

JavaFX Studies/ Aula05: Css

Agora vamos entender como aplicar estilos CSS no java-FX:

Primeiro crie a pasta Css e dentro dela o nome_do_arquivo.css, para que sua scene receba esse css, você deve fazer a seguinte linha de código:

```
nome_da_sua_scene.getStylesheets().add(getClass().getResource("./css/  
nome_do_arquivo.css").toExternalForm());
```

```
// Cria a cena  
Scene scene = new Scene(root);  
scene.getStylesheets().add(getClass().getResource  
(name: "./css/main.css").toExternalForm());
```

Caso eu tenha mais de uma scene e eu quero usar para todas elas um unico arquivo CSS, eu faço da seguinte forma:

```
(name: "./css/main.css").toExternalForm());  
String css = this.getClass().getResource(name: "./  
css/main.css").toExternalForm();  
scene.getStylesheets().add(css);
```

Agora vamos começar a nossa estilização CSS, primeiro é importante ressaltar que os elementos que vão ser estilizados NECESSITAM ter um ID, para que possam ser indentificados no nosso_arquivo.css, é importante ressaltar como estilizar uma classe de objetos e como estilizar objetos especificos por id: para estilizar uma classe utilize .nome_da_classe{}, e para estilizar algo pelo ID use #ID_do_objeto{}:

```
.root{
```

```
}
```

```
.label{
```

```
}
```

```
#title{
```

```
}
```

Para estilizar objetos fx vc precisa utilizar um prefixo, exemplo:

```
1  .root{
2      -fx-background-color: #111111;
3  }
4
5  .label{
6      -fx-font-size: 20px;
7      -fx-font-family: 'CopperPlate';
8      -fx-text-fill: white;
9      -fx-alignment: center;
10 }
11
12 #title{
13     -fx-font-size: 60px;
14 }
```

Modificando imagens de um button e de uma label pelo CSS:

```

1  .label.linhachegada{
2      -fx-background-image: url("/assets/linha_chegada.png");
3      -fx-pref-width:10px;
4      -fx-pref-height:200px;
5  }
6
7  .button.bandeira {
8      -fx-background-image: url("/assets/bandeira.png");
9      -fx-pref-width:37px;
10     -fx-pref-height:38px;
11 }
12
13 .button.cavalo1 {
14     -fx-background-image: url("/assets/cavalo1.png");
15     -fx-pref-width:55px;
16     -fx-pref-height:30px;
17 }
18
19 .button.cavalo2 {
20     -fx-background-image: url("/assets/cavalo2.png");
21     -fx-pref-width:55px;
22     -fx-pref-height:30px;
23 }
24
25 .button.cavalo3 {

```

Dessa forma conseguimos mudar completamente a imagem que um node vai representar na tela, ele passa a ser igual a imagem, sem possuir seu padrão de Node, e obedecendo apenas aos comandos do CSS.

#JavaFX