

```
1  #ifndef GAME_H
2  #define GAME_H
3  // Use the vs debug / release mode effect of defining _DEBUG to turn on/off
4  // features used only in debug mode
5  #ifdef _DEBUG
6  #define TEST_FPS
7  #endif // _DEBUG
8
9
10 #include "Licence.h"
11 #include "SplashScreen.h"
12 #include "MainMenu.h"
13 #include "GamePlay.h"
14 #include "HelpPage.h"
15
16 /// <summary>
17 /// /brief enum used to determine gamestate
18 /// </summary>
19 enum class
20     GameState
21 {
22     None,
23     Licence,
24     Splash,
25     MainMenu,
26     Help,
27     Game
28 };
29
30 /// <summary>
31 /// @author Peter Lowe
32 /// @version 1.0
33 /// @date may 2016
34 /// @brief main game class contains main and run game loop.
35 /// </summary>
36 class Game
37 {
38 public:
39     Game();
40     void run();
41
42     // static properties accessed by all classes
43     // width of the screen in pixels (stored as float? for parameter conversion)
44     static float s_screenWidth; /// screen width in pixels
45     static float s_screenHeight; /// current game state
46     static GameState currentState; // which screen are we on in the game now
47
48 protected:
49     void processEvents();
50     void update(sf::Time);
51     void render();
52     void loadContent();
```

```
53
54
55     sf::Font m_arialFont;    /// font object used for all text
56     sf::RenderWindow m_window; // render window
57 #ifdef TEST_FPS
58     sf::Text x_updateFps;    // text used to display updates per second
59     sf::Text x_drawFps;      // text used to display draw calls per second
60     sf::Time x_secondTime;    // counter used to work out when a second has passed ↗
        counters were reset
61     int x_updateFrameCount;   // updates per second counter
62     int x_drawFrameCount;     // draws per second counter
63
64 #endif // TEST_FPS
65
66     ///instances of classes used to handle the different behaviours needed in the ↗
        complete game
67     /// instance used to run licence screen
68     Licence m_licenceScreen;
69     /// instance used to run splash screen
70     SplashScreen m_splashScreen;
71     /// instance used to run main menu
72     MainMenu m_mainMenu;
73     /// instance used for main gameplay
74     Gameplay m_mainGame;
75     /// instance used to display help page
76     HelpPage m_helpPage;
77 };
78 #endif // !GAME_H
```