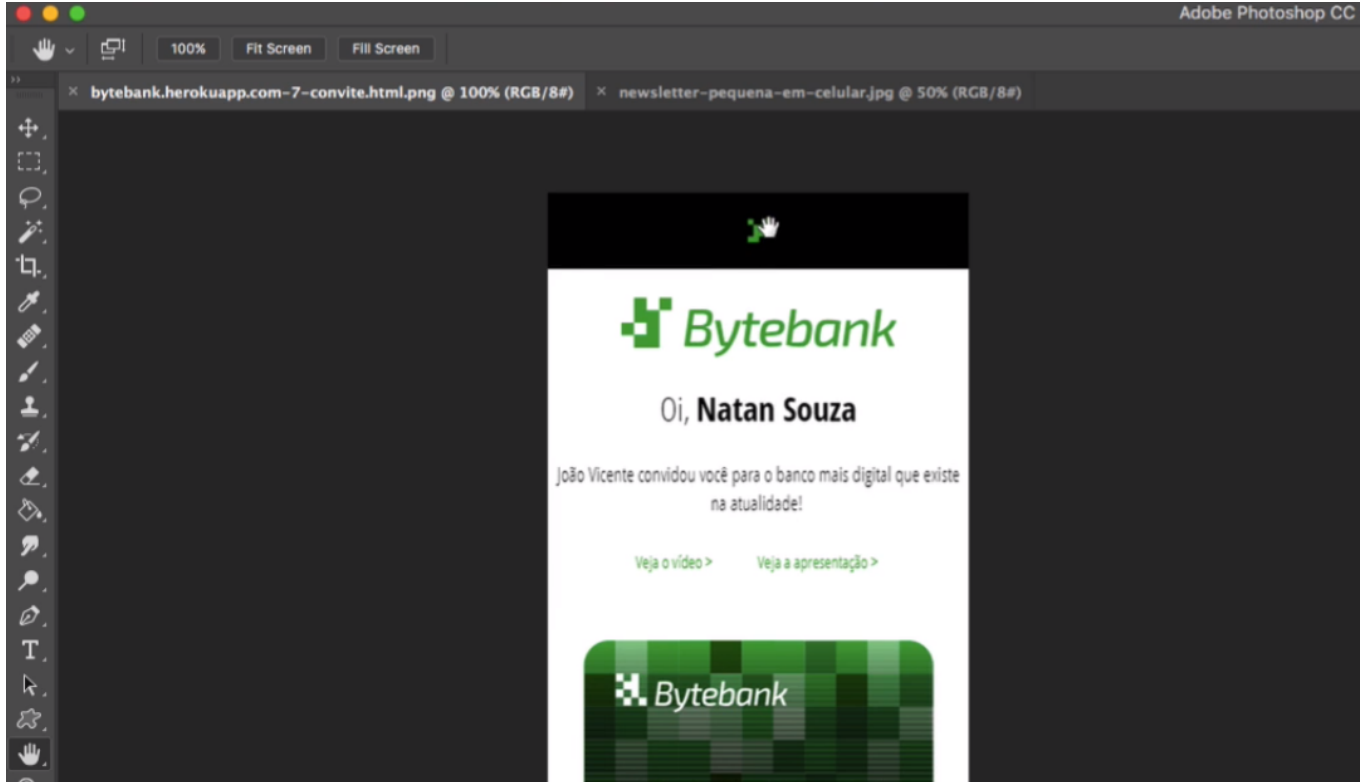


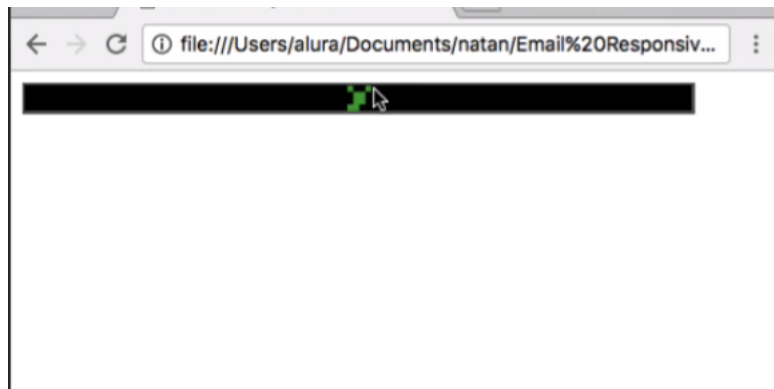
## Linhas e colunas

### Transcrição

Vamos abrir o layout final do e-mail no Photoshop:



Observe que a linha do topo do e-mail tem uma largura grande. No entanto, na nossa página ainda falta um respiro acima e abaixo da imagem.



Para criarmos esse respiro, trabalharemos com o *padding* no logo do cabeçalho. Para isto, criaremos uma classe com o nome bem relacionado: `cabecalho-logo`.

```
<table border="1" width="90%">
  <tr bgcolor="#000000">
    <td align="center" class="cabecalho-logo">
      
    </td>
  </tr>
</table>
```

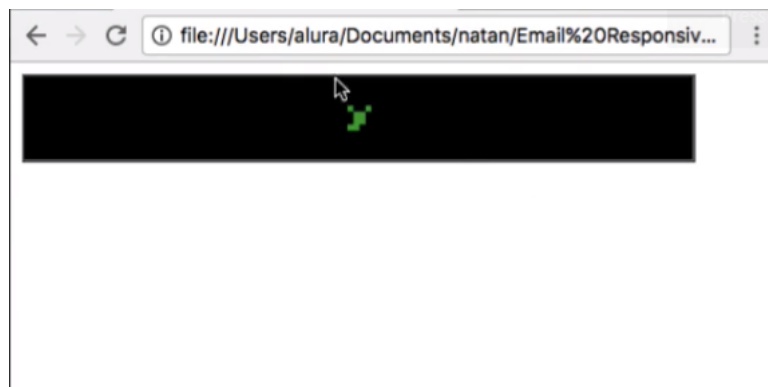
Na parte interna de estilo no arquivo, criaremos a classe `.cabecalho-logo`, em que incluiremos o `padding`.

```
<style>
  img {
    display: block;
  }

  .cabecalho-logo {
    padding: 20px 0;
  }
</style>
```

Já veremos a diferença no browser, com o cabeçalho mais similar ao layout. Vamos alterar o valor do `padding` para `15px`:

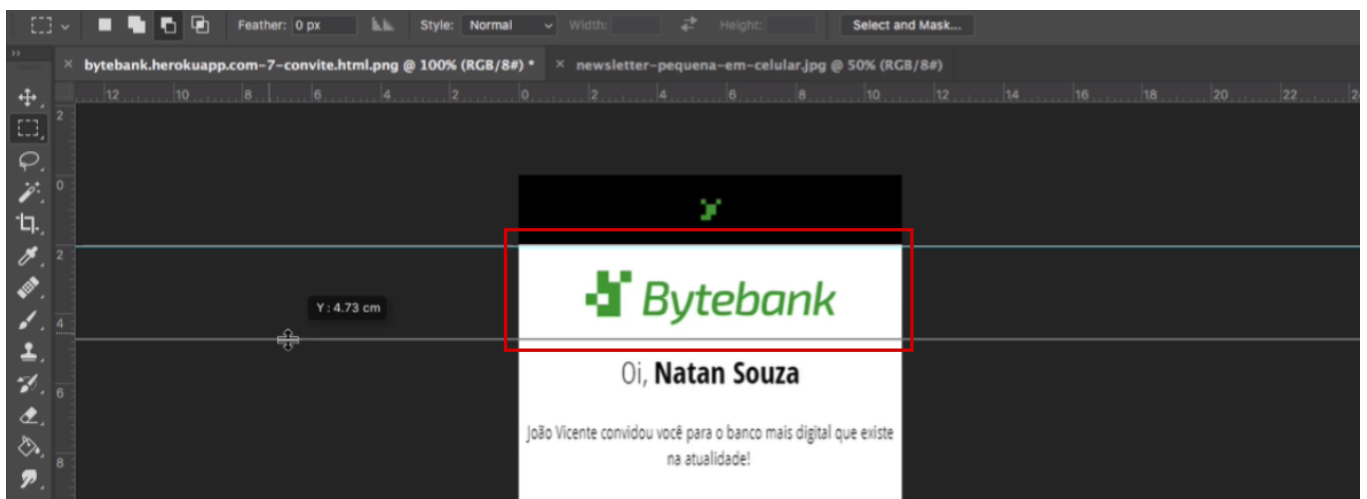
```
.cabecalho-logo {
  padding: 15px 0;
}
```



O próximo passo será incluir o logo do Bytebank.

Neste curso, você não precisa de Photoshop. Estamos utilizando apenas para visualização dos layouts do e-mail.

Criamos o cabeçalho na primeira `<tr>` da tabela, e agora adicionaremos uma segunda para incluir a imagem do logo.



Se preferir, pode-se reaproveitar o código criado anteriormente, aquele que deixamos comentado. No entanto, com a segunda <tr>, nosso código ficará da seguinte maneira:

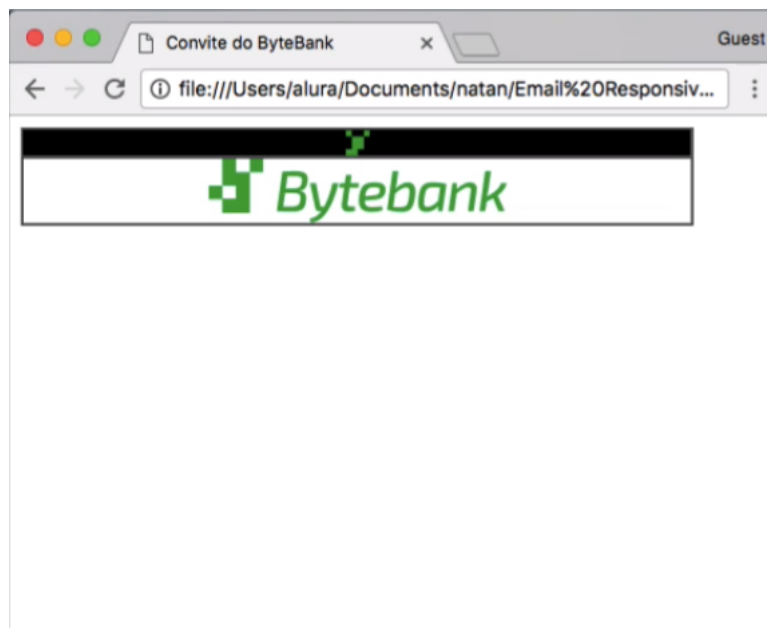
```
<table border="1" width="90%">
  <tr bgcolor="#000000">
    <td align="center" class="cabecalho-logo">
      
  </tr>
  <tr>
    <td align="center">
      
    </td>
  </tr>
</table>
```

Observe que o atributo `align` foi adicionado à nova `td`.

Ao analisarmos o layout final, veremos um espaço entre o logo e a linha do cabeçalho. Para fazer esse ajuste, utilizaremos novamente a classe `cabecalho-logo`:

```
<tr>
  <td align="center" class="cabecalho-logo">
    
  </td>
</tr>
```

Porém, precisamos pensar na situação em que o client e-mail usado pelo usuário não dá suporte para `padding`, o que faria a imagem ser exibida sem o espaçamento.



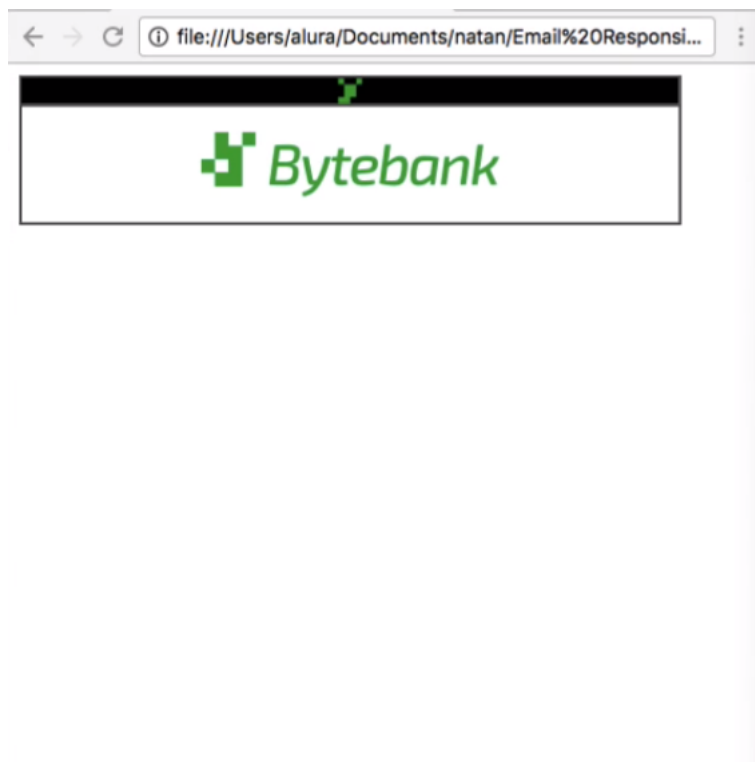
O Sublime possui o comando "command + /" que permite comentar uma linha automaticamente - ele também pode ser usado em outros editores. Iremos utilizá-lo para comentar a linha do `padding` e simular nosso e-mail sem o espaçamento do logo.

```
.cabecalho-logo {  
  /*padding: 15px 0;*/  
}
```

Como não podemos depender do `padding`, um recurso útil para garantirmos o espacinho é saltar uma linha adicionando a tag `<br>`, usada para incluirmos uma quebra de linha.

```
<table border="1" width="90%">  
  <tr bgcolor="#000000">  
    <td align="center" class="cabecalho-logo">  
      <br>  
        
      <br>  
    </td>  
  </tr>  
  <tr>  
    <td align="center" class="cabecalho-logo">  
      <br>  
        
      <br>  
    </td>  
  </tr>
```

Agora, mesmo com o `padding` comentado, a imagem do logo vai receber um espaço.



Com certeza, a tag `<br>` será interpretada por e-mail clients bem antigos. Continuaremos criando o layout, e o próximo passo será adicionar outra `tr`.

```
<tr>  
  <td align="center">  
    <h1>
```

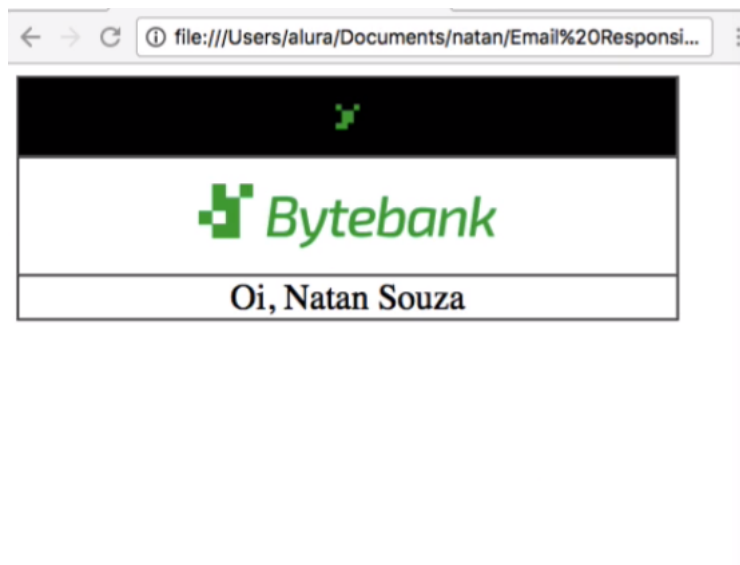
Oi, Natan Souza

```
</h1>
</td>
</tr>
```

Adicionaremos também a tag `<h1>`, para deixar o texto com o corpo maior. No entanto, há o risco do Gmail, por exemplo, aplicar um estilo padrão para `h1`. Evitaremos este tipo de problema utilizando um recurso que está fora das boas práticas (e deixa o meu coração partido), a tag `<span>`, em que incluiremos o `style`.

```
<tr>
  <td align="center">
    <span style="font-size: 25px;">
      Oi, Natan Souza
    </span>
  </td>
</tr>
```

Nossa abordagem foca no objetivo de solucionar os possíveis problemas da criação de um e-mail marketing responsivo. Como adicionamos `<span>`, poderemos apagar a classe `.cabecalho-logo`. No navegador, veremos o e-mail da seguinte maneira:

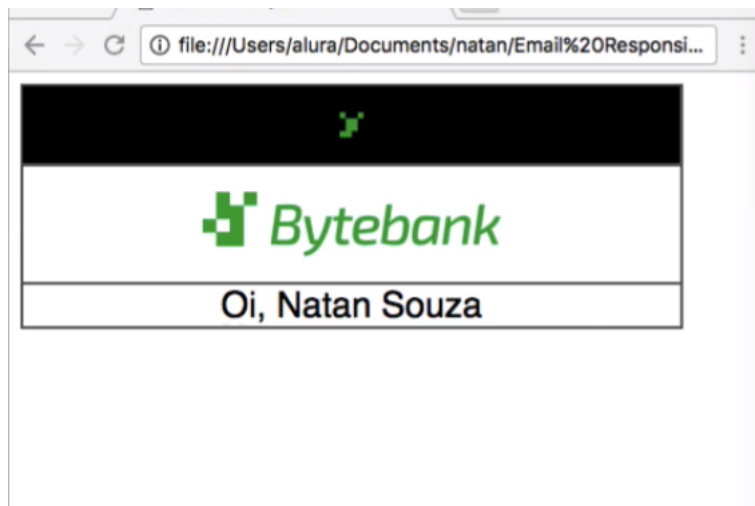


De acordo com o layout final, a fonte usada não possui serifa - os pequenos ornamentos que às vezes estão presentes em alguns tipos de fonte.



Um exemplo de fonte com serifa é a *Times New Roman*, mas se o layout está sem serifa, devemos adotar uma família de fonte semelhante, especificando que ela será `sans-serif`.

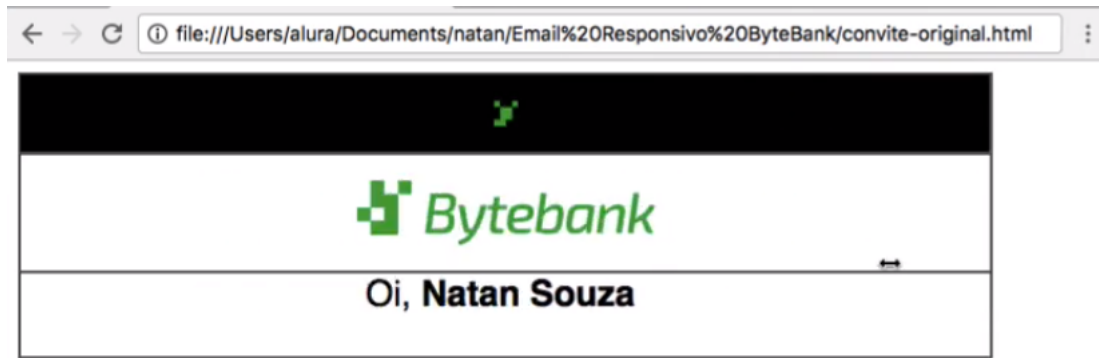
```
<tr>
  <td align="center">
    <span style="font-size: 25px; font-family: sans-serif">
      Oi, Natan Souza
    </span>
  </td>
</tr>
```



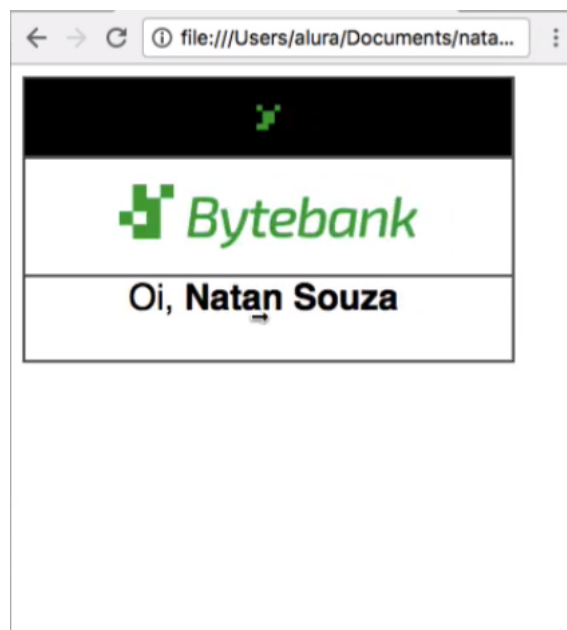
A fonte é exibida sem serifa no navegador. O passo seguinte será deixar o nome do usuário em negrito, o que resolveremos adicionando a tag `<strong>`. Para garantir os espaçamentos da linha, também incluiremos `<br>`, lembrando que nosso foco não está nas boas práticas.

```
<tr>
  <td align="center">
    <span style="font-size: 25px; font-family: sans-serif">
      Oi, <strong>Natan Souza</strong>
      <br><br>
    </span>
  </td>
</tr>
```

```
</span>  
</td>  
</tr>
```



Agora, se testarmos o reajuste do tamanho da tela, nosso e-mail se encontra responsivo.



Nosso e-mail marketing começa a atender qualquer tipo de tela utilizada pelo usuário!