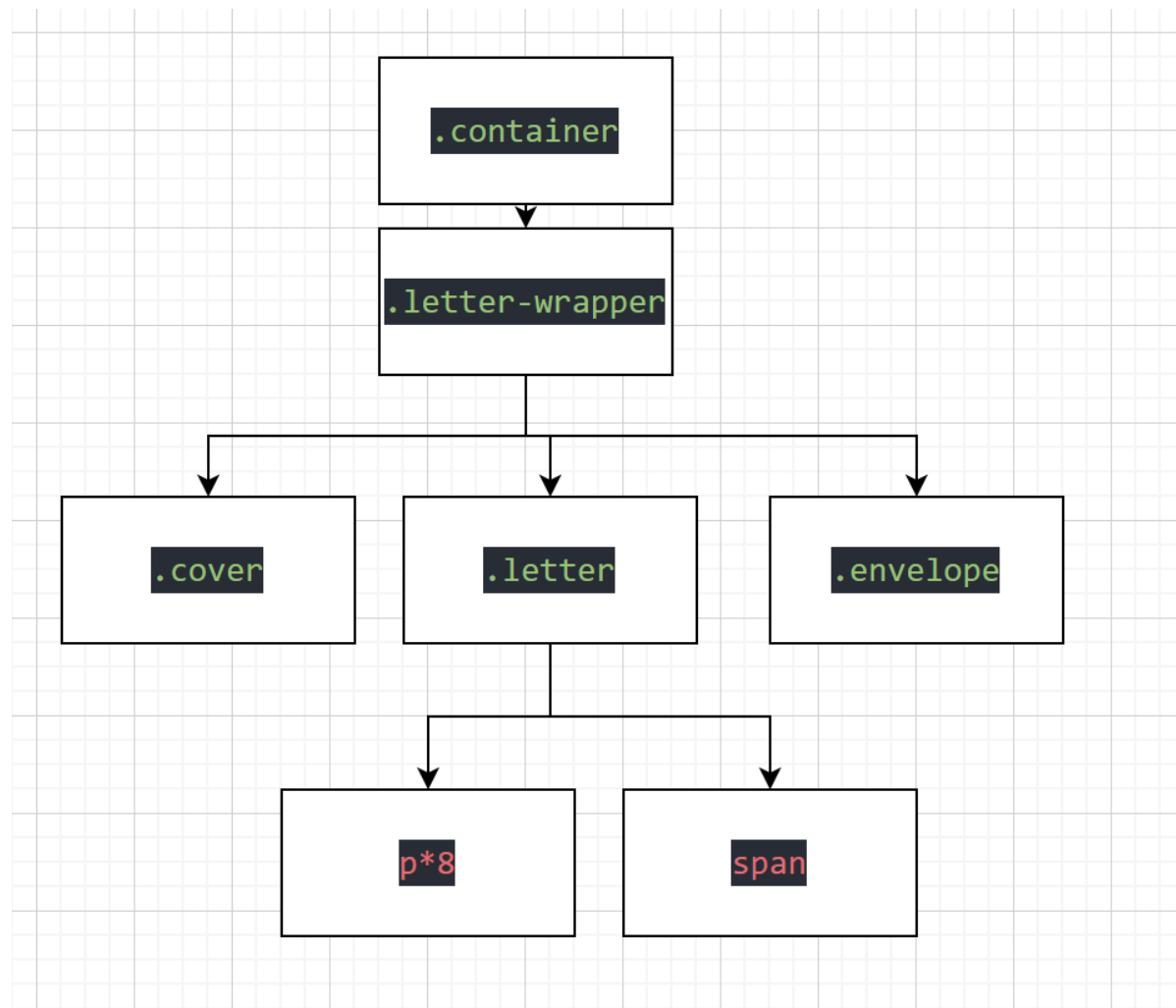


信封

一.html结构



二.格式

1.全局

```
* {
  box-sizing: border-box;
  margin: 0;
  padding: 0;
}
:root {
  --clr: #039d5d;
}
body {
  width: 100%;
  min-height: 100vh;
  background: var(--clr);
  display: flex;
  justify-content: center;
  align-items: center;
}
```

2.container

```
.container {
  position: relative;
  width: 500px;
  height: 600px;
  overflow: hidden;
  display: flex;
  flex-direction: column;
  justify-content: flex-end;
  align-items: center;
  /* cubic-bezier用来控制整体速度 */
  animation: 2.2s cubic-bezier(.37, -0.09, 0, 1.6) bounceInUp;
}
@keyframes bounceInUp {

  /* 由1/10 往下走 上升时成为正常大小 */
  /*从原始位置往下2000px*/
  from {
    opacity: 0;
    transform: translateY(2000px) scale(.1);
  }
  65% {
    opacity: 1;
    transform: translateY(-100px);
  }
  75% {
    transform: translateY(20px);
  }
  90% {
    transform: translateY(-20px);
  }
  100% {
    transform: translateY(0) scale(1);
  }
}
```

3.letter-wrapper

```
.container .letter-wrapper {
  position: relative;
  width: 100%;
  height: 360px;
  background-image: url(./images/bg.png);
  background-repeat: no-repeat;
  display: flex;
  justify-content: center;
  align-items: flex-end;
}
```

- align-items:flex-end究竟有什么作用，我们可以消去cover,envelope部分进行观察，同时将overflow:hidden消去。

```
align-items: flex-start;
```

@baizexiaotang

嗨 同学你好!

我是小唐，一名Web前端开发者。

我在这里想通过简单高效直接的方式分享Web

```
align-items: center;
```

@baizexiaotang

嗨 同学你好!

我是小唐，一名Web前端开发者。

```
align-items: flex-end;
```

@baizexiaotang

嗨 同学你好!

我是小唐，一名Web前端开发者。

我在这里想通过简单高效直接的方式分享Web

开发技能和知识，来帮助大家成为一名Web前端开发

- 容易发现是白色部分相对于黄色部分的位置，而黄色部分正是letter-wrapper,而白色部分是letter，align-items: flex-end决定了letter在上下方向上如何与letter进行对齐

4.cover

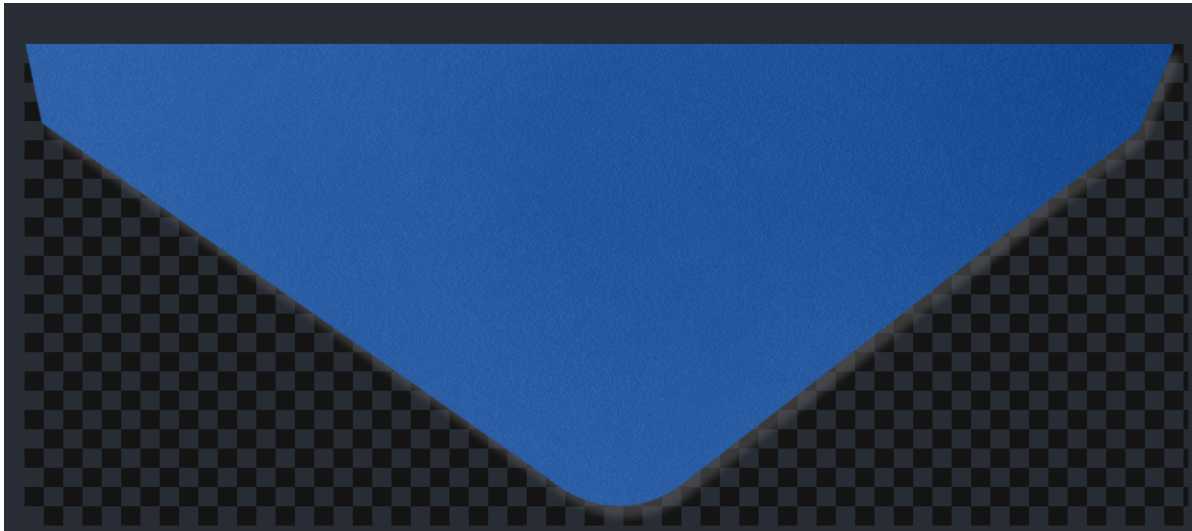
```
.container .letter-wrapper .cover {
  position: absolute;
  width: 100%;
  height: 210px;
  z-index: 5;
  background-image: url(../images/cover-front.png);
  /* background-position: 0; */
  /* background-repeat: no-repeat; */
  background-size: cover;
  top: 0;
  /* forwards停留在最后一帧 */
  /* 有.3s的原因:因为在0秒时container非常小，动画效果看不到，所以要在.3秒开始 */
  animation: .8s .3s rollingOver forwards;
  /* 初始的原点在元素中心，而下面的意思是以元素左上角为
  参考点，x轴方向上不变，而y轴方向上向下5px
```

```

    令人疑惑的是原点坐标为0 0时会让信封多往上旋转 */
    transform-origin: 0 5px;
}
@keyframes rollingover {
    10% {
        background-image: url(../images/cover-front.png);
    }
    100% {
        z-index: 2;
        background-image: url(../images/cover-back.png);
        /* 向上旋转 */
        transform: rotateX(180deg);
        /* transform: rotateX(-180deg); 向下旋转
        transform: rotateY(180deg); 向左旋转
        transform: rotateY(-180deg); 向右旋转*/
    }
}

```

- 原图如下:

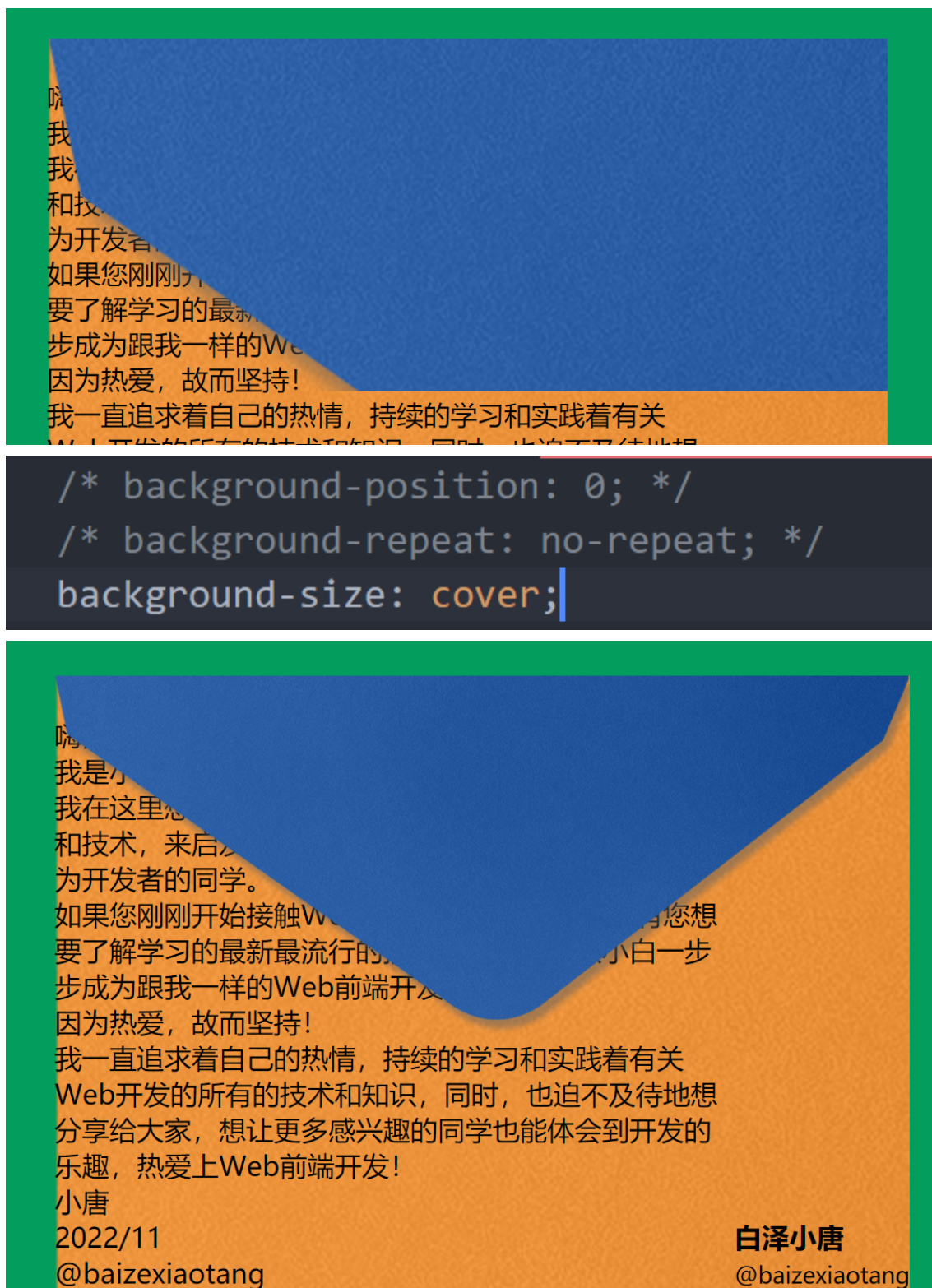


- size属性的存在与否

```

/* background-position: 0; */
/* background-repeat: no-repeat; */
/* background-size: cover; */

```



- 元素的大小和图片的形状并不一致，此时就需要background-size这一属性
- cover旋转会让深度和颜色变化，一开始深度大是为了遮盖字体，疑惑：cover到最后本由于其位置就不会遮住字体，那么变化其深度用意何在呢？错误，在动画中字体也会向上弹，因此要变化其深度

5.letter

```
.container .letter-wrapper .letter {  
  position: absolute;  
  z-index: 3;  
  width: 450px;
```

```

height: 500px;
/* margin: 20px; */
/* 这里的上距和下距是无用的 */
padding: 35px 35px 35px;
background: #fff;
border-radius: 16px;
/* 元素右边和下边具有阴影，其实下边的阴影是看不到的 */
box-shadow: 2px 5px 10px #000;
transform: translateY(180px);
animation: .7s .7s forwards letterBounceInUp;
}
@keyframes letterBounceInUp {
  from {
    transform: translateY(180px);
  }
  40% {
    transform: translateY(-20px);
  }
  75% {
    transform: translateY(10px);
  }
  90% {
    transform: translateY(-10px);
  }
  100% {
    transform: translateY(0px);
  }
  /* 晃动轨迹：下-->上-->下-->上-->原处 */
}

.container .letter-wrapper .letter p {
  font-size: 18px;
  line-height: 1.6;
  color: #213547;
  /* 同一个p内字间上下为10px,左右为0px */
  margin: 10px 0;
}
/* 可以看到这是一个字体元素，而在其中有1个字体 */
.container .letter-wrapper .letter p:first-child::first-letter {
  font-size: 60px;
  line-height: 1;
  /* float CSS 属性指定一个元素应沿其容器的左侧或右侧放置，允许文本和内联元素环绕它。
  该元素从网页的正常流动（文档流）中移除，尽管仍然保持部分的流动性（与绝对定位相反）
  其他字虽然不是同一个p，但仍然可以围绕它*/
  float: left;
  margin-right: 8px;
}
.container .letter-wrapper .letter span {
  position: absolute;
  font-size: 24px;
  /* 字体粗细 */
  font-weight: 700;
  /* letter上边朝下20px,右边朝左25px */
  top: 20px;
  right: 25px;
}

```

```
    opacity: .1;
}
```

6.envelope

```
.container .letter-wrapper .envelope {
  position: absolute;
  width: 500px;
  height: 360px;
  z-index: 4;
  background-image: url(./images/envelope.png);
  /* 下两个属性无用 */
  /* background-position: center;
  background-repeat: no-repeat; */
  /* 元素和背景图片形状不同 */
  background-size: cover;
  display: flex;
  /* 纵轴为主轴，先按主轴排列 */
  flex-direction: column;
  /* 主轴方向上位于end处 */
  justify-content: flex-end;
  align-items: center;
  /* 调整flex-end的影响 */
  padding-bottom: 40px;
  font-size: 24px;
  color: #000;
}

.container .letter-wrapper .envelope b {
  opacity: .3;
}

.container .letter-wrapper .envelope small {
  opacity: .3;
  font-weight: 200;
}
```