

# Namespace UnityMathQuiz

## Classes

[LifeControl](#)

[SplashControl](#)

# Class LifeControl

Namespace: [UnityMathQuiz](#)

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()

Updates the UI element that displays extra lives beyond the default limit.

```
private void RefreshExtraLife()
```

Remarks

If the `totalLife` exceeds 5, this method calculates the extra lives and updates the `textExtraLife` UI text to show the surplus in the format "+X". If `totalLife` is 5 or less, the extra life text is cleared.

## RemoveLife()

```
public void RemoveLife()
```

## ResetLife()

```
public void ResetLife()
```

## Start()

```
private void Start()
```

## StillHasLife()

```
public bool StillHasLife()
```

## Returns

bool

## \_addLife()

```
private IEnumerator _addLife()
```

## Returns

IEnumerator



# Class SplashControl

Namespace: [UnityMathQuiz](#)

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()

This is a private function example.

```
private IEnumerator WaitingAndLoad()
```

Returns

IEnumerator