

Namespace FreakingMath

Classes

[BrainScript](#)

[GameControl](#)

[GameDataManager](#)

[GameOverControl](#)

[Hand](#)

[HintControl](#)

[HomeControl](#)

[LifeControl](#)

[SplashControl](#)

[TimerBarControl](#)

Enums

[MathOperator](#)

Class BrainScript

Namespace: [FreakingMath](#)

Assembly: Assembly-CSharp.dll

```
public class BrainScript : MonoBehaviour
```

Inheritance

object ← Object ← Component ← Behaviour ← MonoBehaviour ← BrainScript

Fields

correctResult

```
private int correctResult
```

Field Value

int

mathOperator

```
private MathOperator mathOperator
```

Field Value

[MathOperator](#)

operandA

```
private int operandA
```

Field Value

int

operandB

```
private int operandB
```

Field Value

int

question

```
private string question
```

Field Value

string

results

```
private int[] results
```

Field Value

int[]

Methods

GenerateFakeResults()

```
private void GenerateFakeResults()
```

GenerateQuiz()

```
public void GenerateQuiz()
```

GetCorrectResult()

```
public int GetCorrectResult()
```

Returns

int

GetCorrectResultIndex()

```
public int GetCorrectResultIndex()
```

Returns

int

GetQuestion()

```
public string GetQuestion()
```

Returns

string

GetResults()

```
public int[] GetResults()
```

Returns

`int[]`

Shuffle(int[])

```
public void Shuffle(int[] arr)
```

Parameters

`arr` `int[]`

Class GameControl

Namespace: [FreakingMath](#)

Assembly: Assembly-CSharp.dll

```
public class GameControl : MonoBehaviour
```

Inheritance

object ← Object ← Component ← Behaviour ← MonoBehaviour ← GameControl

Fields

brainScript

```
[SerializeField]  
private BrainScript brainScript
```

Field Value

[BrainScript](#)

correctResult

```
private int correctResult
```

Field Value

int

countCorrect

```
private int countCorrect
```

Field Value

int

gameDataManager

```
private GameDataManager gameDataManager
```

Field Value

[GameDataManager](#)

gameOverControl

```
[SerializeField]
```

```
private GameOverControl gameOverControl
```

Field Value

[GameOverControl](#)

hands

```
[SerializeField]
```

```
private GameObject[] hands
```

Field Value

GameObject[]

imageBonusLife

```
[SerializeField]
```

```
private Image imageBonusLife
```

Field Value

Image

lifeControl

```
[SerializeField]  
private LifeControl lifeControl
```

Field Value

[LifeControl](#)

score

```
private int score
```

Field Value

int

textAnswer1

```
[SerializeField]  
private Text textAnswer1
```

Field Value

Text

textAnswer2

```
[SerializeField]  
private Text textAnswer2
```


Field Value

Text

textAnswer3

```
[SerializeField]  
private Text textAnswer3
```

Field Value

Text

textAnswer4

```
[SerializeField]  
private Text textAnswer4
```

Field Value

Text

textQuestion

```
[SerializeField]  
private Text textQuestion
```

Field Value

Text

textScore

```
[SerializeField]
```

```
private Text textScore
```

Field Value

Text

timerBarController

```
[SerializeField]  
private TimerBarController timerBarController
```

Field Value

[TimerBarController](#)

Methods

GenerateQuestion()

```
private void GenerateQuestion()
```

GiveAnswer()

```
public void GiveAnswer()
```

HandleClick(int)

```
public void HandleClick(int index)
```

Parameters

index int

ResetBonusLife()

```
public void ResetBonusLife()
```

ShowGameOver()

```
public void ShowGameOver()
```

Start()

```
private void Start()
```

Class GameManager

Namespace: [FreakingMath](#)

Assembly: Assembly-CSharp.dll

```
public class GameManager : MonoBehaviour
```

Inheritance

object ← Object ← Component ← Behaviour ← MonoBehaviour ← GameManager

Fields

dataManager

```
public static GameManager dataManager
```

Field Value

[GameManager](#)

Properties

HighScore

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

Property Value

int

Methods

Awake()

```
private void Awake()
```

OnDestroy()

```
private void OnDestroy()
```

Start()

```
private void Start()
```

Class GameOverControl

Namespace: [FreakingMath](#)

Assembly: Assembly-CSharp.dll

```
public class GameOverControl : MonoBehaviour
```

Inheritance

object ← Object ← Component ← Behaviour ← MonoBehaviour ← GameOverControl

Fields

lifeControl

```
[SerializeField]  
private LifeControl lifeControl
```

Field Value

[LifeControl](#)

textScore

```
[SerializeField]  
private Text textScore
```

Field Value

Text

timerBarController

```
[SerializeField]  
private TimerBarController timerBarController
```

Field Value

[TimerBarControl](#)

Methods

ClickNoThankYou()

```
public void ClickNoThankYou()
```

ClickToContinue()

```
public void ClickToContinue()
```

SetScore(int)

```
public void SetScore(int score)
```

Parameters

score int

Class Hand

Namespace: [FreakingMath](#)

Assembly: Assembly-CSharp.dll

```
public class Hand : MonoBehaviour
```

Inheritance

object ← Object ← Component ← Behaviour ← MonoBehaviour ← Hand

Methods

OnAnimationCompleted()

```
public void OnAnimationCompleted()
```


Class HintControl

Namespace: [FreakingMath](#)

Assembly: Assembly-CSharp.dll

```
public class HintControl : MonoBehaviour
```

Inheritance

object ← Object ← Component ← Behaviour ← MonoBehaviour ← HintControl

Fields

gameControl

```
[SerializeField]  
private GameController gameControl
```

Field Value

[GameControl](#)

timerControl

```
[SerializeField]  
private TimerBarController timerControl
```

Field Value

[TimerBarController](#)

Methods

DoGiveAnswer()

```
public void DoGiveAnswer()
```

DoPauseTimer()

```
public void DoPauseTimer()
```

Class HomeControl

Namespace: [FreakingMath](#)

Assembly: Assembly-CSharp.dll

```
public class HomeControl : MonoBehaviour
```

Inheritance

object ← Object ← Component ← Behaviour ← MonoBehaviour ← HomeControl

Fields

gameDataManager

```
private GameDataManager gameDataManager
```

Field Value

[GameDataManager](#)

textScore

```
[SerializeField]  
private Text textScore
```

Field Value

Text

Methods

ClickToGameScene()

```
public void ClickToGameScene()
```

Start()

```
private void Start()
```

Class LifeControl

Namespace: [FreakingMath](#)

Assembly: Assembly-CSharp.dll

```
public class LifeControl : MonoBehaviour
```

Inheritance

object ← Object ← Component ← Behaviour ← MonoBehaviour ← LifeControl

Fields

animator

```
[SerializeField]  
private Animator animator
```

Field Value

Animator

textExtraLife

```
[SerializeField]  
private Text textExtraLife
```

Field Value

Text

toggles

```
[SerializeField]  
private Toggle[] toggles
```

Field Value

Toggle[]

totalLife

```
[SerializeField]  
private int totalLife
```

Field Value

int

Methods

AddLife()

```
public void AddLife()
```

IsDie()

```
public bool IsDie()
```

Returns

bool

RefreshExtraLife()

```
private void RefreshExtraLife()
```

RemoveLife()

```
public void RemoveLife()
```

ResetLife()

```
public void ResetLife()
```

Start()

```
private void Start()
```

StillHasLife()

```
public bool StillHasLife()
```

Returns

bool

_addLife()

```
private IEnumerator _addLife()
```

Returns

IEnumerator

Enum MathOperator

Namespace: [FreakingMath](#)

Assembly: Assembly-CSharp.dll

```
internal enum MathOperator
```

Fields

DEVIDE = 3

MINUS = 1

MULTIPLY = 2

PLUS = 0

Class SplashControl

Namespace: [FreakingMath](#)

Assembly: Assembly-CSharp.dll

```
public class SplashControl : MonoBehaviour
```

Inheritance

object ← Object ← Component ← Behaviour ← MonoBehaviour ← SplashControl

Fields

delayBeforeHome

```
[SerializeField]  
private float delayBeforeHome
```

Field Value

float

Methods

Start()

```
private void Start()
```

WaitingAndLoad()

```
private IEnumerator WaitingAndLoad()
```

Returns

Class TimerBarControl

Namespace: [FreakingMath](#)

Assembly: Assembly-CSharp.dll

```
public class TimerBarControl : MonoBehaviour
```

Inheritance

object ← Object ← Component ← Behaviour ← MonoBehaviour ← TimerBarControl

Fields

durationPerQuestion

```
[SerializeField]  
private float durationPerQuestion
```

Field Value

float

gameControl

```
[SerializeField]  
private GameController gameControl
```

Field Value

[GameControl](#)

imageTimerBar

```
[SerializeField]  
private Image imageTimerBar
```

Field Value

Image

isTimeOut

```
private bool isTimeOut
```

Field Value

bool

lifeControl

```
[SerializeField]  
private LifeControl lifeControl
```

Field Value

[LifeControl](#)

timeCounter

```
private float timeCounter
```

Field Value

float

Methods

Pause()

```
public void Pause()
```

RemoveLifeAndResetTimer()

```
private void RemoveLifeAndResetTimer()
```

Reset()

```
public void Reset()
```

Start()

```
private void Start()
```

Update()

```
private void Update()
```