    transition: opacity .5s;
}

.form-container-inner{
    margin: auto;
    display: flex;
    flex-direction: column;
    gap: var(--gap);
    text-align: center;
}

```

Now we get a properly aligned search bar that prompts for searching from users:

**Search bar.PNG**

Form JavaScript styles:

```

.search-form-container.activated{
    opacity: 1;
    transform: scale(1);
}

```

To close the middle search bar anytime we want, in main.js

We need to modify the open/close search forms popup:

```

const formOpenBtn = selectElement('#search-icon');
const formCloseBtn = selectElement('#form-close-btn');
const searchFormContainer = selectElement('#search-form-container');

```

Enable the click-then-pop-up function by adding server listener for form open button:

```

formOpenBtn.addEventListener('click', () =>
searchFormContainer.classList.add('activated'));

```

Also activate the form close button with add server event:

```

formCloseBtn.addEventListener('click', () =>
searchFormContainer.classList.remove('activated'));

```

So the search bar will be closed when the user clicks the closing button.

Close the search form popup on ESC keypress:

In main.js:

```

window.addEventListener('keyup', event =>{
    if(event.key === 'Escape')
searchFormContainer.classList.remove('activated');
});

```

So when the user presses the ESC key, the search bar will close.

### Featured Articles HTML:

In index.html:

Adding featured details to enrich the home page content:

```
<!-- Featured articles -->
<section class="featured-articles section-header-offset">
  <div class="feature-articles-container container d-grid">
    <div class="featured-content d-grid">
      <div class="headline-banner">
        <h3 class="headline fancy-border">
          <span class="place-items-center">Breaking
News</span>
          </h3>
          <span class="headline-description">Apple Announces
a New Partnership...</span>
        </div>
      ...
```

Now the featured articles sections looks good:

**featured articles initial look.PNG**

### Featured articles-CSS:

Get nicer layout for the background colors:

```
.fancy-border::before{
  content: '';
  position: absolute;
  top: 0;
  left: 0;
  width: 200%;
  height: 100%;
  background: var(--gradient-color);
  background-size: 50% 100%;
  z-index: 5;
}

.fancy-border span{
  position: relative;
  z-index: 10;
  display: inline-flex;
  background-color: var(--secondary-background-color);
}

Also add article effects to enrich the text captions categories:

.article{
  position: relative;
  min-height: var(--item-min-height-md);
}
```

```
.article-image{
  position: absolute;
  width: 100%;
  height: 100%;
  object-fit: cover;
  transition: transform .25s;
}
```

```
.article-category{
  position: absolute;
  top: 1rem;
  right: 1rem;
  padding: 1rem;
  font-size: 1rem;
  background-color: var(--primary-background-color);
  color: var(--light-color);
  text-transform: uppercase;
  z-index: 10;
}
```

```
.article-data-container{
  position: absolute;
  bottom: 0;
  left: 0;
  padding: 2rem 1.5rem;
  background-color: var(--transparent-dark-color);
  z-index: 10;
  width: 100%;
  transition: background-color .5s;
}
```

To make the background solid color instead of hover:

```
.article:hover .article-data-container{
  background-color: var(--hover-dark-color);
  color: var(--hover-light-color); }
```

Customize the titles with some bold and highlighted effects:

```
.title{
  color: var(--light-color); }
.article-title{
  font-size: var(--font-size-md); }
```

For sidebar cards, in index.html:

```
<a href="" class = "trending-news-box">
```

```

        <div class = "trending-news-image-box">
            <span class="trending-number
place-items-center">01</span>
            
        </div>
        <div class = "trending-news-data">
            <div class="article-data">
                <span>October 10th 2022</span>
                <span class="article-data-spacer"></span>
                <span>8 mins read estimated.</span>
            </div>
            <h3 class = "title article-title">Sample
Article Title</h3>
        </div></a>
...

```

#### Add styles to the featured sidebar contents:

In main.css:

```

.sidebar{
    background-color: var(--secondary-background-color);
    padding: 2.5rem;
}

.trending-news-box{
    display: flex;
    flex-direction: column;
    gap: var(--gap);
    padding-block: 2rem;
    border: 1px solid --transparent-light-color;
}

.trending-news-box:last-of-type{
    border-bottom: none;
}

```

Since we used absolute in index.html, so we can use relative here directly to form all the images in box/in grid properly:

```

.trending-news-image-box{
    position: relative; }

```

**all trending news images in box.PNG**

**sidebar features css done-1.PNG sidebar features css done-2.PNG**

**sidebar features css done-3.PNG sidebar features css done-4.PNG**

### Quick Read-HTML:

In index.html, implement the structure for the Swiper carousel:

```
<a href = "#" class = "article swiper-slide">
    <img src
= "../assets/images/quick_read/quick-read-1.png" alt="" class =
"article-image">
    <div class="article-data-container">
        <div class="article-data">
            <span>October 10th 2022</span>
            <span
class="article-data-spacer"></span>
            <span>8 mins read estimated.</span>
        </div>
        <h3 class="title article-title">...
</h3>
    </div>
</a>
```

Add navigation controls for the quick read contents underneath the Swiper wrapper:

```
<div class = "swiper-button-prev swiper-controls"></div>
<div class = "swiper-button-next swiper-controls"></div>
```

### **quick read next navigation shows.PNG**

Add the pagination:

```
<div class = "swiper-pagination"></div>
```

### Quick read CSS:

Add nicer styles and layout for Quick Read section in main.css:

```
.section-title {
    position: relative;
    font-size: var(--font-size-md);
    margin-bottom: 6rem;
}
.section-title::after {
    content: attr(data-name);
    font-size: var(--font-size-lg);
    text-transform: uppercase;
    opacity: .1;
    position: absolute;
    top: 1.5rem;
    left: 2rem;
}
.article.swiper-slide {
    height: 40rem;}
```

### Add more styles to Quick Read swiper:

```
.swiper-wrapper{
  height: auto;
}

.swiper-controls{
  width: 3.5rem;
  height: 3.5rem;
  background-color: var(--secondary-background-color);
}

.swiper-controls::after{
  font-size: 1.6rem;
  color: var(--light-color);
}
```

### SwiperJS:

For the pagination to show up, first initialize the swiper. So in main.js:

```
const swiper = new Swiper('.swiper', {
  slidesPerView: 1,
  spaceBetween: 20,
  navigation:{
    nextEl: '.swiper-button-next',
    prevEl: '.swiper-button-prev'
  },
  pagination: {
    el: '.swiper-pagination'
  }
});
```

Now the swiper pagination works well as we sliding left and right:

**swiper pagination works-1.PNG**

**swiper pagination works-2.PNG**

**swiper pagination works-3.PNG**

**swiper pagination works-4.PNG**

**swiper pagination works-5.PNG**

**swiper pagination works-6.PNG**

Add responsive breakpoints to separate the blocks:

```
breakpoints: {
  700: {
    slidesPerView: 2},
  1200: {
    slidesPerView: 3
  }, }
```

Now if we expand the webpage, all quick-read grid are aligned horizontally separated:

**quick read in separate grid after adding breakpoints.PNG**

Also set up styles for the swiper pagination bullet points in main.css:

```
.swiper-pagination{
  bottom: .5rem !important;
}
.swiper-pagination-bullet{
  background-color: var(--light-color); }
```

Now the bullet points underneath the quick-read contents look more striking:

**more striking bullet points.PNG**

Older Posts - HTML:

In index.html:

To make overshadowing effect:

```
<h2 class = "title section-title" data-name="Older posts">Older
posts</h2>...
```

Adding images to the older-posts swiper wrapper grids:

```
<a href="#" class = "article d-grid">
  <div class = "older-posts-article-image-wrapper">
    <img src =
"./assets/images/older_posts/older-post-1.png" alt=""
class="article-image">
  </div>
  <div class="article-data-container">

    <div class="article-data">
      <span>October 10th 2022</span>
      <span class="article-data-spacer"></span>
      <span>8 mins read estimated.</span>
    </div>
    <h3 class="title article-title">Are Tesla
Electric Cars with Its Radiation Bad for Human
Health?
    </h3>
    <p class = "randomdescription...">
  </div>
</a>
```

Now we have all the cards here with each individual image:

**all older posts cards with individual image.PNG**

Older posts - CSS:

In main.css:

```
.older-posts-grid-wrapper{
  gap: var(--gap);
}
.older-posts-article-image-wrapper,
```

```

.older-posts .article{
  min-height: var(--item-min-height-sm);
}
.older-posts .article{
  background-color: var(--secondary-background-color);
}
.older-posts-article-image-wrapper{
  position: relative;
}
.older-posts-grid-wrapper .article-data-container{
  position: static;
  background-color: transparent;
}
.older-posts-grid-wrapper .article-title{
  margin-bottom: var(--margin-sm);
}

```

#### Popular Tags-HTML:

```

<a href="#" class="article">
  <span class="tag-name">#travel</span>
  
  </a> ...

```

#### Popular Tags -CSS:

```

.popular-tags-container{
  gap: var(--gap);
}

.popular-tags-container .article{
  min-height: var(--item-min-height-sm);
}

.tag-name{
  position: absolute;
  top: 50%;
  left: 50%;
  transform: translate(-50%, -50%);
  background-color: var(--transparent-dark-color);
  padding: 1rem 2rem;
  z-index: 10;
}

```

So now the padding tags appear on the center of each image grid:

**popular tags initial look.PNG**



### Newsletter - HTML:

```
<section class="newsletter section">
  <div class="container">
    <h2 class="title section-title"
data-name="Newsletter">Newsletter</h2>
    <div class="form-container-inner">
      <h6 class="title newsletter-title">Subscribe to
BloggieBlog</h6>
      <p class="newsletter-description"> h
sihewjdufufuyefrertryruyferyreyyfggyeryreyytryetyetuwyt</p>
      <form action="" class="form">
        <input type="text" class="form-input"
placeholder="Enter your email address.">
        <button class="btn form-btn" type="submit">
          <i class="ri-mail-send-line"></i>
        </button>
      </form>
    </div>
  </div>
</section>
```

Now we have the icon and blank that prompt users to subscribe to the blog's newsletter:

**newsletter section initial look.PNG**

### Newsletter - CSS:

```
.newsletter{
  padding-bottom: 6rem;
}
.newsletter-title{
  font-size: var(--font-size-lg);
}
.newsletter-description{
  margin-bottom: 3rem;
}
```

Now we have: **newsletter final look.PNG**

### Footer HTML:

To add all list of data, along with social media contacts:

```
<footer class = "footer section">
  <div class = "footer-container container d-grid">
    <div class = "company-data">
      <a href="./index.html">
        <h2 class="logo">BloggieBlog</h2>
      </a>
      <p class = "company-description">
```

```

hsg uddiyduye
ufftfafeutetfteuetrfeuftteywayweudfewudaew
</p>
<ul class = "list social-media">
  <li class = "list-item">
    <a href="#" class = "list-link">
      <i class="ri-instagram-line"></i>
    </a>
    <a href="#" class = "list-link">
      <i class="ri-facebook-circle-line"></i>
    </a>
    <a href="#" class = "list-link">
      <i class="ri-twitter-line"></i>
    </a>
    <a href="#" class = "list-link">
      <i class="ri-pinterest-line"></i>
    </a>
  </li>

```

Now we have all button-alike social media icons:

### footer social media icons.PNG

Add a reserved copyright to BloggieBlog:

```

<span class = "copyright-notice">&copy; 2022 BloggieBlog. All rights
reserved.</span>

```

### all copyrights reserved.PNG

To add different lists of items to the footer:

```

<div>
<h6 class="title footer-title">Useful Links</h6>
<ul class="list footer-list">
  <li class="list-item">
    <a href="#" class="list-link">Sample Item</a>
  </li>
  <li class="list-item">
    <a href="#" class="list-link">Sample Item</a>
  </li>
  <li class="list-item">
    <a href="#" class="list-link">Sample Item</a>
  </li>
  <li class="list-item">
    <a href="#" class="list-link">Sample Item</a>
  </li>
  <li class="list-item">
    <a href="#" class="list-link">Sample Item</a>
  </li>

```

```

        <li class="list-item">
            <a href="#" class="list-link">Sample Item</a>
        </li>
    </ul>
</div> ...

```

Now we have **footer sample items look.PNG**

#### Footer-CSS:

```

.footer{
    background-color: var(--secondary-background-color);
}
.footer-container{
    gap: var(--gap);
}
.company-data{
    display: flex;
    flex-direction: column;
    order: 1;
}
.company-description{
    margin-block: var(--margin-sm);
    max-width: 75%;
}
.social-media{
    margin-bottom: var(--margin-md);
}
.copyright-notice{
    opacity: .8;
}
.footer-list{
    flex-direction: column;
    align-items: flex-start;
}
.footer-title{
    font-size: var(--font-size-md);
    text-transform: uppercase;
    margin-bottom: var(--margin-sm);
}

```

**vertically aligned footer items.PNG**

## Blog Posts - HTML:

To add contents to the featured blogs, in post.html:

```
<section class="blog-post section-header-offset">
  <div class="blog-post-container container">
    <div class="blog-post-data">
      <h3 class="title blog-post-title">Is Venom 3 Sequel
Related to Spiderman's Origin?</h3>
      <div class="article-data">
        <span>October 10th 2022</span>
        <span class="article-data-spacer"></span>
        <span>8 mins read</span>
      </div>
      
    </div>
    <div class = "container">
      <p>hh
uw9iudhidewytfuytwefuytsdufgfyertfyuewtytetfw7rewyftsfehtwf
      </p>
      <blockquote class = "quote">
        <p>
          <span>
            <i class = "ri-double-quotes-l">
              </i>
            </span>hasudisgykdhksuyfgaskfk7ygsdggwDTIDTYDTDYTLI
            <span><i class = "ri-double-quotes-r">
              </i>
            </span>
          </p>
        </blockquote>
```

Also add the author's details/description underneath the article:

```
<p>jgjdwydgkuwgkuetwfekrkftg9aegfkuglyfkgkaiyldglfvurdgvfkuyi</p>
  <div class = "author d-grid">
    <div class="author-image-box">
      <img src = "./assets/images/author.jpg" alt=""
class = "article-image">
    </div>
    <div class = "author-about">
      <h3 class = "author-name">John Doe</h3>
<p>hdbiqwgkduhlgfwyuifsxgkyjsxdbusygydyusjdbuatd</p>
```

**blog post content and author details.PNG**

### Blog posts - CSS:

Select the box container style:

```
.blog-post{
  padding-bottom: 5rem;
}
.blog-post-data{
  display: flex;
  flex-direction: column;
  align-items: center;
  text-align: center;
  padding-block: 2.5rem;
}
.blog-post-title{
  font-size: clamp(3rem, 5vw, 6rem);
}
.blog-post-data .article-data{
  margin-bottom: 2.5rem;
}
.blog-post-data img{
  width: 100%;
  height: var(--item-min-height-sm);
  object-fit: cover;
  object-position: center 10%;
}
.blog-post-container .container{
  max-width: 90rem;
  padding: 0;
  display: flex;
  flex-direction: column;
  gap: var(--gap);
}
.quote{
  background-color: var(--secondary-background-color);
  padding: 2.5rem;
}
.quote p{
  font-style: italic;
}
.quote span i{
  font-size: 3.5rem;
}
```

Now the blog posts look like: **blog post overview look.PNG**

### Media Queries:

Arrange Trending News titles to align on the side of each grid in main.css:

```
@media screen and (min-width: 700px){  
  .trending-news-box{  
    flex-direction: row;  
    align-items: center;  
  }  
  
  .trending-news-image-box{  
    width: 50%;  
  }  
  .trending-news-data{  
    margin: auto;  
    text-align: center;  
  }  
}
```

Now we have **remodified trending news alignment.PNG**

Also change the layout of the Popular Tag section to be side-by-side layout:

```
.popular-tags-container,  
.footer-container{  
  grid-template-columns: repeat(2, 1fr );  
}
```

### **side-by-side popular tags.PNG**

Adding styles to Featured articles to make them aligned better:

```
@media screen and (min-width: 768px) {  
  .featured-content {  
    grid-template-columns: repeat(4, 1fr);  
    grid-template-rows: 5rem repeat(4, 1fr);}  
  .headline-banner {  
    grid-column: 1/-1;  
    font-size: var(--font-size-sm);  
  }  
  .featured-article{  
    grid-column: 3/-1;  
    grid-row: 2/4;}  
  .featured-article-1{  
    grid-column: 1/3;  
  }  
  .featured-article-3 {  
    grid-column: 1/-1;  
    grid-row: 4/6;}  
}
```

Stretch the webpage and Featured articles are properly aligned:

**better aligned featured articles.PNG**

Also add styles to older posts:

```
.older-posts .article{
  grid-template-columns: 25rem 1fr;
}
```

So we get: **horizontally aligned older posts.PNG**

Add redirection links to Menu:

```
@media screen and (min-width: 1024px) {
  .menu{
    position: initial;
    width: initial;
    padding: initial;
    background-color: transparent;
    opacity: initial;
    transform: initial;
  }
  .menu > .list{
    flex-direction: row;
  }
}
```

Now we have the horizontally aligned Menu bar on the very top:

**menu bar on the very top.PNG**

Add Neon effects to Sign in and Sign up boxes:

```
.sign-up-btn{
  padding: .4rem;
}
.btn.fancy-border:hover::before{
  animation: animate_border .75s linear
  infinite;
}
@keyframes animate_border{
  to{
    transform: translateX(-50%);
  }
}
```

So now we have neon-colored sign-up button:

**neon color hover sign up button.PNG**

Edit all contents, then the final overview of BloggieBlog:

**[BloggieBlog Overview-1.PNG](#)**

**[BloggieBlog Overview-2.PNG](#)**

**[BloggieBlog Overview-3.PNG](#)**

**[BloggieBlog Overview-4.PNG](#)**

**[BloggieBlog Overview-5.PNG](#)**