

# 1031

1. adapt showlyric
  - i. take in a parameter like "isNew" (boolean)
  - ii. only span in this class are applied with the animation
  - iii. add on to the origin instead of just displaying
  - iv. 带有isNew属性的不会清空上一行，但是false的会进行一个清空
  - v. beats不需要管还是按照原本的逻辑

# 1101

新写好的showlyric逻辑：

1. 先传入lyric，判断 isNew property
  - i. 如果为false即清空屏幕上所有的span (div)
  - ii. 如果为true，略过清空
2. 对lyric进行拆span处理
  - i. 正则表达式：找到所有的 <br> 和 <span>，split 函数会在分割后保留匹配的标签
    - a. 匹配 <br>
    - b. <span[^>]\*>.\*?<\/span>：匹配所有 <span>
      - a. <span[^>]\*> 匹配 <span> 标签开头部分（所有block）以及所有属性
      - b. .\*? 匹配标签中的任意内容（非贪婪）
  - ii. .map(part => { ... })
    - a. map函数遍历每个分割后的部分
    - b. if (part === '<br>')
      - a. 直接返回 <br>，保留html换行
    - c. else if (part.match(/<span[^>]\*>.\*?<\/span>/))
      - a. 说明它是一个 <span> 标签包裹的内容
      - b. 不做任何处理
    - d. else
      - a. 说明它是纯文本
        - a. 使用 split('') 将该文本字符串拆分成单个字符
        - b. 对每个字符加上 <span class="\${spanClass}">，这样每个字符会被包裹在 <span> 标签中
        - c. spanClass 是一个变量，代表要添加的样式类

e. `.join('')`

a. 最后将 `map` 返回的结果拼接成一个完整的字符串

3. 根据 `isNew` 的值决定是否将新内容追加

i. `lyricDiv.html()` 用来获取`lyricDiv`元素的当前HTML内容

ii. 如果为`true`，将新的歌词内容`chars`追加到已有内容的后面

iii. 如果为`false`，则用`chars`直接覆盖原有内容

4. 动画效果处理

i. 只将动画效果作用在 `isNew` 为 `true` 的`span`上

bug: 第一行歌词不显示（大概率是因为4）

そしてクオリアを

持った，出场时就没有动画

debug要避免一个问题，就是当新的内容被添加到屏幕上的时候，已经在屏幕上的歌词不要重新过一遍动画，毕竟这个动画是作用在所用的`span`上的

## 1102

思路：无论 `isNew` 是 `true` 还是 `false`，新添加的歌词字符都带有一个特定的类

16:00

成功实现

debug过程略

- 除了这个，为了debug学了一点jQuery，但还是决定用原生js
- 把`lyricDiv.html()`替换为`lyricDiv.innerHTML`

o1有帮助到

I understand that you're experiencing an issue where the first line of text (when `isNew = false` and previous text is cleared) does not have the animation applied, even after removing the `if (isNew)` condition around the animation logic. Let me help you diagnose and resolve this problem.

### Problem Explanation:

The main issue is that when `isNew` is `false`, the spans wrapping your characters **do not have the class** `new-lyric`. Since your animation targets `span.new-lyric`, the animation doesn't apply to these spans because they lack the necessary class.

Here's the critical part of your code:

javascript

 复制代码

```
// Define the class to add to new spans for animation
let spanClass = isNew ? 'new-lyric' : '';
```

When `isNew` is `false`, `spanClass` becomes an empty string (`''`), so the spans created do not have any class. Consequently, the animation targeting `span.new-lyric` doesn't find any elements to animate when `isNew` is `false`.

### Solution:

修改:

1. 无论`isNew`的值, 都给新字符添加`animate-me`类 (每次有新字符添加到`lyricDiv`时, 都对其应用动画)
2. 动画针对所有`animate-me`类的元素, 包括清空后添加的新内容&追加, 完成后, 移除

至此换行逻辑成功实现

调试每个小节的换行时机

17:00

至此换行逻辑成功实现

bug: 在后半段歌词以0.5拍急速刷新的部分, 下一句歌词进来的时候, `animation`还没结束, 导致`animate`状态又被刷新的情况

debug思路: 在新歌词添加时立即移除`animate-me`类

```

103
104     anime({
105     ..... begin: function() {
106     .....     document.querySelectorAll('span.animate-me').forEach(function
107     .....         (el) {
108     .....             el.classList.remove('animate-me');
109     .....         });
110     .....     },
111     targets: 'span.animate-me',
112     translateX: [50, 0],
113     delay: anime.stagger(35),
114     duration: 500,
115     easing: 'easeOutExpo',
116     });

```

吃了个晚饭

bug: 详见视频■■も■■も■■も伝えられなきゃ这一段

实际改变block的宽度会导致文字被推挤



debug思路: 方块一开始就占据最终的宽度, 只是通过动画让其从左到右逐渐显示为可见