Table of Contents

Namespace SpaceVsInvaders

Classes

CastleConfig

Castle configuration layout

Config

Handles configuration loading

EnemyConfig

Enemy configuration layout

MainGame

This is the main type for your game.

Program

Main program of Space Vs Invaders

TowerConfig

Tower configuration layout

Enums

SceneType

Scene types

Class CastleConfig

Castle configuration layout

Inheritance

System.Object
CastleConfig

Inherited Members

System.Object.Equals(System.Object)
System.Object.Equals(System.Object, System.Object)
System.Object.GetHashCode()
System.Object.GetType()
System.Object.MemberwiseClone()
System.Object.ReferenceEquals(System.Object, System.Object)
System.Object.ToString()

Namespace: SpaceVsInvaders

Assembly: SpaceVsInvaders.dll

Syntax

```
public class CastleConfig
```

Properties

Health

Health of the castle

Declaration

```
public int Health { get; set; }
```

Property Value

TYPE	DESCRIPTION
System.Int32	Health of the castle

UpgradeCost

Upgrade cost of the castle

Declaration

```
public int UpgradeCost { get; set; }
```

Property Value

TYPE	DESCRIPTION
System.Int32	Upgrade cost of the castle

Class Config

Handles configuration loading

Inheritance

System.Object
Config

Inherited Members

System.Object.Equals(System.Object)
System.Object.Equals(System.Object, System.Object)
System.Object.GetHashCode()
System.Object.GetType()
System.Object.MemberwiseClone()
System.Object.ReferenceEquals(System.Object, System.Object)
System.Object.ToString()

Namespace: SpaceVsInvaders

Assembly: SpaceVsInvaders.dll

Syntax

```
public static class Config
```

Methods

GetValue<T>(String)

Retrieves a value from the configuration

Declaration

```
public static T GetValue<T>(string key)
```

Parameters

TYPE	NAME	DESCRIPTION
System.String	key	Key of the config option

Returns

TYPE	DESCRIPTION
T	Config value

Type Parameters

NAME	DESCRIPTION
T	Type of value to return

Initiate(String)

Uses the given file path as the configuration data

Declaration

```
public static void Initiate(string file)
```

Parameters

TYPE	NAME	DESCRIPTION
System.String	file	Config file name, the path relative to DIR/data/

Class EnemyConfig

Enemy configuration layout

Inheritance

System.Object
EnemyConfig

Inherited Members

System.Object.Equals(System.Object)
System.Object.Equals(System.Object, System.Object)
System.Object.GetHashCode()
System.Object.GetType()
System.Object.MemberwiseClone()
System.Object.ReferenceEquals(System.Object, System.Object)
System.Object.ToString()

Namespace: SpaceVsInvaders

Assembly: SpaceVsInvaders.dll

Syntax

```
public class EnemyConfig
```

Properties

Damage

Damage of the enemy

Declaration

```
public int Damage { get; set; }
```

Property Value

TYPE	DESCRIPTION
System.Int32	Damage of the enemy

Health

Health of the enemy

Declaration

```
public int Health { get; set; }
```

Property Value

TYPE	DESCRIPTION
System.Int32	Health of the enemy

Movement

Ticks it takes the enemy to move

Declaration

```
public int Movement { get; set; }
```

Property Value

TYPE	DESCRIPTION
System.Int32	Ticks it takes the enemy to move

TickTime

Ticks it takes the enemy to attack

Declaration

```
public int TickTime { get; set; }
```

Property Value

TYPE	DESCRIPTION
System.Int32	Ticks it takes the enemy to attack

Class MainGame

This is the main type for your game.

Inheritance

System.Object
Microsoft.Xna.Framework.Game
MainGame

Implements

System.IDisposable

Inherited Members

Microsoft.Xna.Framework.Game.Dispose()
Microsoft.Xna.Framework.Game.Dispose(System.Boolean)
Microsoft.Xna.Framework.Game.Exit()
Microsoft.Xna.Framework.Game.ResetElapsedTime()
Microsoft.Xna.Framework.Game.SuppressDraw()
Microsoft.Xna.Framework.Game.RunOneFrame()
Microsoft.Xna.Framework.Game.Run()
Microsoft.Xna.Framework.Game.Run(Microsoft.Xna.Framework.GameRunBehavior)
Microsoft.Xna.Framework.Game.Tick()
Microsoft.Xna.Framework.Game.BeginDraw()
Microsoft.Xna.Framework.Game.EndDraw()
Microsoft.Xna.Framework.Game.BeginRun()
Microsoft.Xna.Framework.Game.EndRun()
Microsoft.Xna.Framework.Game.OnExiting(System.Object, System.EventArgs)
Microsoft.Xna.Framework.Game.OnActivated(System.Object, System.EventArgs)
Microsoft.Xna.Framework.Game.OnDeactivated(System.Object, System.EventArgs)
Microsoft.Xna.Framework.Game.LaunchParameters
Microsoft.Xna.Framework.Game.Components
Microsoft.Xna.Framework.Game.InactiveSleepTime
Microsoft.Xna.Framework.Game.MaxElapsedTime
Microsoft.Xna.Framework.Game.IsActive
Microsoft.Xna.Framework.Game.IsMouseVisible
Microsoft.Xna.Framework.Game.TargetElapsedTime
Microsoft.Xna.Framework.Game.IsFixedTimeStep
Microsoft.Xna.Framework.Game.Services
Microsoft.Xna.Framework.Game.Content
Microsoft.Xna.Framework.Game.GraphicsDevice
Microsoft.Xna.Framework.Game.Window
Microsoft.Xna.Framework.Game.Activated
Microsoft.Xna.Framework.Game.Deactivated
Microsoft.Xna.Framework.Game.Disposed
Microsoft.Xna.Framework.Game.Exiting
System.Object.Equals(System.Object)
System.Object.Equals(System.Object, System.Object)
System.Object.GetHashCode()
System.Object.GetType()
System.Object.MemberwiseClone()
System.Object.ReferenceEquals(System.Object, System.Object)
System.Object.ToString()

Namespace: SpaceVsInvaders

Assembly: SpaceVsInvaders.dll

Syntax

```
public class MainGame : Game, IDisposable
```

Constructors

MainGame()

Constructor of `MainGame`

Declaration

```
public MainGame()
```

Methods

Draw(GameTime)

This is called when the game should draw itself.

Declaration

```
protected override void Draw(GameTime gameTime)
```

Parameters

TYPE	NAME	DESCRIPTION
Microsoft.Xna.Framework.GameTime	gameTime	Provides a snapshot of timing values.

Overrides

Microsoft.Xna.Framework.Game.Draw(Microsoft.Xna.Framework.GameTime)

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.

Declaration

```
protected override void Initialize()
```

Overrides

Microsoft.Xna.Framework.Game.Initialize()

LoadContent()

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

Declaration

```
protected override void LoadContent()
```

Overrides

Microsoft.Xna.Framework.Game.LoadContent()

UnloadContent()

UnloadContent will be called once per game and is the place to unload game-specific content.

Declaration

```
protected override void UnloadContent()
```

Overrides

Microsoft.Xna.Framework.Game.UnloadContent()

Update(GameTime)

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

Declaration

```
protected override void Update(GameTime gameTime)
```

Parameters

TYPE	NAME	DESCRIPTION
Microsoft.Xna.Framework.GameTime	gameTime	Provides a snapshot of timing values.

Overrides

Microsoft.Xna.Framework.Game.Update(Microsoft.Xna.Framework.GameTime)

Implements

System.IDisposable

Class Program

Main program of Space Vs Invaders

Inheritance

System.Object
Program

Inherited Members

System.Object.Equals(System.Object)
System.Object.Equals(System.Object, System.Object)
System.Object.GetHashCode()
System.Object.GetType()
System.Object.MemberwiseClone()
System.Object.ReferenceEquals(System.Object, System.Object)
System.Object.ToString()

Namespace: SpaceVsInvaders

Assembly: SpaceVsInvaders.dll

Syntax

```
public static class Program
```

Enum SceneType

Scene types

Namespace: SpaceVsInvaders

Assembly: SpaceVsInvaders.dll

Syntax

```
public enum SceneType
```

Fields

NAME	DESCRIPTION
Game	Game Scene
Logo	Logo Scene
MainMenu	Main Menu Scene
Pause	Pause Scene

Class TowerConfig

Tower configuration layout

Inheritance

System.Object
TowerConfig

Inherited Members

System.Object.Equals(System.Object)
System.Object.Equals(System.Object, System.Object)
System.Object.GetHashCode()
System.Object.GetType()
System.Object.MemberwiseClone()
System.Object.ReferenceEquals(System.Object, System.Object)
System.Object.ToString()

Namespace: SpaceVsInvaders

Assembly: SpaceVsInvaders.dll

Syntax

```
public class TowerConfig
```

Properties

Cost

Cost of the tower

Declaration

```
public int Cost { get; set; }
```

Property Value

TYPE	DESCRIPTION
System.Int32	Cost of the tower

Health

Health of the tower

Declaration

```
public int Health { get; set; }
```

Property Value

TYPE	DESCRIPTION
System.Int32	Health of the tower

MaxHealth

Max Health of the tower

Declaration

```
public int MaxHealth { get; set; }
```

Property Value

TYPE	DESCRIPTION
System.Int32	Max Health of the tower

Range

Attack range of the tower

Declaration

```
public int Range { get; set; }
```

Property Value

TYPE	DESCRIPTION
System.Int32	Attack range of the tower

TickTime

Ticks it takes the tower to do its action

Declaration

```
public int TickTime { get; set; }
```

Property Value

TYPE	DESCRIPTION
System.Int32	Ticks it takes the tower to do its action

Namespace SpaceVsInvaders.Model

Classes

Coordinate

Describes a coordinate on the board

SVsICastle

Castle's class.

SVsIEventArgs

Class to contain event data.

SVsIModel

The model's class.

SVsIModelException

The model's own kind of exception.

WaveSpawner

Controls the distribution of enemies to the field.

Enums

CatastropheType

Enumeration of possible catasrophes.

EnemyType

Enumeration of possible enemy types.

TowerType

Enumeration of possible tower types.

Enum CatastropheType

Enumeration of possible catasrophes.

Namespace: SpaceVsInvaders.Model

Assembly: SpaceVsInvaders.dll

Syntax

```
public enum CatastropheType
```

Fields

NAME	DESCRIPTION
Asteroid	Incase this type of disaster occurs, a randomly picked tower or a group of enemies standing on the same field will be hurt or exterminated.
Healing	Incase this type of disaster occurs, a randomly picked tower or a group of enemies standing on the same field will be healed.

Class Coordinate

Describes a coordinate on the board

Inheritance

System.Object
Coordinate

Inherited Members

System.Object.Equals(System.Object, System.Object)
System.Object.GetType()
System.Object.MemberwiseClone()
System.Object.ReferenceEquals(System.Object, System.Object)
System.Object.ToString()

Namespace: SpaceVsInvaders.Model

Assembly: SpaceVsInvaders.dll

Syntax

```
public class Coordinate
```

Constructors

Coordinate(Int32, Int32)

constructor of a coordinate

Declaration

```
public Coordinate(int x, int y)
```

Parameters

TYPE	NAME	DESCRIPTION
System.Int32	x	x coordinate
System.Int32	y	y coordinate

Properties

X

Row

Declaration

```
public int X { get; set; }
```

Property Value

TYPE	DESCRIPTION
System.Int32	Row

Y

Coloumn

Declaration

```
public int Y { get; set; }
```

Property Value

TYPE	DESCRIPTION
System.Int32	Coloumn

Methods

Equals(Object)

override of the Equal function

Declaration

```
public override bool Equals(object obj)
```

Parameters

TYPE	NAME	DESCRIPTION
System.Object	obj	the compareable coordinate

Returns

TYPE	DESCRIPTION
System.Boolean	whether the objects are equal

Overrides

System.Object.Equals(System.Object)

GetHashCode()

override of the GetHashCode function

Declaration

```
public override int GetHashCode()
```

Returns

TYPE	DESCRIPTION
System.Int32	a unique hashcode of a coordinate

Overrides

System.Object.GetHashCode()

Enum EnemyType

Enumeration of possible enemy types.

Namespace: SpaceVsInvaders.Model

Assembly: SpaceVsInvaders.dll

Syntax

```
public enum EnemyType
```

Fields

NAME	DESCRIPTION
Buff	Slow but harmful enemy.
Normal	Avarage enemy.
Speedy	Fast but less harmful enemy.

Extension Methods

ContentLoader.GetTexture()

Class SVsICastle

Castle's class.

Inheritance

System.Object
SVsICastle

Inherited Members

System.Object.Equals(System.Object)
System.Object.Equals(System.Object, System.Object)
System.Object.GetHashCode()
System.Object.GetType()
System.Object.MemberwiseClone()
System.Object.ReferenceEquals(System.Object, System.Object)
System.Object.ToString()

Namespace: SpaceVsInvaders.Model

Assembly: SpaceVsInvaders.dll

Syntax

```
public class SVsICastle
```

Constructors

SVsICastle()

Castle constructor.

Declaration

```
public SVsICastle()
```

Properties

CurrentUpgradeCost

Calculates the current upgrade cost of the castle.

Declaration

```
public int CurrentUpgradeCost { get; }
```

Property Value

TYPE	DESCRIPTION
System.Int32	CurrentUpgradeCost

Health

Castle's health

Declaration

```
public int Health { get; set; }
```

Property Value

TYPE	DESCRIPTION
System.Int32	Health

Level

Castle's current level

Declaration

```
public int Level { get; set; }
```

Property Value

TYPE	DESCRIPTION
System.Int32	Level

UpgradeCost

Castle's original upgrade cost

Declaration

```
public int UpgradeCost { get; set; }
```

Property Value

TYPE	DESCRIPTION
System.Int32	UpgradeCost

Methods

Upgrade()

Upgrade castle

Declaration

```
public void Upgrade()
```

Class SVsIEventArgs

Class to contain event data.

Inheritance

System.Object
System.EventArgs
SVsIEventArgs

Inherited Members

System.EventArgs.Empty
System.Object.Equals(System.Object)
System.Object.Equals(System.Object, System.Object)
System.Object.GetHashCode()
System.Object.GetType()
System.Object.MemberwiseClone()
System.Object.ReferenceEquals(System.Object, System.Object)
System.Object.ToString()

Namespace: SpaceVsInvaders.Model

Assembly: SpaceVsInvaders.dll

Syntax

```
public class SVsIEventArgs : EventArgs
```

Constructors

SVsIEventArgs(Boolean)

EventArgs for setting 'GameOver' true.

Declaration

```
public SVsIEventArgs(bool gameover)
```

Parameters

TYPE	NAME	DESCRIPTION
System.Boolean	gameover	

SVsIEventArgs(Int32, Int32)

EventArgs for catastrophes, tower destroyals, enemy exterminations.

Declaration

```
public SVsIEventArgs(int whereX, int whereY)
```

Parameters

TYPE	NAME	DESCRIPTION
System.Int32	whereX	The row where the event occurs.
System.Int32	whereY	The coloumn where the event occurs.

SVsIEventArgs(Int32, Int32, EnemyType)

EventArgs for the enemy's entry to the castle.

Declaration

```
public SVsIEventArgs(int whereX, int whereY, EnemyType type)
```

Parameters

TYPE	NAME	DESCRIPTION
System.Int32	whereX	The row where it was before the entry.
System.Int32	whereY	The coloumn where it was before the entry.
EnemyType	type	The type of the enemy.

SVsIEventArgs(Int32, Int32, Int32, Int32)

EventArgs for movement and shot.

Declaration

```
public SVsIEventArgs(int fromX, int fromY, int toX, int toY)
```

Parameters

TYPE	NAME	DESCRIPTION
System.Int32	fromX	The starting row of the action.
System.Int32	fromY	The starting coloumn of the action.
System.Int32	toX	The ending row of the action.
System.Int32	toY	The ending coloumn of the action.

Properties

From

'From' coordinates of an event.

Declaration

```
public Coordinate From { get; }
```

Property Value

TYPE	DESCRIPTION
Coordinate	From

GameOver

Boolean that represents whether the game is over or not.

Declaration

```
public bool GameOver { get; }
```

Property Value

TYPE	DESCRIPTION
System.Boolean	GameOver

To

'To' coordinates of an event.

Declaration

```
public Coordinate To { get; }
```

Property Value

TYPE	DESCRIPTION
Coordinate	To

Type

Type of the enemy.

Declaration

```
public EnemyType Type { get; }
```

Property Value

TYPE	DESCRIPTION
EnemyType	EnemyType

Where

'Where' coordinates of an event.

Declaration

```
public Coordinate Where { get; }
```

Property Value

TYPE	DESCRIPTION
Coordinate	

Class SVsIModel

The model's class.

Inheritance

System.Object
SVsIModel

Inherited Members

System.Object.Equals(System.Object)
System.Object.Equals(System.Object, System.Object)
System.Object.GetHashCode()
System.Object.GetType()
System.Object.MemberwiseClone()
System.Object.ReferenceEquals(System.Object, System.Object)
System.Object.ToString()

Namespace: SpaceVsInvaders.Model

Assembly: SpaceVsInvaders.dll

Syntax

```
public class SVsIModel
```

Constructors

SVsIModel()

Model's constructor.

Declaration

```
public SVsIModel()
```

Fields

Castle

Player's castle.

Declaration

```
public SVsICastle Castle
```

Field Value

TYPE	DESCRIPTION
SVsICastle	

Enemies

Board of enemy lists.

Declaration

```
public List<SVsIEnemy>[, ] Enemies
```

Field Value

TYPE	DESCRIPTION
System.Collections.Generic.List<SVsIEnemy>[,]	

Towers

Board of towers.

Declaration

```
public SVsITower[, ] Towers
```

Field Value

TYPE	DESCRIPTION
SVsITower[,]	

WS

WaveSpawner.

Declaration

```
public WaveSpawner WS
```

Field Value

TYPE	DESCRIPTION
WaveSpawner	

Properties

Cols

Width of the gametable.

Declaration

```
public int Cols { get; set; }
```

Property Value

TYPE	DESCRIPTION
System.Int32	Cols

Difficulty

Difficulty of the game.

Declaration

```
public int Difficulty { get; }
```

Property Value

TYPE	DESCRIPTION
System.Int32	Difficulty

IsCatastrophe

Catasrophes can occur.

Declaration

```
public bool IsCatastrophe { get; set; }
```

Property Value

TYPE	DESCRIPTION
System.Boolean	

IsGameOver

Is game over.

Declaration

```
public bool IsGameOver { get; }
```

Property Value

TYPE	DESCRIPTION
System.Boolean	IsGameOver

IsSpawningEnemies

Wavespawner is spawning enemies.

Declaration

```
public bool IsSpawningEnemies { get; set; }
```

Property Value

TYPE	DESCRIPTION
System.Boolean	IsSpawningEnemies

Money

Players current amount of money.

Declaration

```
public int Money { get; set; }
```

Property Value

TYPE	DESCRIPTION
System.Int32	Money

ReinforceTimes

Interval of the enemies' reinforcement.

Declaration

```
public int ReinforceTimes { get; }
```

Property Value

TYPE	DESCRIPTION
System.Int32	ReinforceTimes

Rows

Heigth of the gametable.

Declaration

```
public int Rows { get; set; }
```

Property Value

TYPE	DESCRIPTION
System.Int32	Rows

SecondsElapsed

Seconds elapsed since the game has started.

Declaration

```
public int SecondsElapsed { get; }
```

Property Value

TYPE	DESCRIPTION
System.Int32	SecondsElapsed

TowerCounter

Tower counter.

Declaration

```
public int TowerCounter { get; }
```

Property Value

TYPE	DESCRIPTION
System.Int32	TowerCounter

TowerUpdates

Amount of tower updates.

Declaration

```
public int TowerUpdates { get; }
```

Property Value

TYPE	DESCRIPTION
System.Int32	TowerUpdates

Methods

Catastrophe()

Generates catastrophes.

Declaration

```
public void Catastrophe()
```

generateCoordinates()

Generates a random coordinate on the board

Declaration

```
public Coordinate generateCoordinates()
```

Returns

TYPE	DESCRIPTION
Coordinate	A random coordinate on the board

HandleAsteroidCatastrophe(Int32, Int32, Int32)

Handles Damage Catastrophe

Declaration

```
public void HandleAsteroidCatastrophe(int i, int j, int dmg)
```

Parameters

TYPE	NAME	DESCRIPTION
System.Int32	i	Affected row

TYPE	NAME	DESCRIPTION
System.Int32	j	Affected col
System.Int32	dmg	Damage to be dealt

HandleHealingCatastrophe(Int32, Int32, Int32)

Handles Heal Catastrophe

Declaration

```
public void HandleHealingCatastrophe(int i, int j, int heal)
```

Parameters

TYPE	NAME	DESCRIPTION
System.Int32	i	Affected row
System.Int32	j	Affected col
System.Int32	heal	Healing to be dealt

HandleHealTower(Int32, Int32)

Increases other towers' healthpoints near itself.

Declaration

```
public void HandleHealTower(int row, int col)
```

Parameters

TYPE	NAME	DESCRIPTION
System.Int32	row	
System.Int32	col	

HandleTick()

Handler of time passing.

Declaration

```
public void HandleTick()
```

NewGame(Int32, Int32)

Stars a new game.

Declaration

```
public void NewGame(int rows, int cols)
```

Parameters

TYPE	NAME	DESCRIPTION
System.Int32	rows	Number of rows in the new game.
System.Int32	cols	Number of coloumns in the new game.

onAsteroidCatastrophe(Int32, Int32)

Asteroid catasrophe event sender.

Declaration

```
public void onAsteroidCatastrophe(int whereX, int whereY)
```

Parameters

TYPE	NAME	DESCRIPTION
System.Int32	whereX	The row where it occurs.
System.Int32	whereY	The coloumn where it occurs.

onEnemyDead(Int32, Int32)

Enemy exterminated event sender.

Declaration

```
public void onEnemyDead(int whereX, int whereY)
```

Parameters

TYPE	NAME	DESCRIPTION
System.Int32	whereX	The row where it has been exterminated.
System.Int32	whereY	The coloumn where it has been exterminated.

onEnemyMoved(Int32, Int32, Int32, Int32)

Enemy movement event sender.

Declaration

```
public void onEnemyMoved(int fromX, int fromY, int toX, int toY)
```

Parameters

TYPE	NAME	DESCRIPTION
System.Int32	fromX	The row it moved from.
System.Int32	fromY	The coloumn it moved from.
System.Int32	toX	The row it moved to.
System.Int32	toY	The coloumn it moved to.

onEnemyMovedToCastle(Int32, Int32, EnemyType)

Enemy moved to castle event sender.

Declaration

```
public void onEnemyMovedToCastle(int whereX, int whereY, EnemyType type)
```

Parameters

TYPE	NAME	DESCRIPTION
System.Int32	whereX	The row it moved to the castle from.
System.Int32	whereY	The coloumn it moved to the castle from.
EnemyType	type	The type of the enemy that moved to the castle.

onGameOver(Boolean)

Game over event sender.

Declaration

```
public void onGameOver(bool victory)
```

Parameters

TYPE	NAME	DESCRIPTION
System.Boolean	victory	Whether the game has been won or lost.

onHealingCatastrophe(Int32, Int32)

Healing catasrophe event sender.

Declaration

```
public void onHealingCatastrophe(int whereX, int whereY)
```

Parameters

TYPE	NAME	DESCRIPTION
System.Int32	whereX	The row where it occurs.
System.Int32	whereY	The coloumn where it occurs.

onTowerDestroyed(Int32, Int32)

Tower destroyed event sender.

Declaration

```
public void onTowerDestroyed(int whereX, int whereY)
```

Parameters

TYPE	NAME	DESCRIPTION
System.Int32	whereX	The row where it has been destroyed.
System.Int32	whereY	The coloumn where it has been destroyed.

onTowerHasAttacked(Int32, Int32, Int32, Int32)

Tower attacked event sender.

Declaration

```
public void onTowerHasAttacked(int fromX, int fromY, int toX, int toY)
```

Parameters

TYPE	NAME	DESCRIPTION
System.Int32	fromX	The row it has attacked from.
System.Int32	fromY	The coloumn it has attacked from.
System.Int32	toX	The row of the enemy it has attacked.
System.Int32	toY	The coloumn of the enemy it has attacked.

PlaceEnemy(Int32, Int32, EnemyType)

Places given type of enemy on the board.

Declaration

```
public void PlaceEnemy(int row, int col, EnemyType enemyType)
```

Parameters

TYPE	NAME	DESCRIPTION
System.Int32	row	The row where the enemy will be placed.
System.Int32	col	The coloumn where the enemy will be placed.
EnemyType	enemyType	The type of the enemy that will be placed.

PlaceTower(Int32, Int32, TowerType)

Places a tower of the selected type if the player has enough money for it.

Declaration

```
public void PlaceTower(int row, int col, TowerType type)
```

Parameters

TYPE	NAME	DESCRIPTION
System.Int32	row	The row where the tower will be placed.
System.Int32	col	The coloumn where the tower will be placed.
TowerType	type	The type of the tower that will be placed.

SellTower(Int32, Int32)

Sells the selected tower and increases the player's money.

Declaration

```
public void SellTower(int row, int col)
```

Parameters

TYPE	NAME	DESCRIPTION
System.Int32	row	The row of the tower that will be sold.

TYPE	NAME	DESCRIPTION
System.Int32	col	The coloumn of the tower that will be sold.

UpgradeCastle()

Upgrades the castle if the player has enough money for it. Returns 'true' if the upgrade was successful.

Declaration

```
public void UpgradeCastle()
```

UpgradeTower(Int32, Int32)

Upgrades the chosen tower if the player has enough money for it.

Declaration

```
public void UpgradeTower(int row, int col)
```

Parameters

TYPE	NAME	DESCRIPTION
System.Int32	row	The row of the tower that will be upgraded.
System.Int32	col	The coloumn of the tower that will be upgraded.

WhichEnemy(EnemyType, Int32)

Places given type of enemy on the gameboard.

Declaration

```
public void WhichEnemy(EnemyType type, int i)
```

Parameters

TYPE	NAME	DESCRIPTION
EnemyType	type	
System.Int32	i	

Events

AsteroidCatastrophe

Asteroid catastrophe occured event.

Declaration

```
public event EventHandler<SVsIEventArgs> AsteroidCatastrophe
```

Event Type

TYPE	DESCRIPTION
System.EventHandler<SVsIEventArgs>	

EnemyDead

Enemy exterminated event.

Declaration

```
public event EventHandler<SVsIEventArgs> EnemyDead
```

Event Type

TYPE	DESCRIPTION
System.EventHandler<SVsIEventArgs>	

EnemyMoved

Enemy movement event.

Declaration

```
public event EventHandler<SVsIEventArgs> EnemyMoved
```

Event Type

TYPE	DESCRIPTION
System.EventHandler<SVsIEventArgs>	

EnemyMovedToCastle

Enemy entered the castle event.

Declaration

```
public event EventHandler<SVsIEventArgs> EnemyMovedToCastle
```

Event Type

TYPE	DESCRIPTION
System.EventHandler<SVsIEventArgs>	

GameOver

Game over event.

Declaration

```
public event EventHandler<bool> GameOver
```

Event Type

TYPE	DESCRIPTION
System.EventHandler<System.Boolean>	

HealingCatastrophe

Healing catastrophe occured event.

Declaration

```
public event EventHandler<SVsIEventArgs> HealingCatastrophe
```

Event Type

TYPE	DESCRIPTION
System.EventHandler<SVsIEventArgs>	

TowerDestroyed

Tower destroyed event.

Declaration

```
public event EventHandler<SVsIEventArgs> TowerDestroyed
```

Event Type

TYPE	DESCRIPTION
System.EventHandler<SVsIEventArgs>	

TowerHasAttacked

Tower attack event.

Declaration

```
public event EventHandler<SVsIEventArgs> TowerHasAttacked
```

Event Type

TYPE	DESCRIPTION
System.EventHandler<SVsIEventArgs>	

Class SVsIModelException

The model's own kind of exception.

Inheritance

System.Object
System.Exception
SVsIModelException

Implements

System.Runtime.Serialization.ISerializable

Inherited Members

System.Exception.GetBaseException()
System.Exception.GetObjectData(System.Runtime.Serialization.SerializationInfo, System.Runtime.Serialization.StreamingContext)
System.Exception.GetType()
System.Exception.ToString()
System.Exception.Data
System.Exception.HelpLink
System.Exception.HResult
System.Exception.InnerException
System.Exception.Message
System.Exception.Source
System.Exception.StackTrace
System.Exception.TargetSite
System.Exception.SerializeObjectState
System.Object.Equals(System.Object)
System.Object.Equals(System.Object, System.Object)
System.Object.GetHashCode()
System.Object.MemberwiseClone()
System.Object.ReferenceEquals(System.Object, System.Object)

Namespace: SpaceVsInvaders.Model

Assembly: SpaceVsInvaders.dll

Syntax

```
public class SVsIModelException : Exception, ISerializable
```

Constructors

SVsIModelException(String)

The exception's constuctor.

Declaration

```
public SVsIModelException(string message)
```

Parameters

TYPE	NAME	DESCRIPTION
System.String	message	Custom message reasoning the exception's thrown.

Implements

System.Runtime.Serialization.ISerializable

Enum TowerType

Enumeration of possible tower types.

Namespace: SpaceVsInvaders.Model

Assembly: SpaceVsInvaders.dll

Syntax

```
public enum TowerType
```

Fields

NAME	DESCRIPTION
Damage	Tower that can damage the enemy.
Gold	Tower that produces money for the player.
Heal	Tower that can heal other towers.

Extension Methods

ContentLoader.GetTexture()

Class WaveSpawner

Controls the distribution of enemies to the field.

Inheritance

System.Object
WaveSpawner

Inherited Members

System.Object.Equals(System.Object)
System.Object.Equals(System.Object, System.Object)
System.Object.GetHashCode()
System.Object.GetType()
System.Object.MemberwiseClone()
System.Object.ReferenceEquals(System.Object, System.Object)
System.Object.ToString()

Namespace: SpaceVsInvaders.Model

Assembly: SpaceVsInvaders.dll

Syntax

```
public class WaveSpawner
```

Constructors

WaveSpawner()

Constructor of the class WaveSpawner

Declaration

```
public WaveSpawner()
```

Methods

AreEnemiesLeft()

Determines whether there are enemies left in the spawner's queue

Declaration

```
public bool AreEnemiesLeft()
```

Returns

TYPE	DESCRIPTION
System.Boolean	Bool

GetSpawnedEnemies(Int32)

Provides as many enemies, as the count of coloumns of the game

Declaration

```
public List<EnemyType> GetSpawnedEnemies(int n)
```

Parameters

TYPE	NAME	DESCRIPTION
System.Int32	n	Number of coloumns

Returns

TYPE	DESCRIPTION
System.Collections.Generic.List<EnemyType>	List of EnemyTypes

SpawnEnemies(Int32, Int32, Int32, Int32, Int32)

Creates the EnemyTypes to be spawned

Declaration

```
public void SpawnEnemies(int time, int col, int towercount, int towerupdates, int threeminutes)
```

Parameters

TYPE	NAME	DESCRIPTION
System.Int32	time	Time elapsed ingame
System.Int32	col	Number of cols
System.Int32	towercount	Number of towers
System.Int32	towerupdates	Count of updates of the towers
System.Int32	threeminutes	How many times has 3 minutes passed

Namespace SpaceVsInvaders.Model.Enemies

Classes

SVsIBuffEnemy

Derived class of enemy that is slow but harmful.

SVsIEnemy

Base class of enemies.

SVsINormalEnemy

Derived class of normal enemy.

SVsISpeedyEnemy

Derived class of enemy that is speedy but less harmful.

Class SVsIBuffEnemy

Derived class of enemy that is slow but harmful.

Inheritance

System.Object

SVsIEnemy

SVsIBuffEnemy

Inherited Members

SVsIEnemy.Health

SVsIEnemy.MaxHealth

SVsIEnemy.Movement

SVsIEnemy.Damage

SVsIEnemy.TickTime

SVsIEnemy.CoolDown

System.Object.Equals(System.Object)

System.Object.Equals(System.Object, System.Object)

System.Object.GetHashCode()

System.Object.GetType()

System.Object.MemberwiseClone()

System.Object.ReferenceEquals(System.Object, System.Object)

System.Object.ToString()

Namespace: SpaceVsInvaders.Model.Enemies

Assembly: SpaceVsInvaders.dll

Syntax

```
public class SVsIBuffEnemy : SVsIEnemy
```

Constructors

SVsIBuffEnemy()

SVsIBuffEnemy constructor.

Declaration

```
public SVsIBuffEnemy()
```

Class SVsIEnemy

Base class of enemies.

Inheritance

System.Object
SVsIEnemy
SVsIBuffEnemy
SVsINormalEnemy
SVsISpeedyEnemy

Inherited Members

System.Object.Equals(System.Object)
System.Object.Equals(System.Object, System.Object)
System.Object.GetHashCode()
System.Object.GetType()
System.Object.MemberwiseClone()
System.Object.ReferenceEquals(System.Object, System.Object)
System.Object.ToString()

Namespace: SpaceVsInvaders.Model.Enemies

Assembly: SpaceVsInvaders.dll

Syntax

```
public class SVsIEnemy
```

Constructors

SVsIEnemy()

SVsIEnemy constructor - public.

Declaration

```
public SVsIEnemy()
```

SVsIEnemy(EnemyConfig)

SVsIEnemy constructor - protected.

Declaration

```
protected SVsIEnemy(EnemyConfig conf)
```

Parameters

TYPE	NAME	DESCRIPTION
EnemyConfig	conf	

Properties

CoolDown

How many ticks are left until the next attack.

Declaration

```
public int CoolDown { get; set; }
```

Property Value

TYPE	DESCRIPTION
System.Int32	CoolDown

Damage

How much damage the enemy causes.

Declaration

```
public int Damage { get; set; }
```

Property Value

TYPE	DESCRIPTION
System.Int32	Damage

Health

The enemy's current health.

Declaration

```
public int Health { get; set; }
```

Property Value

TYPE	DESCRIPTION
System.Int32	Health

MaxHealth

The maximum value of the enemy's health.

Declaration

```
public int MaxHealth { get; set; }
```

Property Value

TYPE	DESCRIPTION
System.Int32	MaxHealth

Movement

Speed of the enemy.

Declaration

```
public int Movement { get; set; }
```

Property Value

TYPE	DESCRIPTION
System.Int32	Movement

TickTime

How often the enemy attacks.

Declaration

```
public int TickTime { get; set; }
```

Property Value

TYPE	DESCRIPTION
System.Int32	TickTime

Class SVsINormalEnemy

Derived class of normal enemy.

Inheritance

System.Object
SVsIEnemy
SVsINormalEnemy

Inherited Members

SVsIEnemy.Health
SVsIEnemy.MaxHealth
SVsIEnemy.Movement
SVsIEnemy.Damage
SVsIEnemy.TickTime
SVsIEnemy.CoolDown
System.Object.Equals(System.Object)
System.Object.Equals(System.Object, System.Object)
System.Object.GetHashCode()
System.Object.GetType()
System.Object.MemberwiseClone()
System.Object.ReferenceEquals(System.Object, System.Object)
System.Object.ToString()

Namespace: SpaceVsInvaders.Model.Enemies

Assembly: SpaceVsInvaders.dll

Syntax

```
public class SVsINormalEnemy : SVsIEnemy
```

Constructors

SVsINormalEnemy()

SVsINormalEnemy constructor

Declaration

```
public SVsINormalEnemy()
```

Class SVsISpeedyEnemy

Derived class of enemy that is speedy but less harmful.

Inheritance

System.Object

SVsIEnemy

SVsISpeedyEnemy

Inherited Members

SVsIEnemy.Health

SVsIEnemy.MaxHealth

SVsIEnemy.Movement

SVsIEnemy.Damage

SVsIEnemy.TickTime

SVsIEnemy.CoolDown

System.Object.Equals(System.Object)

System.Object.Equals(System.Object, System.Object)

System.Object.GetHashCode()

System.Object.GetType()

System.Object.MemberwiseClone()

System.Object.ReferenceEquals(System.Object, System.Object)

System.Object.ToString()

Namespace: SpaceVsInvaders.Model.Enemies

Assembly: SpaceVsInvaders.dll

Syntax

```
public class SVsISpeedyEnemy : SVsIEnemy
```

Constructors

SVsISpeedyEnemy()

SVsISpeedyEnemy constructor

Declaration

```
public SVsISpeedyEnemy()
```

Namespace SpaceVsInvaders.Model.Towers

Classes

SVsIDamageTower

Derived class of tower that can damage the enemy.

SVsIGoldTower

Derived class of tower that can produce money for the player.

SVsIHealTower

Derived class of tower that can heal other towers.

SVsITower

Base class of towers.

Class SVsIDamageTower

Derived class of tower that can damage the enemy.

Inheritance

System.Object
SVsITower
SVsIDamageTower

Inherited Members

SVsITower.Health
SVsITower.MaxHealth
SVsITower.Cost
SVsITower.Level
SVsITower.TickTime
SVsITower.CoolDown
SVsITower.Range
SVsITower.UpgradeCost
SVsITower.SellCost
SVsITower.Upgrade()
System.Object.Equals(System.Object)
System.Object.Equals(System.Object, System.Object)
System.Object.GetHashCode()
System.Object.GetType()
System.Object.MemberwiseClone()
System.Object.ReferenceEquals(System.Object, System.Object)
System.Object.ToString()

Namespace: SpaceVsInvaders.Model.Towers

Assembly: SpaceVsInvaders.dll

Syntax

```
public class SVsIDamageTower : SVsITower
```

Constructors

SVsIDamageTower()

SVsIDamageTower constructor.

Declaration

```
public SVsIDamageTower()
```

Methods

Damage()

Defines how much damage it causes.

Declaration

```
public int Damage()
```

Returns

TYPE	DESCRIPTION
System.Int32	Its damage

Class SVsIGoldTower

Derived class of tower that can produce money for the player.

Inheritance

System.Object
SVsITower
SVsIGoldTower

Inherited Members

SVsITower.Health
SVsITower.MaxHealth
SVsITower.Cost
SVsITower.Level
SVsITower.TickTime
SVsITower.CoolDown
SVsITower.Range
SVsITower.UpgradeCost
SVsITower.SellCost
SVsITower.Upgrade()
System.Object.Equals(System.Object)
System.Object.Equals(System.Object, System.Object)
System.Object.GetHashCode()
System.Object.GetType()
System.Object.MemberwiseClone()
System.Object.ReferenceEquals(System.Object, System.Object)
System.Object.ToString()

Namespace: SpaceVsInvaders.Model.Towers

Assembly: SpaceVsInvaders.dll

Syntax

```
public class SVsIGoldTower : SVsITower
```

Constructors

SVsIGoldTower()

SVsIGoldTower constructor.

Declaration

```
public SVsIGoldTower()
```

Methods

Gold()

Defines how much money it produces.

Declaration

```
public int Gold()
```

Returns

TYPE	DESCRIPTION
System.Int32	Its production.

Class SVsIHealTower

Derived class of tower that can heal other towers.

Inheritance

System.Object
SVsITower
SVsIHealTower

Inherited Members

SVsITower.Health
SVsITower.MaxHealth
SVsITower.Cost
SVsITower.Level
SVsITower.TickTime
SVsITower.CoolDown
SVsITower.Range
SVsITower.UpgradeCost
SVsITower.SellCost
SVsITower.Upgrade()
System.Object.Equals(System.Object)
System.Object.Equals(System.Object, System.Object)
System.Object.GetHashCode()
System.Object.GetType()
System.Object.MemberwiseClone()
System.Object.ReferenceEquals(System.Object, System.Object)
System.Object.ToString()

Namespace: SpaceVsInvaders.Model.Towers

Assembly: SpaceVsInvaders.dll

Syntax

```
public class SVsIHealTower : SVsITower
```

Constructors

SVsIHealTower()

SVsIHealTower constructor.

Declaration

```
public SVsIHealTower()
```

Methods

Heal()

Defines how much the other towers' healthpoints are increased by.

Declaration

```
public int Heal()
```

Returns

TYPE	DESCRIPTION
System.Int32	Its healing.

Class SVsITower

Base class of towers.

Inheritance

System.Object
SVsITower
SVsIDamageTower
SVsIGoldTower
SVsIHealTower

Inherited Members

System.Object.Equals(System.Object)
System.Object.Equals(System.Object, System.Object)
System.Object.GetHashCode()
System.Object.GetType()
System.Object.MemberwiseClone()
System.Object.ReferenceEquals(System.Object, System.Object)
System.Object.ToString()

Namespace: SpaceVsInvaders.Model.Towers

Assembly: SpaceVsInvaders.dll

Syntax

```
public abstract class SVsITower
```

Constructors

SVsITower()

SVsITower constructor - public.

Declaration

```
public SVsITower()
```

SVsITower(TowerConfig)

SVsITower constructor - protected.

Declaration

```
protected SVsITower(TowerConfig conf)
```

Parameters

TYPE	NAME	DESCRIPTION
TowerConfig	conf	

Properties

CoolDown

How many ticks are left until the next action.

Declaration

```
public int CoolDown { get; set; }
```

Property Value

TYPE	DESCRIPTION
System.Int32	CoolDown

Cost

The tower's cost.

Declaration

```
public int Cost { get; protected set; }
```

Property Value

TYPE	DESCRIPTION
System.Int32	Cost

Health

The tower's current health.

Declaration

```
public int Health { get; set; }
```

Property Value

TYPE	DESCRIPTION
System.Int32	Health

Level

The current level of the tower.

Declaration

```
public int Level { get; set; }
```

Property Value

TYPE	DESCRIPTION
System.Int32	Level

MaxHealth

The maximum value of the tower's health.

Declaration

```
public int MaxHealth { get; set; }
```

Property Value

TYPE	DESCRIPTION
System.Int32	MaxHealth

Range

The range within the specific action is made.

Declaration

```
public int Range { get; set; }
```

Property Value

TYPE	DESCRIPTION
System.Int32	Range

SellCost

The amount of money the player receives when the tower is sold.

Declaration

```
public int SellCost { get; }
```

Property Value

TYPE	DESCRIPTION
System.Int32	SellCost

TickTime

How often the tower's specific action repeats.

Declaration

```
public int TickTime { get; set; }
```

Property Value

TYPE	DESCRIPTION
System.Int32	TickTime

UpgradeCost

The cost of the tower's upgrade.

Declaration

```
public int UpgradeCost { get; }
```

Property Value

TYPE	DESCRIPTION
System.Int32	

Methods

Upgrade()

Upgrades the tower.

Declaration

```
public void Upgrade()
```

Namespace SpaceVsInvaders.View

Classes

ContentLoader

Singleton that can load textures, fonts, sounds or music

KeyboardHandler

Handles key presses and emits events accordingly

StateManager

Manages the game view's state

Class ContentLoader

Singleton that can load textures, fonts, sounds or music

Inheritance

System.Object
ContentLoader

Inherited Members

System.Object.Equals(System.Object)
System.Object.Equals(System.Object, System.Object)
System.Object.GetHashCode()
System.Object.GetType()
System.Object.MemberwiseClone()
System.Object.ReferenceEquals(System.Object, System.Object)
System.Object.ToString()

Namespace: SpaceVsInvaders.View

Assembly: SpaceVsInvaders.dll

Syntax

```
public static class ContentLoader
```

Methods

AttachGraphicsDevice(GraphicsDevice)

Attach the Graphics Device to the ContentLoader, to allow the creating of textures

Declaration

```
public static void AttachGraphicsDevice(GraphicsDevice gd)
```

Parameters

TYPE	NAME	DESCRIPTION
Microsoft.Xna.Framework.Graphics.GraphicsDevice	gd	Main Graphics Device

CreateSolidtexture(Color)

Creates a solid 1px by 1px texture

Declaration

```
public static Texture2D CreateSolidtexture(Color color)
```

Parameters

TYPE	NAME	DESCRIPTION
Microsoft.Xna.Framework.Color	color	Color of the texture

Returns

TYPE	DESCRIPTION
Microsoft.Xna.Framework.Graphics.Texture2D	The created texture

Examples

spriteBatch.Draw(ContentLoader.CreateSolidtexture(Color.Green), new Rectangle(0,0, Width, Height), Color.White * backOpacity);

GetFont(String)

Fetches the font of the given path

Declaration

```
public static SpriteFont GetFont(string path)
```

Parameters

TYPE	NAME	DESCRIPTION
System.String	path	The font's path

Returns

TYPE	DESCRIPTION
Microsoft.Xna.Framework.Graphics.SpriteFont	The requested font

Exceptions

TYPE	CONDITION
System.ArgumentException	Invalid path

GetSong(String)

Fetches the song of the given path

Declaration

```
public static Song GetSong(string path)
```

Parameters

TYPE	NAME	DESCRIPTION
System.String	path	The song's path

Returns

TYPE	DESCRIPTION

TYPE	DESCRIPTION
Microsoft.Xna.Framework.Media.Song	The requested song

Exceptions

TYPE	CONDITION
System.ArgumentException	Invalid path

GetSoundEffect(String)

Fetches the sound effect of the given path

Declaration

```
public static SoundEffect GetSoundEffect(string path)
```

Parameters

TYPE	NAME	DESCRIPTION
System.String	path	The sound effect's path

Returns

TYPE	DESCRIPTION
Microsoft.Xna.Framework.Audio.SoundEffect	The requested sound effect

Exceptions

TYPE	CONDITION
System.ArgumentException	Invalid path

GetTexture(EnemyType)

Fetches the enemy's texture

Declaration

```
public static Texture2D GetTexture(this EnemyType tile)
```

Parameters

TYPE	NAME	DESCRIPTION
EnemyType	tile	The enemy's type

Returns

TYPE	DESCRIPTION
Microsoft.Xna.Framework.Graphics.Texture2D	The enemy's texture

GetTexture(TowerType)

Fetches the tower's texture

Declaration

```
public static Texture2D GetTexture(this TowerType tile)
```

Parameters

TYPE	NAME	DESCRIPTION
TowerType	tile	The tower's type

Returns

TYPE	DESCRIPTION
Microsoft.Xna.Framework.Graphics.Texture2D	The tower's texture

GetTexture(String)

Fetches the texture of the given path

Declaration

```
public static Texture2D GetTexture(string path)
```

Parameters

TYPE	NAME	DESCRIPTION
System.String	path	The texture's path

Returns

TYPE	DESCRIPTION
Microsoft.Xna.Framework.Graphics.Texture2D	The requested texture

Exceptions

TYPE	CONDITION
System.ArgumentException	Invalid path

LoadContent(ContentManager)

Loads content into the object using the ContentManger

Declaration

```
public static void LoadContent(ContentManager content)
```

Parameters

TYPE	NAME	DESCRIPTION
Microsoft.Xna.Framework.Content.ContentManager	content	ContentManager to load the content from

Remarks

Should be used during the initializaton of the main game

Class KeyboardHandler

Handles key presses and emits events accordingly

Inheritance

System.Object
KeyboardHandler

Inherited Members

System.Object.Equals(System.Object)
System.Object.Equals(System.Object, System.Object)
System.Object.GetHashCode()
System.Object.GetType()
System.Object.MemberwiseClone()
System.Object.ReferenceEquals(System.Object, System.Object)
System.Object.ToString()

Namespace: SpaceVsInvaders.View

Assembly: SpaceVsInvaders.dll

Syntax

```
public class KeyboardHandler
```

Constructors

KeyboardHandler()

Constructor of `KeyboardHandler`

Declaration

```
public KeyboardHandler()
```

Methods

Update(GameTime)

Checks the key states and emits pressed events

Declaration

```
public void Update(GameTime gameTime)
```

Parameters

TYPE	NAME	DESCRIPTION
Microsoft.Xna.Framework.GameTime	gameTime	Game time

Events

KeyPressed

Event of a key pressed

Declaration

```
public event EventHandler<Keys> KeyPressed
```

Event Type

TYPE	DESCRIPTION
System.EventHandler<Microsoft.Xna.Framework.Input.Keys>	

Class StateManager

Manages the game view's state

Inheritance

System.Object
StateManager

Inherited Members

System.Object.Equals(System.Object)
System.Object.Equals(System.Object, System.Object)
System.Object.GetHashCode()
System.Object.GetType()
System.Object.MemberwiseClone()
System.Object.ReferenceEquals(System.Object, System.Object)
System.Object.ToString()

Namespace: SpaceVsInvaders.View

Assembly: SpaceVsInvaders.dll

Syntax

```
public class StateManager
```

Constructors

StateManager(SVsIModel, ErrorDisplay)

Constructor of `StateManager`

Declaration

```
public StateManager(SVsIModel model, ErrorDisplay errorDisplay)
```

Parameters

TYPE	NAME	DESCRIPTION
SVsIModel	model	Model to be used
ErrorDisplay	errorDisplay	Error display to be used

Properties

GameOver

Whether the game is over

Declaration

```
public bool GameOver { get; }
```

Property Value

TYPE	DESCRIPTION

TYPE	DESCRIPTION
System.Boolean	Whether the game is over

PlacingTower

Whether a tower is currently being placed

Declaration

```
public bool PlacingTower { get; }
```

Property Value

TYPE	DESCRIPTION
System.Boolean	Whether a tower is currently being placed

SelectedPos

Selected tower's position

Declaration

```
public (int, int) SelectedPos { get; }
```

Property Value

TYPE	DESCRIPTION
System.ValueTuple<System.Int32, System.Int32>	Selected tower's position

SelectedTower

Currently selected tower

Declaration

```
public SVsITower SelectedTower { get; }
```

Property Value

TYPE	DESCRIPTION
SVsITower	Currently selected tower

TowerPlacingType

Type of the tower currently being placed

Declaration

```
public TowerType TowerPlacingType { get; }
```

Property Value

TYPE	DESCRIPTION
TowerType	Type of the tower currently being placed

Victory

Whether the game resulted in victory

Declaration

```
public bool Victory { get; }
```

Property Value

TYPE	DESCRIPTION
System.Boolean	Whether the game resulted in victory

Methods

HandleCastleUpgradeClicked(Object, EventArgs)

Handles Castle upgrade button click event

Declaration

```
public void HandleCastleUpgradeClicked(object sender, EventArgs args)
```

Parameters

TYPE	NAME	DESCRIPTION
System.Object	sender	Sender object
System.EventArgs	args	(not used)

HandleEnterPressed(Object, EventArgs)

Handles Enter key press event, if currently placing tower it places it on the selected position

Declaration

```
public void HandleEnterPressed(object sender, EventArgs args)
```

Parameters

TYPE	NAME	DESCRIPTION
System.Object	sender	Sender object
System.EventArgs	args	(not used)

HandleEscapePressed(Object, EventArgs)

Handles Escape key press event, if PlacingTower then it cancles it, otherwise it opens the pause menu

Declaration

```
public void HandleEscapePressed(object sender, EventArgs args)
```

Parameters

TYPE	NAME	DESCRIPTION
System.Object	sender	Sender object
System.EventArgs	args	(not used)

HandleGameOver(Object, Boolean)

Handles game over event

Declaration

```
public void HandleGameOver(object sender, bool victory)
```

Parameters

TYPE	NAME	DESCRIPTION
System.Object	sender	Sender object
System.Boolean	victory	Whether the player won or not

HandleMoveKeysPressed(Object, Keys)

Handles move key press event, adjusts the current selected position accordingly

Declaration

```
public void HandleMoveKeysPressed(object sender, Keys key)
```

Parameters

TYPE	NAME	DESCRIPTION
System.Object	sender	Sender object
Microsoft.Xna.Framework.Input.Keys	key	Key pressed (must be one of the arrow keys)

HandleNewTowerType(TowerType)

Handles tower buy button click event

Declaration

```
public void HandleNewTowerType(TowerType type)
```

Parameters

TYPE	NAME	DESCRIPTION
TowerType	type	Tower type to be bought

HandleTileClicked(Object, (Int32, Int32))

Handles tile clicks, either places a tower or selects the tower

Declaration

```
public void HandleTileClicked(object sender, (int, int) pos)
```

Parameters

TYPE	NAME	DESCRIPTION
System.Object	sender	Sender object
System.ValueTuple<System.Int32, System.Int32>	pos	Tile position (row, col)

HandleTowerBuyClicked(Object, TowerType)

Handles tower buy button click event

Declaration

```
public void HandleTowerBuyClicked(object sender, TowerType towerType)
```

Parameters

TYPE	NAME	DESCRIPTION
System.Object	sender	Sender object
TowerType	towerType	Tower to be bought

HandleTowerSellClicked(Object, EventArgs)

Handles upgrade tower button click event

Declaration

```
public void HandleTowerSellClicked(object sender, EventArgs args)
```

Parameters

TYPE	NAME	DESCRIPTION

TYPE	NAME	DESCRIPTION
System.Object	sender	Sender object
System.EventArgs	args	(not used)

HandleTowerUpgradeClicked(Object, EventArgs)

Handles tower upgrade buttons click event

Declaration

```
public void HandleTowerUpgradeClicked(object sender, EventArgs args)
```

Parameters

TYPE	NAME	DESCRIPTION
System.Object	sender	Sender object
System.EventArgs	args	(not used)

Events

OpenPauseMenu

Pause menu should be opened event

Declaration

```
public event EventHandler OpenPauseMenu
```

Event Type

TYPE	DESCRIPTION
System.EventHandler	

Namespace SpaceVsInvaders.View.Boards

Classes

Board

Game board component

Catastrophe

Includes main informations about a Catastrophe

CatastropheAnimator

Puts the catastrophes on the board

EnemyTile

Manages one or more enemies on a given tile

Mothership

Animated mothership component

Shot

Data of a shot

ShotAnimator

Shot Animator component

Tile

Tile component used in the Game Board

TowerTile

Tower tile used in the Game Board

UnderCursorTower

Displays the current tower to be bought under the cursor

Class Board

Game board component

Inheritance

System.Object
Component
Board

Inherited Members

Component.height
Component.width
Component.position
Component.area
System.Object.Equals(System.Object)
System.Object.Equals(System.Object, System.Object)
System.Object.GetHashCode()
System.Object.GetType()
System.Object.MemberwiseClone()
System.Object.ReferenceEquals(System.Object, System.Object)
System.Object.ToString()

Namespace: SpaceVsInvaders.View.Boards

Assembly: SpaceVsInvaders.dll

Syntax

```
public class Board : Component
```

Constructors

Board(Vector2, Int32, Int32, SVsIModel, StateManager)

Constructor of `Board`

Declaration

```
public Board(Vector2 position, int height, int width, SVsIModel model, StateManager stateManager)
```

Parameters

TYPE	NAME	DESCRIPTION
Microsoft.Xna.Framework.Vector2	position	Position
System.Int32	height	Height of the board
System.Int32	width	Width of the board
SVsIModel	model	Model to get the data from
StateManager	stateManager	State manager to get the state from

Properties

CatastropheAnimator

CatastropheAnimator of the board

Declaration

```
public CatastropheAnimator CatastropheAnimator { get; set; }
```

Property Value

TYPE	DESCRIPTION
CatastropheAnimator	CatastropheAnimator of the board

ShotAnimator

Shot animator of the board

Declaration

```
public ShotAnimator ShotAnimator { get; set; }
```

Property Value

TYPE	DESCRIPTION
ShotAnimator	Shot animator of the board

Methods

Draw(SpriteBatch)

Draws the board to the spritebatch

Declaration

```
public override void Draw(SpriteBatch spriteBatch)
```

Parameters

TYPE	NAME	DESCRIPTION
Microsoft.Xna.Framework.Graphics.SpriteBatch	spriteBatch	Spritebatch to draw to

Overrides

Component.Draw(SpriteBatch)

Update(GameTime)

Updates the board

Declaration

```
public override void Update(GameTime gameTime)
```

Parameters

TYPE	NAME	DESCRIPTION
Microsoft.Xna.Framework.GameTime	gameTime	Gametime

Overrides

Component.Update(GameTime)

Events

TileClicked

Board tile has been clicked event, the tuple is the row and column of the tile

Declaration

```
public event EventHandler<(int, int)> TileClicked
```

Event Type

TYPE	DESCRIPTION
System.EventHandler<System.ValueTuple<System.Int32, System.Int32>>	

Class Catastrophe

Includes main informations about a Catastrophe

Inheritance

System.Object
Catastrophe

Inherited Members

System.Object.Equals(System.Object)
System.Object.Equals(System.Object, System.Object)
System.Object.GetHashCode()
System.Object.GetType()
System.Object.MemberwiseClone()
System.Object.ReferenceEquals(System.Object, System.Object)
System.Object.ToString()

Namespace: SpaceVsInvaders.View.Boards

Assembly: SpaceVsInvaders.dll

Syntax

```
public class Catastrophe
```

Constructors

Catastrophe(CatastropheType, Int32, Int32, Int32)

Constructor of a Catastrophe

Declaration

```
public Catastrophe(CatastropheType type, int x, int y, int sec)
```

Parameters

TYPE	NAME	DESCRIPTION
CatastropheType	type	Type of the catastrophe
System.Int32	x	X coordinate
System.Int32	y	y coordinate
System.Int32	sec	Seconds on the board

Properties

SecRemaining

How many seconds the catastrophe is on the board

Declaration

```
public int SecRemaining { get; set; }
```

Property Value

TYPE	DESCRIPTION
System.Int32	int

type

The type of the catastrophe

Declaration

```
public CatastropheType type { get; set; }
```

Property Value

TYPE	DESCRIPTION
CatastropheType	

X

the X coordinate of the catastrophe

Declaration

```
public int X { get; set; }
```

Property Value

TYPE	DESCRIPTION
System.Int32	int

Y

The Y coordinate of the catastrophe

Declaration

```
public double Y { get; set; }
```

Property Value

TYPE	DESCRIPTION
System.Double	int

Class CatastropheAnimator

Puts the catastrophes on the board

Inheritance

System.Object
Component
CatastropheAnimator

Inherited Members

Component.height
Component.width
Component.position
Component.area
System.Object.Equals(System.Object)
System.Object.Equals(System.Object, System.Object)
System.Object.GetHashCode()
System.Object.GetType()
System.Object.MemberwiseClone()
System.Object.ReferenceEquals(System.Object, System.Object)
System.Object.ToString()

Namespace: SpaceVsInvaders.View.Boards

Assembly: SpaceVsInvaders.dll

Syntax

```
public class CatastropheAnimator : Component
```

Constructors

CatastropheAnimator(Vector2, Int32, Int32, Int32, Int32)

Constructor of CatastropheAnimator

Declaration

```
public CatastropheAnimator(Vector2 position, int height, int width, int colWidth, int rowHeight)
```

Parameters

TYPE	NAME	DESCRIPTION
Microsoft.Xna.Framework.Vector2	position	where it is on the board
System.Int32	height	
System.Int32	width	
System.Int32	colWidth	
System.Int32	rowHeight	

Methods

Draw(SpriteBatch)

Draws the Catastrophe on the screen

Declaration

```
public override void Draw(SpriteBatch spriteBatch)
```

Parameters

TYPE	NAME	DESCRIPTION
Microsoft.Xna.Framework.Graphics.SpriteBatch	spriteBatch	Includes texture of the Catastrophe

Overrides

Component.Draw(SpriteBatch)

HandleAsteroids(Object, SVsIEventArgs)

Eventhandler, that handles Damage Catastrophes

Declaration

```
public void HandleAsteroids(object sender, SVsIEventArgs args)
```

Parameters

TYPE	NAME	DESCRIPTION
System.Object	sender	Sender object
SVsIEventArgs	args	The information that is sent

HandleHealing(Object, SVsIEventArgs)

Eventhandler, that handles Healing Catastrophes

Declaration

```
public void HandleHealing(object sender, SVsIEventArgs args)
```

Parameters

TYPE	NAME	DESCRIPTION
System.Object	sender	Sender object
SVsIEventArgs	args	The information that is sent

Update(GameTime)

This handles the function of the class, is called in every tick

Declaration

```
public override void Update(GameTime gameTime)
```

Parameters

TYPE	NAME	DESCRIPTION
Microsoft.Xna.Framework.GameTime	gameTime	The time elapsed ingame

Overrides

Component.Update(GameTime)

Class EnemyTile

Manages one or more enemies on a given tile

Inheritance

System.Object

Component

Clickable

Tile

EnemyTile

Inherited Members

Tile.Row

Tile.Col

Tile.DrawOutlinedString(SpriteBatch, SpriteFont, String, Vector2, Color)

Clickable.LeftClicked

Clickable.RightClicked

Clickable.MouseEnter

Clickable.CurrentlyClicked

Clickable.isMouseOver()

Clickable.Update(GameTime)

Component.height

Component.width

Component.position

Component.area

System.Object.Equals(System.Object)

System.Object.Equals(System.Object, System.Object)

System.Object.GetHashCode()

System.Object.GetType()

System.Object.MemberwiseClone()

System.Object.ReferenceEquals(System.Object, System.Object)

System.Object.ToString()

Namespace: SpaceVsInvaders.View.Boards

Assembly: SpaceVsInvaders.dll

Syntax

```
public class EnemyTile : Tile
```

Constructors

EnemyTile(Vector2, Int32, Int32, Int32, Int32, StateManager, List<(EnemyType, Int32)>, Int32, Int32)

Constructor of `EnemyTile`

Declaration

```
public EnemyTile(Vector2 position, int height, int width, int row, int col, StateManager stateManager,
List<(EnemyType, int)> enemies, int currHealth, int maxHealth)
```

Parameters

TYPE	NAME	DESCRIPTION
Microsoft.Xna.Framework.Vector2	position	Position

TYPE	NAME	DESCRIPTION
System.Int32	height	Height
System.Int32	width	Width
System.Int32	row	Row of the tile
System.Int32	col	Column of the tile
StateManager	stateManager	State manager to get the state from
System.Collections.Generic.List<System.ValueTuple<EnemyType, System.Int32>>	enemies	Enemies currently on the tile, the tuple is how much enemies are there of a given type
System.Int32	currHealth	Current health of the first enemy
System.Int32	maxHealth	Max health of the first enemy

Methods

Draw(SpriteBatch)

Draws the tile to the spritebatch

Declaration

```
public override void Draw(SpriteBatch spriteBatch)
```

Parameters

TYPE	NAME	DESCRIPTION
Microsoft.Xna.Framework.Graphics.SpriteBatch	spriteBatch	Spritebatch

Overrides

Tile.Draw(SpriteBatch)

Class Mothership

Animated mothership component

Inheritance

System.Object
Component
Mothership

Inherited Members

Component.height
Component.width
Component.position
Component.area
System.Object.Equals(System.Object)
System.Object.Equals(System.Object, System.Object)
System.Object.GetHashCode()
System.Object.GetType()
System.Object.MemberwiseClone()
System.Object.ReferenceEquals(System.Object, System.Object)
System.Object.ToString()

Namespace: SpaceVsInvaders.View.Boards

Assembly: SpaceVsInvaders.dll

Syntax

```
public class Mothership : Component
```

Constructors

Mothership(Vector2, Int32, Int32, StateManager)

Constructor of `Mothership`

Declaration

```
public Mothership(Vector2 position, int height, int width, StateManager stateManager)
```

Parameters

TYPE	NAME	DESCRIPTION
Microsoft.Xna.Framework.Vector2	position	Position
System.Int32	height	Height of the area to patrol
System.Int32	width	
StateManager	stateManager	

Methods

Draw(SpriteBatch)

Draws the mothership to the spritebatch

Declaration

```
public override void Draw(SpriteBatch spriteBatch)
```

Parameters

TYPE	NAME	DESCRIPTION
Microsoft.Xna.Framework.Graphics.SpriteBatch	spriteBatch	Spritebatch

Overrides

Component.Draw(SpriteBatch)

Update(GameTime)

Updates the position of the mothership

Declaration

```
public override void Update(GameTime gameTime)
```

Parameters

TYPE	NAME	DESCRIPTION
Microsoft.Xna.Framework.GameTime	gameTime	Gametime

Overrides

Component.Update(GameTime)

Class Shot

Data of a shot

Inheritance

System.Object
Shot

Inherited Members

System.Object.Equals(System.Object)
System.Object.Equals(System.Object, System.Object)
System.Object.GetHashCode()
System.Object.GetType()
System.Object.MemberwiseClone()
System.Object.ReferenceEquals(System.Object, System.Object)
System.Object.ToString()

Namespace: SpaceVsInvaders.View.Boards

Assembly: SpaceVsInvaders.dll

Syntax

```
public class Shot
```

Properties

FromY

Starting Y coordinate of the shot

Declaration

```
public int FromY { get; set; }
```

Property Value

TYPE	DESCRIPTION
System.Int32	Starting Y coordinate of the shot

SecRemaining

Seconds remaining of the shot's lifespan

Declaration

```
public double SecRemaining { get; set; }
```

Property Value

TYPE	DESCRIPTION
System.Double	Seconds remaining of the shot's lifespan

ToY

Destination Y coordinate of the shot

Declaration

```
public int ToY { get; set; }
```

Property Value

TYPE	DESCRIPTION
System.Int32	Destination Y coordinate of the shot

X

Current X coordinate of the shot

Declaration

```
public int X { get; set; }
```

Property Value

TYPE	DESCRIPTION
System.Int32	Current X coordinate of the shot

Y

Current Y coordinate of the shot

Declaration

```
public double Y { get; set; }
```

Property Value

TYPE	DESCRIPTION
System.Double	Current Y coordinate of the shot

Class ShotAnimator

Shot Animator component

Inheritance

System.Object
Component
ShotAnimator

Inherited Members

Component.height
Component.width
Component.position
Component.area
System.Object.Equals(System.Object)
System.Object.Equals(System.Object, System.Object)
System.Object.GetHashCode()
System.Object.GetType()
System.Object.MemberwiseClone()
System.Object.ReferenceEquals(System.Object, System.Object)
System.Object.ToString()

Namespace: SpaceVsInvaders.View.Boards

Assembly: SpaceVsInvaders.dll

Syntax

```
public class ShotAnimator : Component
```

Constructors

ShotAnimator(Vector2, Int32, Int32, Int32, Int32)

Constructor of `ShotAnimator`

Declaration

```
public ShotAnimator(Vector2 position, int height, int width, int colWidth, int rowHeight)
```

Parameters

TYPE	NAME	DESCRIPTION
Microsoft.Xna.Framework.Vector2	position	Position of the board to overlay on
System.Int32	height	Height of the board to overlay on
System.Int32	width	Width of the board to overlay on
System.Int32	colWidth	Column width

TYPE	NAME	DESCRIPTION
System.Int32	rowHeight	Row height

Methods

Draw(SpriteBatch)

Draw each shot to the spritebatch

Declaration

```
public override void Draw(SpriteBatch spriteBatch)
```

Parameters

TYPE	NAME	DESCRIPTION
Microsoft.Xna.Framework.Graphics.SpriteBatch	spriteBatch	Spritebatch

Overrides

Component.Draw(SpriteBatch)

HandleNewShot(Object, SVsIEventArgs)

Handles the addition of a enw shot

Declaration

```
public void HandleNewShot(object sender, SVsIEventArgs args)
```

Parameters

TYPE	NAME	DESCRIPTION
System.Object	sender	Sender object
SVsIEventArgs	args	Event args

Update(GameTime)

Update each shot's location

Declaration

```
public override void Update(GameTime gameTime)
```

Parameters

TYPE	NAME	DESCRIPTION
Microsoft.Xna.Framework.GameTime	gameTime	Gametime

Overrides

Component.Update(GameTime)

Class Tile

Tile component used in the Game Board

Inheritance

System.Object
Component
Clickable
Tile
EnemyTile
TowerTile

Inherited Members

Clickable.LeftClicked
Clickable.RightClicked
Clickable.MouseEnter
Clickable.CurrentlyClicked
Clickable.isMouseOver()
Clickable.Update(GameTime)
Component.height
Component.width
Component.position
Component.area
System.Object.Equals(System.Object)
System.Object.Equals(System.Object, System.Object)
System.Object.GetHashCode()
System.Object.GetType()
System.Object.MemberwiseClone()
System.Object.ReferenceEquals(System.Object, System.Object)
System.Object.ToString()

Namespace: SpaceVsInvaders.View.Boards

Assembly: SpaceVsInvaders.dll

Syntax

```
public class Tile : Clickable
```

Constructors

Tile(Vector2, Int32, Int32, Int32, Int32, StateManager)

Constructor of `Tile`

Declaration

```
public Tile(Vector2 position, int height, int width, int row, int col, StateManager stateManager)
```

Parameters

TYPE	NAME	DESCRIPTION
Microsoft.Xna.Framework.Vector2	position	Position
System.Int32	height	Height

TYPE	NAME	DESCRIPTION
System.Int32	width	Width
System.Int32	row	Row of the tile
System.Int32	col	Column of the tile
StateManager	stateManager	State manager to get the state from

Properties

Col

Column of the tile

Declaration

```
public int Col { get; set; }
```

Property Value

TYPE	DESCRIPTION
System.Int32	Column of the tile

Row

Row of the tile

Declaration

```
public int Row { get; set; }
```

Property Value

TYPE	DESCRIPTION
System.Int32	Row of the tile

Methods

Draw(SpriteBatch)

Draws the tile to the spritebatch

Declaration

```
public override void Draw(SpriteBatch spriteBatch)
```

Parameters

TYPE	NAME	DESCRIPTION
Microsoft.Xna.Framework.Graphics.SpriteBatch	spriteBatch	Spritebatch

Overrides

Component.Draw(SpriteBatch)

DrawOutlinedString(SpriteBatch, SpriteFont, String, Vector2, Color)

Draws an outlined string to the spritebatch

Declaration

```
protected void DrawOutlinedString(SpriteBatch spriteBatch, SpriteFont font, string text, Vector2 pos, Color
color)
```

Parameters

TYPE	NAME	DESCRIPTION
Microsoft.Xna.Framework.Graphics.SpriteBatch	spriteBatch	Spritebatch
Microsoft.Xna.Framework.Graphics.SpriteFont	font	Font
System.String	text	Text
Microsoft.Xna.Framework.Vector2	pos	Position
Microsoft.Xna.Framework.Color	color	Color

Class TowerTile

Tower tile used in the Game Board

Inheritance

System.Object
Component
Clickable
Tile
TowerTile

Inherited Members

Tile.Row
Tile.Col
Tile.DrawOutlinedString(SpriteBatch, SpriteFont, String, Vector2, Color)
Clickable.LeftClicked
Clickable.RightClicked
Clickable.MouseEnter
Clickable.CurrentlyClicked
Clickable.isMouseOver()
Clickable.Update(GameTime)
Component.height
Component.width
Component.position
Component.area
System.Object.Equals(System.Object)
System.Object.Equals(System.Object, System.Object)
System.Object.GetHashCode()
System.Object.GetType()
System.Object.MemberwiseClone()
System.Object.ReferenceEquals(System.Object, System.Object)
System.Object.ToString()

Namespace: SpaceVsInvaders.View.Boards

Assembly: SpaceVsInvaders.dll

Syntax

```
public class TowerTile : Tile
```

Constructors

TowerTile(Vector2, Int32, Int32, Int32, Int32, StateManager, TowerType, Int32, Int32)

Constructor of `TowerTile`

Declaration

```
public TowerTile(Vector2 position, int height, int width, int row, int col, StateManager stateManager,
TowerType tower, int currHealth, int maxHealth)
```

Parameters

TYPE	NAME	DESCRIPTION
Microsoft.Xna.Framework.Vector2	position	Position

TYPE	NAME	DESCRIPTION
System.Int32	height	Height
System.Int32	width	Width
System.Int32	row	Row of the tile
System.Int32	col	Column of the tile
StateManager	stateManager	State manager to get the state from
TowerType	tower	Tower type of the tile
System.Int32	currHealth	Current health
System.Int32	maxHealth	Max health

Methods

Draw(SpriteBatch)

Draw the tower tile to the spritebatch

Declaration

```
public override void Draw(SpriteBatch spriteBatch)
```

Parameters

TYPE	NAME	DESCRIPTION
Microsoft.Xna.Framework.Graphics.SpriteBatch	spriteBatch	Spritebatch

Overrides

Tile.Draw(SpriteBatch)

Class UnderCursorTower

Displays the current tower to be bought under the cursor

Inheritance

System.Object

Component

UnderCursorTower

Inherited Members

Component.height

Component.width

Component.position

Component.area

System.Object.Equals(System.Object)

System.Object.Equals(System.Object, System.Object)

System.Object.GetHashCode()

System.Object.GetType()

System.Object.MemberwiseClone()

System.Object.ReferenceEquals(System.Object, System.Object)

System.Object.ToString()

Namespace: SpaceVsInvaders.View.Boards

Assembly: SpaceVsInvaders.dll

Syntax

```
public class UnderCursorTower : Component
```

Constructors

UnderCursorTower(Vector2, Int32, Int32, StateManager)

Constructor of `UnderCursorTower`

Declaration

```
public UnderCursorTower(Vector2 position, int height, int width, StateManager stateManager)
```

Parameters

TYPE	NAME	DESCRIPTION
Microsoft.Xna.Framework.Vector2	position	Position (not used)
System.Int32	height	Height of the component
System.Int32	width	Width of the component
StateManager	stateManager	State manager to get the curent tower to be bought

Methods

Draw(SpriteBatch)

Draws the tower to the mouse position to the spritebatch

Declaration

```
public override void Draw(SpriteBatch spriteBatch)
```

Parameters

TYPE	NAME	DESCRIPTION
Microsoft.Xna.Framework.Graphics.SpriteBatch	spriteBatch	Spritebatch to draw to

Overrides

Component.Draw(SpriteBatch)

Update(GameTime)

Updates the UnderCursorTower, currently does nothing as there is nothing to update

Declaration

```
public override void Update(GameTime gameTime)
```

Parameters

TYPE	NAME	DESCRIPTION
Microsoft.Xna.Framework.GameTime	gameTime	Gametime

Overrides

Component.Update(GameTime)

Namespace SpaceVsInvaders.View.Components

Classes

BasePanel

Base panel component

Button

Button component

BuyPanel

Buy panel with buttons to buy towers

Clickable

Component that emits click events and handles mouse hovers

Component

Component base class, that can be rendered to the sceen

Cursor

Animated cursor component rendered at the current mouse position

ErrorDisplay

Displays errors ingame

GameOverPanel

Game over panel component

InfoPanel

Displays game information and castle upgrade button

TowerInfo

Currently selected Tower information display panel

Class BasePanel

Base panel component

Inheritance

System.Object

Component

BasePanel

BuyPanel

InfoPanel

TowerInfo

Inherited Members

Component.height

Component.width

Component.position

Component.area

Component.Update(GameTime)

System.Object.Equals(System.Object)

System.Object.Equals(System.Object, System.Object)

System.Object.GetHashCode()

System.Object.GetType()

System.Object.MemberwiseClone()

System.Object.ReferenceEquals(System.Object, System.Object)

System.Object.ToString()

Namespace: SpaceVsInvaders.View.Components

Assembly: SpaceVsInvaders.dll

Syntax

```
public abstract class BasePanel : Component
```

Remarks

PanelX,Y,Width,Height are the usable part of the panel that can be drawn to. If you use the base values you draw on the borders of the panel texture.

Constructors

BasePanel(Vector2, Int32, Int32)

Constructor of `BasePanel`

Declaration

```
public BasePanel(Vector2 position, int height, int width)
```

Parameters

TYPE	NAME	DESCRIPTION
Microsoft.Xna.Framework.Vector2	position	Position
System.Int32	height	Height of the panel

TYPE	NAME	DESCRIPTION
System.Int32	width	Width of the panel

Properties

PanelHeight

Usable height of the panel

Declaration

```
protected int PanelHeight { get; }
```

Property Value

TYPE	DESCRIPTION
System.Int32	Height

PanelWidth

Usable width of the panel

Declaration

```
protected int PanelWidth { get; }
```

Property Value

TYPE	DESCRIPTION
System.Int32	Width

PanelX

Usable base X position of the panel

Declaration

```
protected int PanelX { get; }
```

Property Value

TYPE	DESCRIPTION
System.Int32	X position

PanelY

Usable base Y position of the panel

Declaration

```
protected int PanelY { get; }
```

Property Value

TYPE	DESCRIPTION
System.Int32	Y position

Methods

Draw(SpriteBatch)

Draws the panel background to the spritebatch

Declaration

```
public override void Draw(SpriteBatch spriteBatch)
```

Parameters

TYPE	NAME	DESCRIPTION
Microsoft.Xna.Framework.Graphics.SpriteBatch	spriteBatch	Spritebatch to draw to

Overrides

Component.Draw(SpriteBatch)

Class Button

Button component

Inheritance

System.Object
Component
Clickable
Button

Inherited Members

Clickable.LeftClicked
Clickable.RightClicked
Clickable.MouseEnter
Clickable.CurrentlyClicked
Clickable.isMouseOver()
Clickable.Update(GameTime)
Component.height
Component.width
Component.position
Component.area
System.Object.Equals(System.Object)
System.Object.Equals(System.Object, System.Object)
System.Object.GetHashCode()
System.Object.GetType()
System.Object.MemberwiseClone()
System.Object.ReferenceEquals(System.Object, System.Object)
System.Object.ToString()

Namespace: SpaceVsInvaders.View.Components

Assembly: SpaceVsInvaders.dll

Syntax

```
public class Button : Clickable
```

Constructors

Button(Vector2, Int32, Int32, String)

Constructor of `Button`

Declaration

```
public Button(Vector2 position, int height, int width, string text = "NO TEXT")
```

Parameters

TYPE	NAME	DESCRIPTION
Microsoft.Xna.Framework.Vector2	position	Position
System.Int32	height	Height of the button

TYPE	NAME	DESCRIPTION
System.Int32	width	Width of the button
System.String	text	Text of the button

Methods

Draw(SpriteBatch)

Draws the button to the spritebatch

Declaration

```
public override void Draw(SpriteBatch spriteBatch)
```

Parameters

TYPE	NAME	DESCRIPTION
Microsoft.Xna.Framework.Graphics.SpriteBatch	spriteBatch	Spritebatch to draw to

Overrides

Component.Draw(SpriteBatch)

UpdateText(String)

Replaces the text of the button with a new text

Declaration

```
public void UpdateText(string newText)
```

Parameters

TYPE	NAME	DESCRIPTION
System.String	newText	New text

Class BuyPanel

Buy panel with buttons to buy towers

Inheritance

System.Object

Component

BasePanel

BuyPanel

Inherited Members

BasePanel.PanelX

BasePanel.PanelY

BasePanel.PanelWidth

BasePanel.PanelHeight

Component.height

Component.width

Component.position

Component.area

System.Object.Equals(System.Object)

System.Object.Equals(System.Object, System.Object)

System.Object.GetHashCode()

System.Object.GetType()

System.Object.MemberwiseClone()

System.Object.ReferenceEquals(System.Object, System.Object)

System.Object.ToString()

Namespace: SpaceVsInvaders.View.Components

Assembly: SpaceVsInvaders.dll

Syntax

```
public class BuyPanel : BasePanel
```

Constructors

BuyPanel(Vector2, Int32, Int32, StateManager)

Constructor of `BuyPanel`

Declaration

```
public BuyPanel(Vector2 position, int height, int width, StateManager stateManager)
```

Parameters

TYPE	NAME	DESCRIPTION
Microsoft.Xna.Framework.Vector2	position	Position of the panel
System.Int32	height	Height of the panel
System.Int32	width	Width of the panel

TYPE	NAME	DESCRIPTION
StateManager	stateManager	State manager to get the state from

Properties

DamageTowerButton

Damage tower buy button

Declaration

```
public Button DamageTowerButton { get; }
```

Property Value

TYPE	DESCRIPTION
Button	Damage tower buy button

GoldTowerButton

Gold tower buy button

Declaration

```
public Button GoldTowerButton { get; }
```

Property Value

TYPE	DESCRIPTION
Button	Gold tower buy button

HealTowerButton

Heal tower buy button

Declaration

```
public Button HealTowerButton { get; }
```

Property Value

TYPE	DESCRIPTION
Button	Heal tower buy button

Methods

Draw(SpriteBatch)

Draw the buy panel

Declaration

```
public override void Draw(SpriteBatch spriteBatch)
```

Parameters

TYPE	NAME	DESCRIPTION
Microsoft.Xna.Framework.Graphics.SpriteBatch	spriteBatch	Sprite batch to draw to

Overrides

BasePanel.Draw(SpriteBatch)

Update(GameTime)

Update the buttons of the buy panel

Declaration

```
public override void Update(GameTime gameTime)
```

Parameters

TYPE	NAME	DESCRIPTION
Microsoft.Xna.Framework.GameTime	gameTime	Game time

Overrides

Component.Update(GameTime)

Class Clickable

Component that emits click events and handles mouse hovers

Inheritance

System.Object
Component
Clickable
Tile
Button

Inherited Members

Component.height
Component.width
Component.position
Component.area
Component.Draw(SpriteBatch)
System.Object.Equals(System.Object)
System.Object.Equals(System.Object, System.Object)
System.Object.GetHashCode()
System.Object.GetType()
System.Object.MemberwiseClone()
System.Object.ReferenceEquals(System.Object, System.Object)
System.Object.ToString()

Namespace: SpaceVsInvaders.View.Components

Assembly: SpaceVsInvaders.dll

Syntax

```
public abstract class Clickable : Component
```

Constructors

Clickable(Vector2, Int32, Int32)

Constructor of `Clickable`

Declaration

```
public Clickable(Vector2 position, int height, int width)
```

Parameters

TYPE	NAME	DESCRIPTION
Microsoft.Xna.Framework.Vector2	position	Position
System.Int32	height	Height of the component
System.Int32	width	Width of the component

Properties

CurrentlyClicked

Whether the component is currently clicked

Declaration

```
protected bool CurrentlyClicked { get; }
```

Property Value

TYPE	DESCRIPTION
System.Boolean	Whether the component is currently clicked

Methods

isMouseOver()

Returns whether the mouse is over this component

Declaration

```
protected bool isMouseOver()
```

Returns

TYPE	DESCRIPTION
System.Boolean	Whether the mouse is over this component

Update(GameTime)

Handles mouse clicks and hovers based on the current mouse position

Declaration

```
public override void Update(GameTime gameTime)
```

Parameters

TYPE	NAME	DESCRIPTION
Microsoft.Xna.Framework.GameTime	gameTime	Game time

Overrides

Component.Update(GameTime)

Events

LeftClicked

The component has been left clicked event

Declaration

```
public event EventHandler LeftClicked
```

Event Type

TYPE	DESCRIPTION
System.EventHandler	

MouseEnter

The mouse entered the area of the component

Declaration

```
public event EventHandler MouseEnter
```

Event Type

TYPE	DESCRIPTION
System.EventHandler	

RightClicked

The component has been right clicked event

Declaration

```
public event EventHandler RightClicked
```

Event Type

TYPE	DESCRIPTION
System.EventHandler	

Class Component

Component base class, that can be rendered to the sceen

Inheritance

System.Object
Component
Board
CatastropheAnimator
Mothership
ShotAnimator
UnderCursorTower
BasePanel
Clickable
Cursor
ErrorDisplay
GameOverPanel

Inherited Members

System.Object.Equals(System.Object)
System.Object.Equals(System.Object, System.Object)
System.Object.GetHashCode()
System.Object.GetType()
System.Object.MemberwiseClone()
System.Object.ReferenceEquals(System.Object, System.Object)
System.Object.ToString()

Namespace: SpaceVsInvaders.View.Components

Assembly: SpaceVsInvaders.dll

Syntax

```
public abstract class Component
```

Constructors

Component(Vector2, Int32, Int32)

Constructor of `Component`

Declaration

```
public Component(Vector2 position, int height, int width)
```

Parameters

TYPE	NAME	DESCRIPTION
Microsoft.Xna.Framework.Vector2	position	Position
System.Int32	height	Height of the component
System.Int32	width	Width of the component

Fields

height

Height of the component

Declaration

```
protected int height
```

Field Value

TYPE	DESCRIPTION
System.Int32	

position

Position of the component

Declaration

```
protected Vector2 position
```

Field Value

TYPE	DESCRIPTION
Microsoft.Xna.Framework.Vector2	

width

Width of the component

Declaration

```
protected int width
```

Field Value

TYPE	DESCRIPTION
System.Int32	

Properties

area

Area of the component

Declaration

```
protected Rectangle area { get; }
```

Property Value

TYPE	DESCRIPTION
Microsoft.Xna.Framework.Rectangle	Rectangle area of the component

Methods

Draw(SpriteBatch)

Defines how to component should be drawn on the screen

Declaration

```
public abstract void Draw(SpriteBatch spriteBatch)
```

Parameters

TYPE	NAME	DESCRIPTION
Microsoft.Xna.Framework.Graphics.SpriteBatch	spriteBatch	

Update(GameTime)

Defines how the component should be updated each frame

Declaration

```
public abstract void Update(GameTime gameTime)
```

Parameters

TYPE	NAME	DESCRIPTION
Microsoft.Xna.Framework.GameTime	gameTime	

Class Cursor

Animated cursor component rendered at the current mouse position

Inheritance

System.Object
Component
Cursor

Inherited Members

Component.height
Component.width
Component.position
Component.area
System.Object.Equals(System.Object)
System.Object.Equals(System.Object, System.Object)
System.Object.GetHashCode()
System.Object.GetType()
System.Object.MemberwiseClone()
System.Object.ReferenceEquals(System.Object, System.Object)
System.Object.ToString()

Namespace: SpaceVsInvaders.View.Components

Assembly: SpaceVsInvaders.dll

Syntax

```
public class Cursor : Component
```

Constructors

Cursor(Vector2, Int32, Int32)

Constructor of `Cursor`

Declaration

```
public Cursor(Vector2 position, int height, int width)
```

Parameters

TYPE	NAME	DESCRIPTION
Microsoft.Xna.Framework.Vector2	position	Position (not used)
System.Int32	height	Height of the cursor
System.Int32	width	Width of the cursor

Methods

Draw(SpriteBatch)

Draws the cursor to the spritebatch

Declaration

```
public override void Draw(SpriteBatch spriteBatch)
```

Parameters

TYPE	NAME	DESCRIPTION
Microsoft.Xna.Framework.Graphics.SpriteBatch	spriteBatch	Spritebatch to draw to

Overrides

Component.Draw(SpriteBatch)

Update(GameTime)

Updates the currently displaying frame of the animation

Declaration

```
public override void Update(GameTime gameTime)
```

Parameters

TYPE	NAME	DESCRIPTION
Microsoft.Xna.Framework.GameTime	gameTime	Game time

Overrides

Component.Update(GameTime)

Class ErrorDisplay

Displays errors ingame

Inheritance

System.Object

Component

ErrorDisplay

Inherited Members

Component.height

Component.width

Component.position

Component.area

System.Object.Equals(System.Object)

System.Object.Equals(System.Object, System.Object)

System.Object.GetHashCode()

System.Object.GetType()

System.Object.MemberwiseClone()

System.Object.ReferenceEquals(System.Object, System.Object)

System.Object.ToString()

Namespace: SpaceVsInvaders.View.Components

Assembly: SpaceVsInvaders.dll

Syntax

```
public class ErrorDisplay : Component
```

Constructors

ErrorDisplay(Vector2, Int32, Int32)

constructor of ErroDisplay

Declaration

```
public ErrorDisplay(Vector2 position, int height, int width)
```

Parameters

TYPE	NAME	DESCRIPTION
Microsoft.Xna.Framework.Vector2	position	
System.Int32	height	
System.Int32	width	

Fields

Errors

List of Errors

Declaration

```
public List<(string, int)> Errors
```

Field Value

TYPE	DESCRIPTION
System.Collections.Generic.List<System.ValueTuple<System.String, System.Int32>>	

Methods

AddError(String)

Adds an error to the Error List

Declaration

```
public void AddError(string error)
```

Parameters

TYPE	NAME	DESCRIPTION
System.String	error	This is the message of the error

Draw(SpriteBatch)

Draws the error on the screen

Declaration

```
public override void Draw(SpriteBatch spriteBatch)
```

Parameters

TYPE	NAME	DESCRIPTION
Microsoft.Xna.Framework.Graphics.SpriteBatch	spriteBatch	The texture of the Error

Overrides

Component.Draw(SpriteBatch)

Update(GameTime)

This handles the function of the class, is called in every tick

Declaration

```
public override void Update(GameTime gameTime)
```

Parameters

TYPE	NAME	DESCRIPTION
Microsoft.Xna.Framework.GameTime	gameTime	The time elapsed ingame

Overrides

Component.Update(GameTime)

Class GameOverPanel

Game over panel component

Inheritance

System.Object

Component

GameOverPanel

Inherited Members

Component.height

Component.width

Component.position

Component.area

System.Object.Equals(System.Object)

System.Object.Equals(System.Object, System.Object)

System.Object.GetHashCode()

System.Object.GetType()

System.Object.MemberwiseClone()

System.Object.ReferenceEquals(System.Object, System.Object)

System.Object.ToString()

Namespace: SpaceVsInvaders.View.Components

Assembly: SpaceVsInvaders.dll

Syntax

```
public class GameOverPanel : Component
```

Constructors

GameOverPanel(Vector2, Int32, Int32, StateManager)

Constructor of `GameOverPanel`

Declaration

```
public GameOverPanel(Vector2 position, int height, int width, StateManager stateManager)
```

Parameters

TYPE	NAME	DESCRIPTION
Microsoft.Xna.Framework.Vector2	position	Position
System.Int32	height	Height
System.Int32	width	Width
StateManager	stateManager	StateManager to get the state from

Properties

MainMenuButton

Main Menu button

Declaration

```
public Button MainMenuButton { get; }
```

Property Value

TYPE	DESCRIPTION
Button	Main Menu button

Methods

Draw(SpriteBatch)

Draws the game over panel to the spritebatch

Declaration

```
public override void Draw(SpriteBatch spriteBatch)
```

Parameters

TYPE	NAME	DESCRIPTION
Microsoft.Xna.Framework.Graphics.SpriteBatch	spriteBatch	Spritebatch to draw to

Overrides

Component.Draw(SpriteBatch)

Update(GameTime)

Updates whether the game is over and the panel should be drawn

Declaration

```
public override void Update(GameTime gameTime)
```

Parameters

TYPE	NAME	DESCRIPTION
Microsoft.Xna.Framework.GameTime	gameTime	Game time

Overrides

Component.Update(GameTime)

Class InfoPanel

Displays game information and castle upgrade button

Inheritance

System.Object

Component

BasePanel

InfoPanel

Inherited Members

BasePanel.PanelX

BasePanel.PanelY

BasePanel.PanelWidth

BasePanel.PanelHeight

Component.height

Component.width

Component.position

Component.area

System.Object.Equals(System.Object)

System.Object.Equals(System.Object, System.Object)

System.Object.GetHashCode()

System.Object.GetType()

System.Object.MemberwiseClone()

System.Object.ReferenceEquals(System.Object, System.Object)

System.Object.ToString()

Namespace: SpaceVsInvaders.View.Components

Assembly: SpaceVsInvaders.dll

Syntax

```
public class InfoPanel : BasePanel
```

Constructors

InfoPanel(Vector2, Int32, Int32, SVsIModel, StateManager)

Constructor of `InfoPanel`

Declaration

```
public InfoPanel(Vector2 position, int height, int width, SVsIModel model, StateManager stateManager)
```

Parameters

TYPE	NAME	DESCRIPTION
Microsoft.Xna.Framework.Vector2	position	Position of the panel
System.Int32	height	Height of the panel
System.Int32	width	Width of the panel

TYPE	NAME	DESCRIPTION
SVsIModel	model	Model to get the data from
StateManager	stateManager	State manager to get the state from

Properties

UpgradeCastleButton

Upgrade castle button

Declaration

```
public Button UpgradeCastleButton { get; }
```

Property Value

TYPE	DESCRIPTION
Button	Upgrade castle button

Methods

Draw(SpriteBatch)

Draw the info panel

Declaration

```
public override void Draw(SpriteBatch spriteBatch)
```

Parameters

TYPE	NAME	DESCRIPTION
Microsoft.Xna.Framework.Graphics.SpriteBatch	spriteBatch	Spritebatch to draw to

Overrides

BasePanel.Draw(SpriteBatch)

Update(GameTime)

Update the info panel

Declaration

```
public override void Update(GameTime gameTime)
```

Parameters

TYPE	NAME	DESCRIPTION
Microsoft.Xna.Framework.GameTime	gameTime	Gametime

Overrides

Component.Update(GameTime)

Class TowerInfo

Currently selected Tower information display panel

Inheritance

System.Object

Component

BasePanel

TowerInfo

Inherited Members

BasePanel.PanelX

BasePanel.PanelY

BasePanel.PanelWidth

BasePanel.PanelHeight

Component.height

Component.width

Component.position

Component.area

System.Object.Equals(System.Object)

System.Object.Equals(System.Object, System.Object)

System.Object.GetHashCode()

System.Object.GetType()

System.Object.MemberwiseClone()

System.Object.ReferenceEquals(System.Object, System.Object)

System.Object.ToString()

Namespace: SpaceVsInvaders.View.Components

Assembly: SpaceVsInvaders.dll

Syntax

```
public class TowerInfo : BasePanel
```

Constructors

TowerInfo(Vector2, Int32, Int32, StateManager, SVsIModel)

Constructor of `TowerInfo`

Declaration

```
public TowerInfo(Vector2 position, int height, int width, StateManager stateManager, SVsIModel model)
```

Parameters

TYPE	NAME	DESCRIPTION
Microsoft.Xna.Framework.Vector2	position	Position
System.Int32	height	Height of the panel
System.Int32	width	Width of the panel

TYPE	NAME	DESCRIPTION
StateManager	stateManager	State manager to get the state from
SVsIModel	model	Model to get the current tower from

Properties

SellButton

Sell button

Declaration

```
public Button SellButton { get; }
```

Property Value

TYPE	DESCRIPTION
Button	Sell button

UpgradeButton

Upgrade Button

Declaration

```
public Button UpgradeButton { get; }
```

Property Value

TYPE	DESCRIPTION
Button	

Methods

Draw(SpriteBatch)

Draws the tower info panel

Declaration

```
public override void Draw(SpriteBatch spriteBatch)
```

Parameters

TYPE	NAME	DESCRIPTION
Microsoft.Xna.Framework.Graphics.SpriteBatch	spriteBatch	Spritebatch to draw to

Overrides

BasePanel.Draw(SpriteBatch)

Update(GameTime)

Updates the info based on the currently selected tower

Declaration

```
public override void Update(GameTime gameTime)
```

Parameters

TYPE	NAME	DESCRIPTION
Microsoft.Xna.Framework.GameTime	gameTime	Game time

Overrides

Component.Update(GameTime)

Namespace SpaceVsInvaders.View.Scenes

Classes

GameScene

Main game scene

LogoScene

Logo scene displayed in the beginning

MainMenuScene

Main menu scene

PauseScene

Pause scene

Scene

Base scene class

Enums

Difficulty

Difficulty enum

Enum Difficulty

Difficulty enum

Namespace: SpaceVsInvaders.View.Scenes

Assembly: SpaceVsInvaders.dll

Syntax

```
public enum Difficulty
```

Fields

NAME	DESCRIPTION
Hard	Hard difficulty
Normal	Normal difficulty

Class GameScene

Main game scene

Inheritance

System.Object
Scene
GameScene

Inherited Members

Scene.Width
Scene.Height
System.Object.Equals(System.Object)
System.Object.Equals(System.Object, System.Object)
System.Object.GetHashCode()
System.Object.GetType()
System.Object.MemberwiseClone()
System.Object.ReferenceEquals(System.Object, System.Object)
System.Object.ToString()

Namespace: SpaceVsInvaders.View.Scenes

Assembly: SpaceVsInvaders.dll

Syntax

```
public class GameScene : Scene
```

Constructors

GameScene(Int32, Int32)

Constructor of `GameScene`

Declaration

```
public GameScene(int width, int height)
```

Parameters

TYPE	NAME	DESCRIPTION
System.Int32	width	Window width
System.Int32	height	Window Height

Methods

Draw(SpriteBatch)

This is called when the game should draw itself.

Declaration

```
public override void Draw(SpriteBatch spriteBatch)
```

Parameters

TYPE	NAME	DESCRIPTION
Microsoft.Xna.Framework.Graphics.SpriteBatch	spriteBatch	Spritebatch to draw to

Overrides

Scene.Draw(SpriteBatch)

LoadContent()

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

Declaration

```
public override void LoadContent()
```

Overrides

Scene.LoadContent()

NewGame(Difficulty)

Start a new game

Declaration

```
public void NewGame(Difficulty difficulty)
```

Parameters

TYPE	NAME	DESCRIPTION
Difficulty	difficulty	Difficulty of the new game

UnloadContent()

UnloadContent will be called once per game and is the place to unload game-specific content.

Declaration

```
public override void UnloadContent()
```

Overrides

Scene.UnloadContent()

Update(GameTime)

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

Declaration

```
public override void Update(GameTime gameTime)
```

Parameters

TYPE	NAME	DESCRIPTION
Microsoft.Xna.Framework.GameTime	gameTime	Provides a snapshot of timing values.

Overrides

Scene.Update(GameTime)

Events

ExitToMainMenu

Switch to the main menu event

Declaration

```
public event EventHandler ExitToMainMenu
```

Event Type

TYPE	DESCRIPTION
System.EventHandler	

OpenPauseMenu

The pause menu should be opened event

Declaration

```
public event EventHandler OpenPauseMenu
```

Event Type

TYPE	DESCRIPTION
System.EventHandler	

Class LogoScene

Logo scene displayed in the beginning

Inheritance

System.Object
Scene
LogoScene

Inherited Members

Scene.Width
Scene.Height
System.Object.Equals(System.Object)
System.Object.Equals(System.Object, System.Object)
System.Object.GetHashCode()
System.Object.GetType()
System.Object.MemberwiseClone()
System.Object.ReferenceEquals(System.Object, System.Object)
System.Object.ToString()

Namespace: SpaceVsInvaders.View.Scenes

Assembly: SpaceVsInvaders.dll

Syntax

```
public class LogoScene : Scene
```

Constructors

LogoScene(Int32, Int32)

Constructor of `LogoScene`

Declaration

```
public LogoScene(int width, int height)
```

Parameters

TYPE	NAME	DESCRIPTION
System.Int32	width	Window width
System.Int32	height	Window Height

Methods

Draw(SpriteBatch)

Draws the logo scene to the spritebatch

Declaration

```
public override void Draw(SpriteBatch spriteBatch)
```

Parameters

TYPE	NAME	DESCRIPTION
Microsoft.Xna.Framework.Graphics.SpriteBatch	spriteBatch	Spritebatch

Overrides

Scene.Draw(SpriteBatch)

LoadContent()

Loads content, runs once per program run

Declaration

```
public override void LoadContent()
```

Overrides

Scene.LoadContent()

UnloadContent()

Handles the unloading of content

Declaration

```
public override void UnloadContent()
```

Overrides

Scene.UnloadContent()

Update(GameTime)

Updates the logo animation

Declaration

```
public override void Update(GameTime gameTime)
```

Parameters

TYPE	NAME	DESCRIPTION
Microsoft.Xna.Framework.GameTime	gameTime	gametime

Overrides

Scene.Update(GameTime)

Events

End

The logo animation has ended event

Declaration

```
public event EventHandler End
```

Event Type

TYPE	DESCRIPTION
System.EventHandler	

Class MainMenuScene

Main menu scene

Inheritance

System.Object
Scene
MainMenuScene

Inherited Members

Scene.Width
Scene.Height
System.Object.Equals(System.Object)
System.Object.Equals(System.Object, System.Object)
System.Object.GetHashCode()
System.Object.GetType()
System.Object.MemberwiseClone()
System.Object.ReferenceEquals(System.Object, System.Object)
System.Object.ToString()

Namespace: SpaceVsInvaders.View.Scenes

Assembly: SpaceVsInvaders.dll

Syntax

```
public class MainMenuScene : Scene
```

Constructors

MainMenuScene(Int32, Int32)

Constructor of `MainMenuScene`

Declaration

```
public MainMenuScene(int width, int height)
```

Parameters

TYPE	NAME	DESCRIPTION
System.Int32	width	Window Width
System.Int32	height	Window Height

Methods

Draw(SpriteBatch)

Draws the scene to the spritebatch

Declaration

```
public override void Draw(SpriteBatch spriteBatch)
```

Parameters

TYPE	NAME	DESCRIPTION
Microsoft.Xna.Framework.Graphics.SpriteBatch	spriteBatch	Spritebatch

Overrides

Scene.Draw(SpriteBatch)

LoadContent()

Loads content, runs once per program run

Declaration

```
public override void LoadContent()
```

Overrides

Scene.LoadContent()

UnloadContent()

Handles the unloading of content

Declaration

```
public override void UnloadContent()
```

Overrides

Scene.UnloadContent()

Update(GameTime)

Updates the scene

Declaration

```
public override void Update(GameTime gameTime)
```

Parameters

TYPE	NAME	DESCRIPTION
Microsoft.Xna.Framework.GameTime	gameTime	gametime

Overrides

Scene.Update(GameTime)

Events

Exit

The program should exit event

Declaration

```
public event EventHandler Exit
```

Event Type

TYPE	DESCRIPTION
System.EventHandler	

NewGame

New game should be started with the difficulty

Declaration

```
public event EventHandler<Difficulty> NewGame
```

Event Type

TYPE	DESCRIPTION
System.EventHandler<Difficulty>	

Class PauseScene

Pause scene

Inheritance

System.Object

Scene

PauseScene

Inherited Members

Scene.Width

Scene.Height

System.Object.Equals(System.Object)

System.Object.Equals(System.Object, System.Object)

System.Object.GetHashCode()

System.Object.GetType()

System.Object.MemberwiseClone()

System.Object.ReferenceEquals(System.Object, System.Object)

System.Object.ToString()

Namespace: SpaceVsInvaders.View.Scenes

Assembly: SpaceVsInvaders.dll

Syntax

```
public class PauseScene : Scene
```

Constructors

PauseScene(Int32, Int32)

Constructor of `PauseScene`

Declaration

```
public PauseScene(int width, int height)
```

Parameters

TYPE	NAME	DESCRIPTION
System.Int32	width	Window Width
System.Int32	height	Window Height

Methods

Draw(SpriteBatch)

Draws the scene to the spritebatch

Declaration

```
public override void Draw(SpriteBatch spriteBatch)
```

Parameters

TYPE	NAME	DESCRIPTION
Microsoft.Xna.Framework.Graphics.SpriteBatch	spriteBatch	Spritebatch

Overrides

Scene.Draw(SpriteBatch)

LoadContent()

Loads content, runs once per program run

Declaration

```
public override void LoadContent()
```

Overrides

Scene.LoadContent()

UnloadContent()

Handles the unloading of content

Declaration

```
public override void UnloadContent()
```

Overrides

Scene.UnloadContent()

Update(GameTime)

Updates the scene

Declaration

```
public override void Update(GameTime gameTime)
```

Parameters

TYPE	NAME	DESCRIPTION
Microsoft.Xna.Framework.GameTime	gameTime	gametime

Overrides

Scene.Update(GameTime)

Events

Exit

The proram should exit event

Declaration

```
public event EventHandler Exit
```

Event Type

TYPE	DESCRIPTION
System.EventHandler	

ExitToMainMenu

Switch the scene to the main menu event

Declaration

```
public event EventHandler ExitToMainMenu
```

Event Type

TYPE	DESCRIPTION
System.EventHandler	

Resume

The game should be resumed event

Declaration

```
public event EventHandler Resume
```

Event Type

TYPE	DESCRIPTION
System.EventHandler	

Class Scene

Base scene class

Inheritance

System.Object
Scene
GameScene
LogoScene
MainMenuScene
PauseScene

Inherited Members

System.Object.Equals(System.Object)
System.Object.Equals(System.Object, System.Object)
System.Object.GetHashCode()
System.Object.GetType()
System.Object.MemberwiseClone()
System.Object.ReferenceEquals(System.Object, System.Object)
System.Object.ToString()

Namespace: SpaceVsInvaders.View.Scenes

Assembly: SpaceVsInvaders.dll

Syntax

```
public abstract class Scene
```

Constructors

Scene(Int32, Int32)

Constructor of `Scene`

Declaration

```
public Scene(int width, int height)
```

Parameters

TYPE	NAME	DESCRIPTION
System.Int32	width	Width of the scene
System.Int32	height	Height of the scene

Properties

Height

Height of the scene

Declaration

```
protected int Height { get; }
```

Property Value

TYPE	DESCRIPTION
System.Int32	Height of the scene

Width

Width of the scene

Declaration

```
protected int Width { get; }
```

Property Value

TYPE	DESCRIPTION
System.Int32	Width of the scene

Methods

Draw(SpriteBatch)

Draws the scene to the spritebatch

Declaration

```
public abstract void Draw(SpriteBatch spriteBatch)
```

Parameters

TYPE	NAME	DESCRIPTION
Microsoft.Xna.Framework.Graphics.SpriteBatch	spriteBatch	Spritebatch

LoadContent()

Loads content, runs once per program run

Declaration

```
public abstract void LoadContent()
```

UnloadContent()

Handles the unloading of content

Declaration

```
public abstract void UnloadContent()
```

Update(GameTime)

Updates the scene

Declaration

```
public abstract void Update(GameTime gameTime)
```

Parameters

TYPE	NAME	DESCRIPTION
Microsoft.Xna.Framework.GameTime	gameTime	gametime

Namespace SpaceVsInvaders.Tests

Classes

UnitTest1

WStest

Class UnitTest1

Inheritance

System.Object
UnitTest1

Inherited Members

System.Object.Equals(System.Object)
System.Object.Equals(System.Object, System.Object)
System.Object.GetHashCode()
System.Object.GetType()
System.Object.MemberwiseClone()
System.Object.ReferenceEquals(System.Object, System.Object)
System.Object.ToString()

Namespace: SpaceVsInvaders.Tests

Assembly: SpaceVsInvaders.Tests.dll

Syntax

```
public class UnitTest1
```

Constructors

UnitTest1()

Declaration

```
public UnitTest1()
```

Methods

DamageTower()

Declaration

```
[Fact]
public void DamageTower()
```

DamageTower2()

Declaration

```
[Fact]
public void DamageTower2()
```

EnemyDamage()

Declaration

```
[Fact]
public void EnemyDamage()
```

EnemyDamage2()

Declaration

```
[Fact]
public void EnemyDamage2()
```

EnemyMovement()

Declaration

```
[Fact]
public void EnemyMovement()
```

EnemyMovement2()

Declaration

```
[Fact]
public void EnemyMovement2()
```

GoldTower()

Declaration

```
[Fact]
public void GoldTower()
```

HealTower()

Declaration

```
[Fact]
public void HealTower()
```

PlacingTowers()

Declaration

```
[Fact]
public void PlacingTowers()
```

SellTower()

Declaration

```
[Fact]
public void SellTower()
```

UpgradeTower()

Declaration

```
[Fact]
public void UpgradeTower()
```

Class WStest

Inheritance

System.Object
WStest

Inherited Members

System.Object.Equals(System.Object)
System.Object.Equals(System.Object, System.Object)
System.Object.GetHashCode()
System.Object.GetType()
System.Object.MemberwiseClone()
System.Object.ReferenceEquals(System.Object, System.Object)
System.Object.ToString()

Namespace: SpaceVsInvaders.Tests

Assembly: SpaceVsInvaders.Tests.dll

Syntax

```
public class WStest
```

Constructors

WStest()

Declaration

```
public WStest()
```

Methods

DamageCatastrophe_Damage()

Declaration

```
[Fact]
public void DamageCatastrophe_Damage()
```

DamageCatastrophe_Range()

Declaration

```
[Fact]
public void DamageCatastrophe_Range()
```

HealeCatastrophe_Heal()

Declaration

```
[Fact]
public void HealeCatastrophe_Heal()
```

HealeCatastrophe_Range()

Declaration

```
[Fact]
public void HealeCatastrophe_Range()
```

WaveSpawnerTest()

Declaration

```
[Fact]
public void WaveSpawnerTest()
```