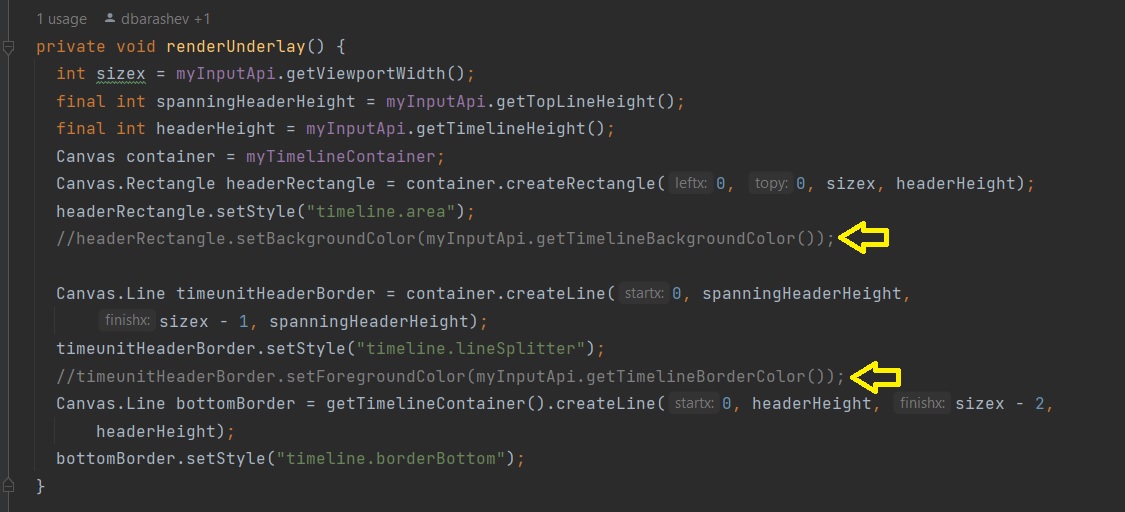
# *Code Smells*

## *Eis 3 Code Smells encontrados no código:*

### Comments

Referenciando, em primeiro lugar, um problema encontrado relativamente aos comentários elaborados no código. Em múltiplas classes, é possível encontrar em comentário vestígios de código antigo (ou possível código novo) por implementar. Por vezes, aparentam ser apenas cópias do código acima.

Pelas linhas 107 e 108 de ganttproject.biz.ganttproject.core.src.main.java.biz.ganttproject.core.chart.scene.TimelineSceneBuilder podemos encontrar, nomeadamente:

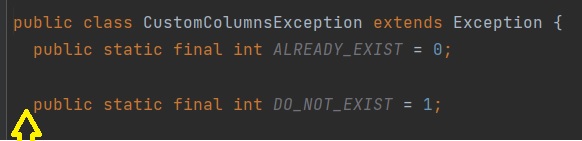


### Dead Code

Foram também encontradas constantes que nunca são utilizadas no código. Estas, naturalmente, também representam um Code Smell, uma vez que podem confundir um novo desenvolvedor. Sendo assim, a melhor solução é retirá-las, até serem realmente necessárias.

Exemplo 1 - Pelas linhas 29 a 31 de ganttproject.ganttproject.src.main.java.biz.ganttproject.customproperty.CustomColumnsException.java, temos:

(*0 usages*)



Exemplo 2 - Pelas linhas 390 a 402 de ganttproject.biz.ganttproject.core.src.main.java.biz.ganttproject.core.chart.canvas.Canvas.java, temos:

(*0 usages*)

### Long Method

Por último, o código nunca deve ter métodos de grande extensão; conhecemos este code smell como Long Method. Um método longo e extensivo torna-se a si mesmo confuso e dificil de ler.

Estes métodos podem ser facilmente decompostos em múltiplos métodos auxiliares de nomes mais fáceis de ler.

Como exemplo, apresento as linhas 35 a 157 de ganttproject.biz.ganttproject.core.src.main.java.org.w3c.util.DateParser.java, com um método de 122 linhas:

