**Experiment Report - 28 - test3\_codehistory**

1. **Summary Table of Errors Found**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Error ID | Line Number | Error Type | Self-Detected? | Peer 1 Found? | Peer 2 Found? |
| E01 | line 23 | Semantic | √ | √ | √ |
| E02 | line 42 | Logic | √ | × | × |
| E03 | line 50 | Semantic | √ | × | × |

Additional Errors Found by Self: 1

Self-Review Detection Rate: 100%

Peer 1 Detection Rate: 33%

Peer 2 Detection Rate: 33%

1. **Source Code**
2. #include "codehistory.h"
3. #include "historylinenumberarea.h"
4. #include <QPainter>
5. #include <QTextBlock>
6. CodeHistory::CodeHistory(QWidget \*parent)
7. : QPlainTextEdit(parent),
8. lineNumberArea(new HistoryLineNumberArea(this)), // 初始化行号区域
9. highlighter(new SyntaxHighlighter(this->document())) // 初始化语法高亮
10. {
11. // 设置编辑器的初始宽度和高亮当前行的功能
12. connect(this, &QPlainTextEdit::blockCountChanged, this, &CodeHistory::updateLineNumberAreaWidth);
13. connect(this, &QPlainTextEdit::updateRequest, this, &CodeHistory::updateLineNumberArea);
14. connect(this, &QPlainTextEdit::cursorPositionChanged, this, &CodeHistory::highlightCurrentLine);
15. updateLineNumberAreaWidth(0);
16. highlightCurrentLine();
17. setEditorFont("Courier",10); // 默认字体设置
18. }
19. void CodeHistory::lineNumberAreaPaintEvent(QPaintEvent \*event)
20. {
21. QPainter painter(lineNumber);
22. painter.fillRect(event->rect(), QColor("#f0f0f0"));
23. QTextBlock block = firstVisibleBlock();
24. int blockNumber = block.blockNumber();
25. int top = (int) blockBoundingGeometry(block).translated(contentOffset()).top();
26. int bottom = top + (int) blockBoundingRect(block).height();
27. while (block.isValid() && top <= event->rect().bottom()) {
28. if (block.isVisible() && bottom >= event->rect().top()) {
29. QString number = QString::number(blockNumber + 1);
30. painter.setPen(Qt::black);
31. painter.drawText(0, top, lineNumberArea->width(), fontMetrics().height(),
32. Qt::AlignRight, number);
33. }
34. block = block.next();
35. top = bottom;
36. bottom = top + (int) blockBoundingRect(block).height();
37. --blockNumber;
38. }
39. }
40. void CodeHistory::resizeEvent(QResizeEvent \*event)
41. {
42. QPlainTextEdit::resizeEvent(event);
43. QRect cr = contentsRect();
44. lineNumberArea->setGeometry(QRect(cr.left(), cr.top(), lineNumberAreaWidth(), cr.width()));
45. }