**Eduardo Sinico; Ezequiel; Matheus. 1- *SENAI* – SA3 LayoutManager**

**Relatório das Calculadoras em JAVA**

Este é um relatório demostrando como foi feito o layout da interface da calculadora de Programador em Java Swing:

Em Java, "layout" se refere à maneira como os componentes gráficos de uma interface de usuário são organizados em um contêiner, como uma janela ou um painel. O layout determina a disposição e o posicionamento dos elementos visuais em relação uns aos outros. Layouts são essenciais para criar interfaces de usuário atraentes e funcionais em aplicativos Java.

**Tópicos:**

* Calculadora de Programador;
* Calculadora de IMC;
* Calculadora Padrão.
* **Calculadora de Programador:**

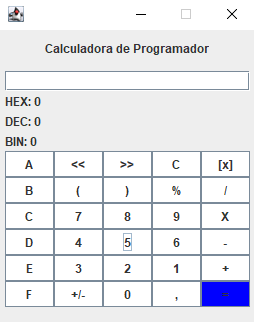
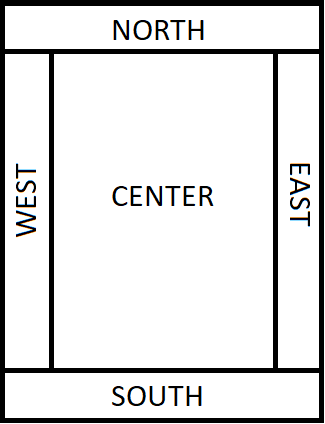


Imagem da Calculadora de Programador

Esses são os três tipos de layout usado no programa:

* **BorderLayout**: BorderLayout é um layout pré-determinado em Java onde ele define as posições de norte, sul, leste, oeste e o centro da página.



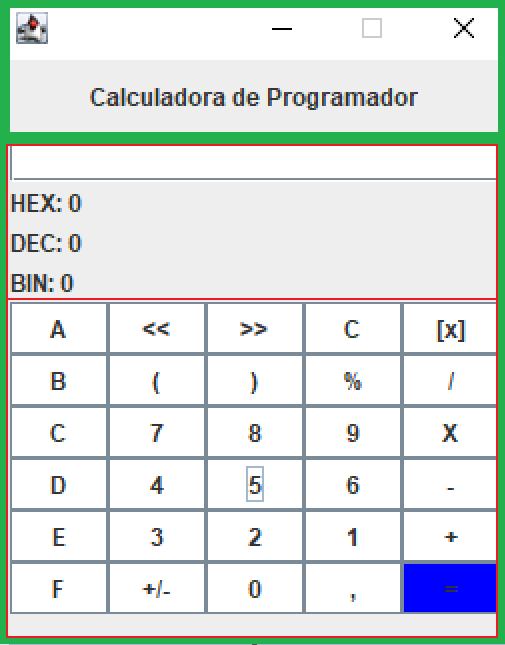
Exemplo de BorderLayout

No projeto:

Primeiramente eu criei um JFrame e adicionei o layout dele para BorderLayout, após isso, adicione dois painéis com suas determinadas posições.

JFrame janela = new JFrame();

janela.setLayout(new BorderLayout());



Exemplo da utilização de BorderLayout na calculadora

No projeto de Java, as cores verde e vermelha representam onde está localizado o BorderLayout. Começando pela parte verde, eu defini dois painéis, uma para o norte:

JPanel painelTop = new JPanel();

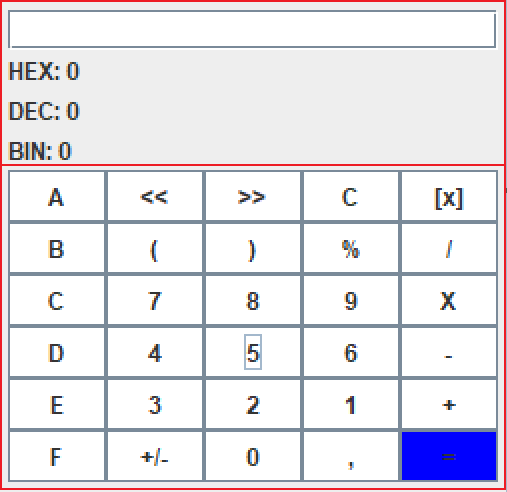
janela.add(painelTop,BorderLayout.NORTH);

E outra para o centro:

JPanel painelCenter = new JPanel();

janela.add(painelCenter,BorderLayout.CENTER);

Indo para o a parte vermelha, eu criei um segundo painel principal, com a mesma configuração de Layout para o JFrame, o BorderLayout:



Neste segundo painel principal, eu criei dois painéis e alinhei eles um de baixo do outro. O TextField do resultado, e os valores HEX, DEC e BIN estão posicionados na parte superior, enquanto, as teclas da calculadora estão posicionadas na parte inferior, utilizando a mesma lógica nos processos anteriores.

* **GridLayout**: GridLayout é outro estilo de Layout bem utilizado em Java.