

Pong Game

Generated by Doxygen 1.8.6

Tue Feb 25 2020 22:33:53

Contents

Chapter 1

Pong Game

This is an implementation of the classical Pong game.

prerequisites

- Visual Studio (Available online for mac and windows)
- MonoGame

Running

This project can be run by using Visual Studio. Once you have the project open on visual studio, press the start button, which will compile and run the game.

Output

The output should look match the following:

Playing

This is a two player game. The left player can use the `W` and `S` keys to move up and down. The right player can use the arrow keys to move up and down. Users can reset the game by clicking `R` key. They can also quit the game by clicking `Q` or `ESC` key.

Author(s)

- Abel Weldaregay (abelweldaregay@gmail.com)

Chapter 2

Namespace Index

2.1 Namespace List

Here is a list of all namespaces with brief descriptions:

Ping_Pong	??
---------------------------	-------	----

Chapter 3

Hierarchical Index

3.1 Class Hierarchy

This inheritance list is sorted roughly, but not completely, alphabetically:

Game	
Ping_Pong.Game1	??
Ping_Pong.GameObject	??
Ping_Pong.Ball	??
Ping_Pong.Paddle	??
Ping_Pong.Program	??

Chapter 4

Data Structure Index

4.1 Data Structures

Here are the data structures with brief descriptions:

Ping_Pong.Ball	??
Ping_Pong.Game1		
This is the main type for your game	??
Ping_Pong.GameObject	??
Ping_Pong.Paddle	??
Ping_Pong.Program	??

Chapter 5

File Index

5.1 File List

Here is a list of all files with brief descriptions:

Pong Base/ Ball.cs	??
Pong Base/ Game1.cs	??
Pong Base/ GameObject.cs	??
Pong Base/ Paddle.cs	??
Pong Base/ Program.cs	??
Pong Base/obj/x86/Debug/ TemporaryGeneratedFile_036C0B5B-1481-4323-8D20-8F5ADCB23D92.cs .	??
Pong Base/obj/x86/Debug/ TemporaryGeneratedFile_5937a670-0e60-4077-877b-f7221da3dda1.cs . . .	??
Pong Base/obj/x86/Debug/ TemporaryGeneratedFile_E7A71F73-0F8D-4B9B-B56E-8E70B10BC5D3.cs .	??
Pong Base/obj/x86/Release/ TemporaryGeneratedFile_036C0B5B-1481-4323-8D20-8F5ADCB23D92.cs	??
Pong Base/obj/x86/Release/ TemporaryGeneratedFile_5937a670-0e60-4077-877b-f7221da3dda1.cs . .	??
Pong Base/obj/x86/Release/ TemporaryGeneratedFile_E7A71F73-0F8D-4B9B-B56E-8E70B10BC5D3.cs	??
Pong Base/Properties/ AssemblyInfo.cs	??

Chapter 6

Namespace Documentation

6.1 Package Ping_Pong

Data Structures

- class [Ball](#)
- class [Game1](#)

This is the main type for your game

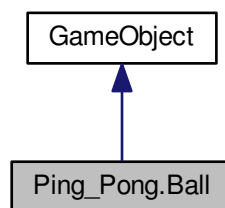
- class [GameObject](#)
- class [Paddle](#)
- class [Program](#)

Chapter 7

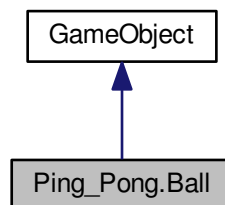
Data Structure Documentation

7.1 Ping_Pong.Ball Class Reference

Inheritance diagram for Ping_Pong.Ball:



Collaboration diagram for Ping_Pong.Ball:



Protected Attributes

- float `m_DX`
- float `m_DY`

Properties

- float [DX](#) [get, set]
- float [DY](#) [get, set]

7.1.1 Detailed Description

Definition at line 8 of file Ball.cs.

7.1.2 Field Documentation

7.1.2.1 float Ping_Pong.Ball.m_DX [protected]

Definition at line 10 of file Ball.cs.

7.1.2.2 float Ping_Pong.Ball.m_DY [protected]

Definition at line 17 of file Ball.cs.

7.1.3 Property Documentation

7.1.3.1 float Ping_Pong.Ball.DX [get], [set]

Definition at line 12 of file Ball.cs.

7.1.3.2 float Ping_Pong.Ball.DY [get], [set]

Definition at line 19 of file Ball.cs.

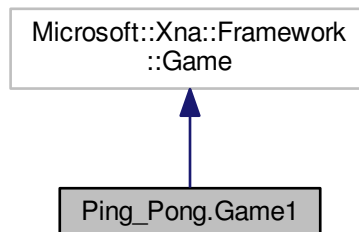
The documentation for this class was generated from the following file:

- Pong Base/[Ball.cs](#)

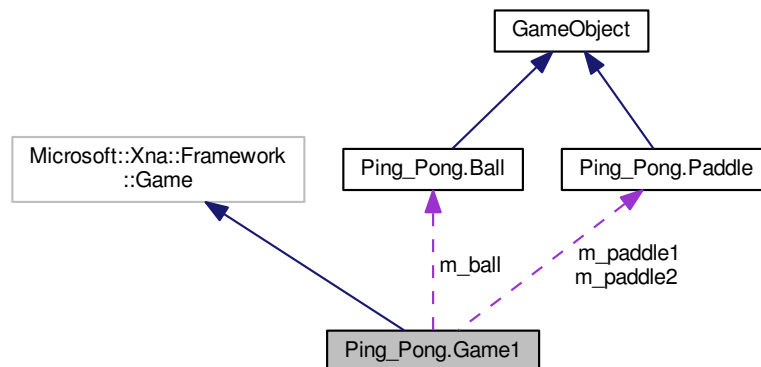
7.2 Ping_Pong.Game1 Class Reference

This is the main type for your game

Inheritance diagram for Ping_Pong.Game1:



Collaboration diagram for Ping_Pong.Game1:



Public Member Functions

- [Game1](#) ()
- void [InitScreen](#) ()
- void [InitGameObjects](#) ()
- void [ResetGame](#) ()
- void [DrawScore](#) (float x, float y, int score)
- void [Render](#) ()

Protected Member Functions

- override void [Initialize](#) ()
Allows the game to perform any initialization it needs to before starting to run. This is where it can query for any required services and load any non-graphic related content. Calling base.Initialize will enumerate through any components and initialize them as well.
- override void [LoadContent](#) ()
LoadContent will be called once per game and is the place to load all of your content.
- void [LoadGameGraphics](#) ()
- override void [UnloadContent](#) ()
UnloadContent will be called once per game and is the place to unload all content.
- override void [Update](#) (GameTime gameTime)
Allows the game to run logic such as updating the world, checking for collisions, gathering input, and playing audio.
- override void [Draw](#) (GameTime gameTime)
This is called when the game should draw itself.

Private Member Functions

- void [MoveBall](#) ()
- bool [CollisionOccurred](#) ()
- void [MovePaddles](#) ()

Private Attributes

- GraphicsDeviceManager [graphics](#)
- SpriteBatch [spriteBatch](#)
- SpriteFont [scoreFont](#)
- SpriteFont [nameFont](#)
- SoundEffect [beep](#)
- Texture2D [background](#)
- Boolean [paused](#)
- KeyboardState [oldState](#)
- Texture2D [pauseOverlay](#)
- Song [backgroundMusic](#)
- int [m_Score1](#) = 0
- int [m_Score2](#) = 0
- Texture2D [m_textureNumbers](#)
- Rectangle[] [m_ScoreRect](#) = null
- Ball [m_ball](#)
- Texture2D [m_textureBall](#)
- Paddle [m_paddle1](#)
- Paddle [m_paddle2](#)
- Texture2D [m_texturePaddle](#)
- const int [SCREEN_WIDTH](#) = 640
- const int [SCREEN_HEIGHT](#) = 480
- const float [PADDLE_STRIDE](#) = 10.0f

7.2.1 Detailed Description

This is the main type for your game

Definition at line 16 of file Game1.cs.

7.2.2 Constructor & Destructor Documentation

7.2.2.1 Ping_Pong.Game1.Game1 () [inline]

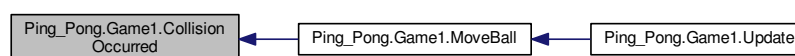
Definition at line 55 of file Game1.cs.

7.2.3 Member Function Documentation

7.2.3.1 bool Ping_Pong.Game1.CollisionOccurred () [inline], [private]

Definition at line 329 of file Game1.cs.

Here is the caller graph for this function:



7.2.3.2 `override void Ping_Pong.Game1.Draw (gameTime gameTime)` `[inline]`, `[protected]`

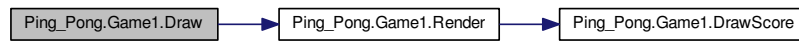
This is called when the game should draw itself.

Parameters

<i>gameTime</i>	Provides a snapshot of timing values.
-----------------	---------------------------------------

Definition at line 438 of file Game1.cs.

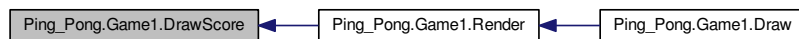
Here is the call graph for this function:



7.2.3.3 void Ping_Pong.Game1.DrawScore (float x, float y, int score) [inline]

Definition at line 453 of file Game1.cs.

Here is the caller graph for this function:



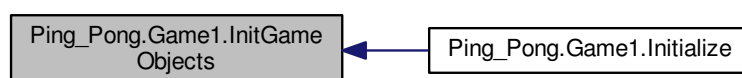
7.2.3.4 void Ping_Pong.Game1.InitGameObjects () [inline]

Definition at line 91 of file Game1.cs.

Here is the call graph for this function:



Here is the caller graph for this function:

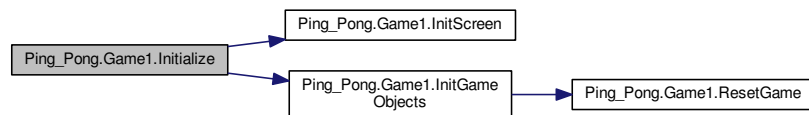


7.2.3.5 override void Ping_Pong.Game1.Initialize () [inline],[protected]

Allows the game to perform any initialization it needs to before starting to run. This is where it can query for any required services and load any non-graphic related content. Calling base.Initialize will enumerate through any components and initialize them as well.

Definition at line 67 of file Game1.cs.

Here is the call graph for this function:



7.2.3.6 void Ping_Pong.Game1.InitScreen () [inline]

Definition at line 81 of file Game1.cs.

Here is the caller graph for this function:

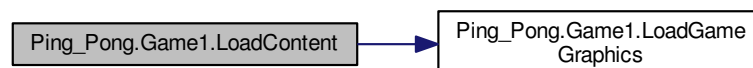


7.2.3.7 override void Ping_Pong.Game1.LoadContent () [inline],[protected]

LoadContent will be called once per game and is the place to load all of your content.

Definition at line 167 of file Game1.cs.

Here is the call graph for this function:



7.2.3.8 void Ping_Pong.Game1.LoadGameGraphics () [inline],[protected]

Definition at line 180 of file Game1.cs.

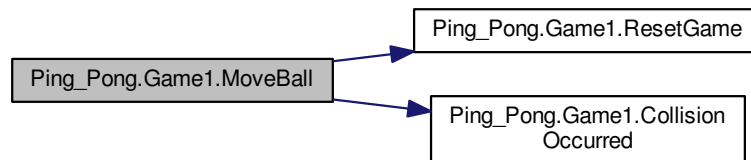
Here is the caller graph for this function:



7.2.3.9 void Ping_Pong.Game1.MoveBall () [inline],[private]

Definition at line 268 of file Game1.cs.

Here is the call graph for this function:



Here is the caller graph for this function:



7.2.3.10 void Ping_Pong.Game1.MovePaddles () [inline],[private]

Definition at line 364 of file Game1.cs.

Here is the caller graph for this function:



7.2.3.11 void Ping_Pong.Game1.Render () [inline]

Definition at line 464 of file Game1.cs.

Here is the call graph for this function:



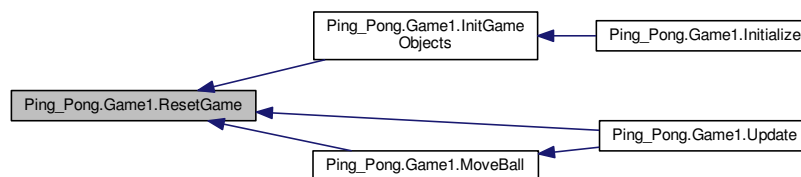
Here is the caller graph for this function:



7.2.3.12 void Ping_Pong.Game1.ResetGame () [inline]

Definition at line 126 of file Game1.cs.

Here is the caller graph for this function:



7.2.3.13 override void Ping_Pong.Game1.UnloadContent () [inline], [protected]

`UnloadContent` will be called once per game and is the place to unload all content.

Definition at line 208 of file Game1.cs.

7.2.3.14 override void Ping_Pong.Game1.Update (gameTime *gameTime*) [inline], [protected]

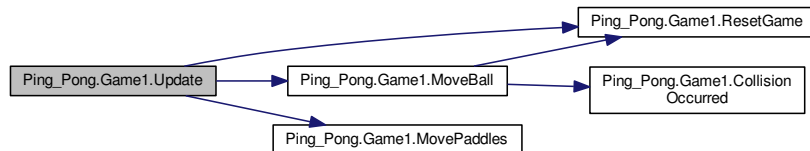
Allows the game to run logic such as updating the world, checking for collisions, gathering input, and playing audio.

Parameters

<i>gameTime</i>	Provides a snapshot of timing values.
-----------------	---------------------------------------

Definition at line 218 of file Game1.cs.

Here is the call graph for this function:



7.2.4 Field Documentation

7.2.4.1 Texture2D Ping_Pong.Game1.background [private]

Definition at line 27 of file Game1.cs.

7.2.4.2 Song Ping_Pong.Game1.backgroundMusic [private]

Definition at line 34 of file Game1.cs.

7.2.4.3 SoundEffect Ping_Pong.Game1.beep [private]

Definition at line 25 of file Game1.cs.

7.2.4.4 GraphicsDeviceManager Ping_Pong.Game1.graphics [private]

Definition at line 18 of file Game1.cs.

7.2.4.5 Ball Ping_Pong.Game1.m_ball [private]

Definition at line 43 of file Game1.cs.

7.2.4.6 Paddle Ping_Pong.Game1.m_paddle1 [private]

Definition at line 47 of file Game1.cs.

7.2.4.7 Paddle Ping_Pong.Game1.m_paddle2 [private]

Definition at line 48 of file Game1.cs.

7.2.4.8 int Ping_Pong.Game1.m_Score1 = 0 [private]

Definition at line 37 of file Game1.cs.

7.2.4.9 `int Ping_Pong.Game1.m_Score2 = 0 [private]`

Definition at line 38 of file Game1.cs.

7.2.4.10 `Rectangle[] Ping_Pong.Game1.m_ScoreRect = null [private]`

Definition at line 40 of file Game1.cs.

7.2.4.11 `Texture2D Ping_Pong.Game1.m_textureBall [private]`

Definition at line 44 of file Game1.cs.

7.2.4.12 `Texture2D Ping_Pong.Game1.m_textureNumbers [private]`

Definition at line 39 of file Game1.cs.

7.2.4.13 `Texture2D Ping_Pong.Game1.m_texturePaddle [private]`

Definition at line 49 of file Game1.cs.

7.2.4.14 `SpriteFont Ping_Pong.Game1.nameFont [private]`

Definition at line 23 of file Game1.cs.

7.2.4.15 `KeyboardState Ping_Pong.Game1.oldState [private]`

Definition at line 30 of file Game1.cs.

7.2.4.16 `const float Ping_Pong.Game1.PADDLE_STRIDE = 10.0f [private]`

Definition at line 361 of file Game1.cs.

7.2.4.17 `Boolean Ping_Pong.Game1.paused [private]`

Definition at line 28 of file Game1.cs.

7.2.4.18 `Texture2D Ping_Pong.Game1.pauseOverlay [private]`

Definition at line 32 of file Game1.cs.

7.2.4.19 `SpriteFont Ping_Pong.Game1.scoreFont [private]`

Definition at line 21 of file Game1.cs.

7.2.4.20 `const int Ping_Pong.Game1.SCREEN_HEIGHT = 480 [private]`

Definition at line 53 of file Game1.cs.

7.2.4.21 `const int Ping_Pong.Game1.SCREEN_WIDTH = 640` `[private]`

Definition at line 52 of file Game1.cs.

7.2.4.22 `SpriteBatch Ping_Pong.Game1.spriteBatch` `[private]`

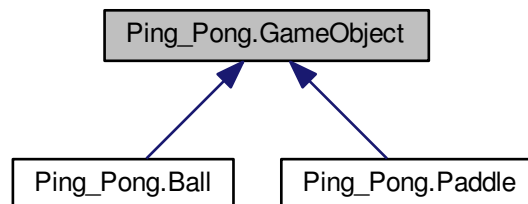
Definition at line 19 of file Game1.cs.

The documentation for this class was generated from the following file:

- Pong Base/[Game1.cs](#)

7.3 Ping_Pong.GameObject Class Reference

Inheritance diagram for Ping_Pong.GameObject:



Protected Attributes

- float [m_X](#)
- float [m_Y](#)
- float [m_Width](#)
- float [m_Height](#)
- object [m_Visual](#) = null

Properties

- float [X](#) `[get, set]`
- float [Y](#) `[get, set]`
- float [Width](#) `[get, set]`
- float [Height](#) `[get, set]`
- Rectangle [Rect](#) `[get]`
- object [Visual](#) `[get, set]`

7.3.1 Detailed Description

Definition at line 9 of file GameObject.cs.

7.3.2 Field Documentation

7.3.2.1 float Ping_Pong.GameObject.m_Height [protected]

Definition at line 32 of file GameObject.cs.

7.3.2.2 object Ping_Pong.GameObject.m_Visual = null [protected]

Definition at line 44 of file GameObject.cs.

7.3.2.3 float Ping_Pong.GameObject.m_Width [protected]

Definition at line 25 of file GameObject.cs.

7.3.2.4 float Ping_Pong.GameObject.m_X [protected]

Definition at line 11 of file GameObject.cs.

7.3.2.5 float Ping_Pong.GameObject.m_Y [protected]

Definition at line 18 of file GameObject.cs.

7.3.3 Property Documentation

7.3.3.1 float Ping_Pong.GameObject.Height [get], [set]

Definition at line 34 of file GameObject.cs.

7.3.3.2 Rectangle Ping_Pong.GameObject.Rect [get]

Definition at line 40 of file GameObject.cs.

7.3.3.3 object Ping_Pong.GameObject.Visual [get], [set]

Definition at line 46 of file GameObject.cs.

7.3.3.4 float Ping_Pong.GameObject.Width [get], [set]

Definition at line 27 of file GameObject.cs.

7.3.3.5 float Ping_Pong.GameObject.X [get], [set]

Definition at line 13 of file GameObject.cs.

7.3.3.6 float Ping_Pong.GameObject.Y [get], [set]

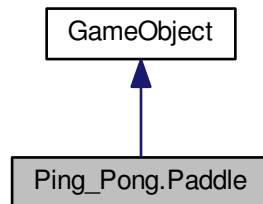
Definition at line 20 of file GameObject.cs.

The documentation for this class was generated from the following file:

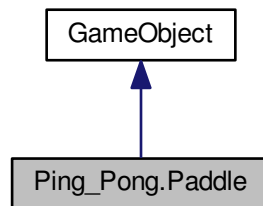
- Pong Base/[GameObject.cs](#)

7.4 Ping_Pong.Paddle Class Reference

Inheritance diagram for Ping_Pong.Paddle:



Collaboration diagram for Ping_Pong.Paddle:



Additional Inherited Members

7.4.1 Detailed Description

Definition at line 8 of file `Paddle.cs`.

The documentation for this class was generated from the following file:

- Pong Base/[Paddle.cs](#)

7.5 Ping_Pong.Program Class Reference

Static Private Member Functions

- static void [Main](#) (string[] args)

The main entry point for the application.

7.5.1 Detailed Description

Definition at line 5 of file Program.cs.

7.5.2 Member Function Documentation

7.5.2.1 `static void Ping_Pong.Program.Main (string[] args) [inline],[static],[private]`

The main entry point for the application.

Definition at line 10 of file Program.cs.

The documentation for this class was generated from the following file:

- Pong Base/[Program.cs](#)

Chapter 8

File Documentation

8.1 Pong Base/Ball.cs File Reference

Data Structures

- class [Ping_Pong.Ball](#)

Namespaces

- package [Ping_Pong](#)

8.2 Pong Base/Game1.cs File Reference

Data Structures

- class [Ping_Pong.Game1](#)

This is the main type for your game

Namespaces

- package [Ping_Pong](#)

8.3 Pong Base/GameObject.cs File Reference

Data Structures

- class [Ping_Pong.GameObject](#)

Namespaces

- package [Ping_Pong](#)

8.4 Pong Base/obj/x86/Debug/Pong Base.csproj.FileListAbsolute.txt File Reference

8.5 Pong Base/obj/x86/Release/Pong Base.csproj.FileListAbsolute.txt File Reference

8.6 Pong Base/obj/x86/Debug/TemporaryGeneratedFile_036C0B5B-1481-4323-8D20-8F5-ADCB23D92.cs File Reference

8.7 Pong Base/obj/x86/Release/TemporaryGeneratedFile_036C0B5B-1481-4323-8D20-8F5ADCB23D92.cs File Reference

8.8 Pong Base/obj/x86/Debug/TemporaryGeneratedFile_5937a670-0e60-4077-877b-f7221da3dda1.cs File Reference

8.9 Pong Base/obj/x86/Release/TemporaryGeneratedFile_5937a670-0e60-4077-877b-f7221da3dda1.cs File Reference

8.10 Pong Base/obj/x86/Debug/TemporaryGeneratedFile_E7A71F73-0F8D-4B9B-B56E-8E70B10BC5D3.cs File Reference

8.11 Pong Base/obj/x86/Release/TemporaryGeneratedFile_E7A71F73-0F8D-4B9B-B56E-8E70B10BC5D3.cs File Reference

8.12 Pong Base/Paddle.cs File Reference

Data Structures

- class [Ping_Pong.Paddle](#)

Namespaces

- package [Ping_Pong](#)

8.13 Pong Base/Program.cs File Reference

Data Structures

- class [Ping_Pong.Program](#)

Namespaces

- package [Ping_Pong](#)

8.14 Pong Base/Properties/AssemblyInfo.cs File Reference

8.15 README.md File Reference