

```
1
2 #include "HelpPage.h"
3 #include "Game.h"
4 #include <iostream>
5
6 /// <summary>
7 /// \author Peter Lowe
8 /// \version 1.0
9 /// </summary>
10 HelpPage::HelpPage()
11 {
12 }
13
14
15 HelpPage::~HelpPage()
16 {
17 }
18
19 /// <summary>
20 /// @brief setup font and message.
21 ///
22 /// Used to get a reference to the font and setup the
23 /// text used on the help page, a simple block of text
24 /// esc key will return to main menu
25 /// </summary>
26 /// <param name="font">Reference to font loaded in main game</param>
27 void HelpPage::initialise(sf::Font& t_font)
28 {
29     m_font = t_font;
30
31     m_message.setFont(m_font);
32     m_message.setString("Use the arrow keys to move \nMario left or right\nUp      ↗
33     arrow will make him jump\nMove off the stage to exit\nEscape to return to  ↗
34     Menu");
35     m_message.setCharacterSize(24);
36     m_message.setFillColor(sf::Color::White);
37
38     sf::FloatRect textSize = m_message.getGlobalBounds();
39     float xposition = Game::s_screenWidth / 2 - textSize.width / 2;
40     m_message.setPosition(xposition, Game::s_screenHeight * 0.25f);
41     m_exitPressed = false;
42 }
43
44 /// <summary>
45 /// @brief check for esc key.
46 ///
47 /// Check if esc key is pressed and return to main menu
48 /// </summary>
49 /// <param name="event">event passed from process events in main loop</param>
50 void HelpPage::processInput(sf::Event t_event)
51 {
52     if ( sf::Event::KeyPressed == t_event.type)
53     {
```

```
51     if ( sf::Keyboard::Escape == t_event.key.code)
52     {
53         m_exitPressed = true;
54     }
55 }
56 }
57 /// <summary>
58 /// @brief update exit on esc.
59 ///
60 /// check if esc key press event occurred during process events
61 /// indicated by the bool m_exitPressed and if so return to main menu
62 /// always set bool to false;
63 /// </summary>
64 /// <param name="time">update delta time</param>
65 void HelpPage::update(sf::Time t_deltaTime)
66 {
67     if (m_exitPressed)
68     {
69         Game::currentState = GameState::MainMenu;
70     }
71     m_exitPressed = false; // will re-set boolean for next time user is here
72 }
73
74 /// <summary>
75 /// @brief draw the help page.
76 ///
77 /// Draw the sf::Text object message.
78 /// clear and display methods will be called in game.cpp
79 ///
80 /// </summary>
81 /// <param name="window">reference to main render window</param>
82 void HelpPage::render(sf::RenderWindow& t_window)
83 {
84     t_window.draw(m_message);
85 }
```