

```

.naglowek{
  display: grid;
  height: 40px;
  min-height: 20px;
  grid-template-columns: auto 70px 70px 70px auto 70px 70px auto;
  .g1{
    background-color: #cecdcd;
  }
  .g2{
    @include centerflex();
    background-color: #ebebeb;
  }
  .g3{
    background-color: #ebebeb;
  }
  .g4{
    @include centerflex();
    background-color: #9e9d9d;
  }
  .g5{
    @include centerflex();
    background-color: $fotlog-color;
  }
  .g6{
    background-color: #9e9d9d;
  }
}

```

Zdefiniowanie w pliku style.scss zmiennych i mixin które będą używane na przestrzeni całego projektu. Głównie często wykorzystywanych kolorów i grup stylów.

```

<body>
  <div class="naglowek">
    <div class="g1"></div>
    <div class="g2"><a class="g2" href="/Blu.png">Buy</a></div>
    <div class="g2"><a class="g2" href="">Sell</a></div>
    <div class="g2"><a class="g2" href="">About</a></div>
    <div class="g3"></div>
    <div class="g4"><a class="g4" href="">Register</a></div>
    <div class="g5"><a class="g5" href="">Login</a></div>
    <div class="g6"></div>
  </div>

```

Stworzenie struktury nagłówka z nadaniem odpowiednich klas dodaniu odnośników

```

.naglowek{
  display: grid;
  height: 40px;
  min-height: 20px;
  grid-template-columns: auto 70px 70px 70px auto 70px 70px auto;
  .g1{
    background-color: #cecdcd;
  }
  .g2{
    @include centerflex();
    background-color: #ebebeb;
  }
  .g3{
    background-color: #ebebeb;
  }
  .g4{
    @include centerflex();
    background-color: #9e9d9d;
  }
  .g5{
    @include centerflex();
    background-color: $fotlog-color;
  }
  .g6{
    background-color: #9e9d9d;
  }
}

```

.naglowek odpowiada za poprawne rozmieszczenie divów przy pomocy grida poprzez zdefiniowanie 8 kolumn i nadanie im odpowiedniej szerokości gdzie auto oznacza zajęcie całej dostępnej zawartości.

W dalszej części są ustalane właściwości poszczególnych bloków według ich przypisanej klasy od g1 do g6 gdzie background-color oznacza ich kolor a @include centerflex(); umieszcza zawartość poszczególnych bloków na środku.

```

<div>
  <ul class="breadcrumbs">
    <li><a href="#Home">Home</a></li>
    <li><a>Login</a></li>
  </ul>
</div>

```

Stworzenie listy która będzie pełniła funkcję Breadcrumbs

```

.breadcrumbs {
  background-color: #f4f4f4;
  margin-top: 0;
  margin-bottom: 30px;
  text-align: left;
  padding-left: 25%;
  li{
    display: inline;
  }
  ul{
    text-align: center;
  }
}
.breadcrumbs > li{
  @include marginPadding();
  display: inline;
}
.breadcrumbs li + li ::before{
  @include marginPadding();
  content: "/";
}
.breadcrumbs li + li {
  color: grey;
}

```

Kod który ustawia kolor, zmienia margines górny na 0, ustawia margines dolny by powstała przerwa między następnym elementem a tym, wyrównanie tekstu do lewej strony, i ustawienie padding-left by zawartość znajdowała się bliżej centralnej części strony.

display: inline; sprawia że lista jest wyświetlana w poziomie nie pionie

Odnoszę się do pierwszego elementu umieszczonego po elemencie li później odwołuję się do elementu ::before co pozwala mi dodać znak / przed Login dzięki temu powstaje ścieżka do podstrony login Home / Login

Ustawiam kolor dla Login jako szary

```

<div class="cialo">
  <h1>Login</h1>
  <h5>To acces your account</h5>
  <div class="log">
    <form>
      <fieldset>
        <p>Username</p>
        <input type="text" name="username" maxlength="100" required>
        <p>Email Adress</p>
        <input type="email" name="email" required>
        <p>Password</p>
        <input type="password" name="password" minlength="7" maxlength="25" required>
        <br>
        <div class="f_password">
          <a href="" class="f_password">Forgot Password?</a>
          <input id="form_button" type="submit" value="Login">
        </div>
      </fieldset>
    </form>
    <div class="right">
      <div class="front">
        <p>Not a member?
        <br>
        <input type="submit" value="Register today">
        </p>
      </div>
      
    </div>
  </div>
</div>

```

Umieszczenie w divie log formularza oraz bloku right w którym będzie znajdować się grafika Blu.png, paragraf i przycisk . Znacznik <form> zawiera 4 znaczniki <input> typu text, email, password, submit.

text: jest odpowiedzialny za nazwę użytkownika posiada ograniczenie co do maksymalnej ilości znaków i jest polem wymaganym

email: Dzięki temu typowi przeglądarka automatycznie sprawdza poprawność adresu email i też jest polem wymaganym

password: odpowiada za hasło, posiada minimalną oraz maksymalną liczbę znaków jakie trzeba umieścić i jest wymagane

submit: przesyła formularz

```

cialo{
  width: 800px;
  height: 350px;
  margin: auto;
}

```

Ustawienie wielkości dla diva w którym będzie znajdować się główna część strony

```

.log {
  display: grid;
  width: 100%;
  height: 100%;
  grid-template-columns: 50% 50%;
  margin: auto;
  margin-top: 40px;
  background-color: $formbg-color;
  border-style: solid;
  border-width: 1px;
  border-color: $border-color;
  border-radius: 3px;
}

```

Ustawienie wyświetlania zawartości log jako grid ustawienie wielkości jako 100% tego co zostało ustawione w ciało i utworzenie dwóch kolumn zajmujących po 50% wartości width log. Reszta odpowiada stylowi pola na którym będzie znajdował się formularz i prawy blok

```

fieldset{
  border-style: none;
  margin-top: 20px;
  margin-bottom: 5px;
  margin-left: 7px;
  width: 100%;
  background-color: #f0f0f0;
}

```

Odpowiada za wygląd pola formularza i robi odstępy między poszczególnymi elementami strony

```

.right{
  background-color: $formbg-color;
  width: 50%;
  height: inherit;
  justify-self: end;
  position: relative;
  margin-right: 30px;
}

```

.right tworzy prostokąt którego długość wynosi 50% kolumny grid w której się znajduje i umieszcza go na końcu kolumny z 30px prawym marginesem

```

.front{
  border-top-style: solid;
  border-bottom-style: solid;
  border-width: 1px;
  border-color: $border-color;
  @include centerflex();
  position: relative;
  text-align: center;
  background-color: $formbg-color;
  width: 100%;
  height: inherit;
  z-index: 2;
}

```

.front znajduje się w tej samej pozycji co .right ma ten sam format ale umieszcza zawarte w nim elementy na środku oraz nadaje mu z-index:2 co sprawia że jest nad innymi elementami na stronie

```

img{
  position: absolute;
  width: 100%;
  height: 130%;
  top: -15%;
  z-index: 1;
}

```

img otrzymuje pozycję absolute i jest potomkiem .front i .right o pozycjach relative dodatkowo jego z-index wynosi 1 co pozwala umieścić go za .front.

```
.footer{  
  position: fixed;  
  height: 10%;  
  width: 100%;  
  bottom: 0;  
  background-color: $fotlog-color;  
  margin: 0%;  
  z-index: -1;  
}
```

Ustawia Stopkę na samym dole strony oraz sprawia że po zmniejszeniu stopka będzie za formularzem a nie przed nim

```
@media screen and (max-width: 800px) {  
  .cialo, .log {  
    width: 100%;  
  }  
}
```

Odpowiada za poziomą responsywność strony poprzez nadanie .cialo i .log właściwości width 100% kiedy szerokość będzie mniejsza od 800px