JULY 3, 2020

# SPINNER

## FREE SPIN

MUHAMMAD TAHA
BRILLIANT GAMEZ
Kalma Chowk

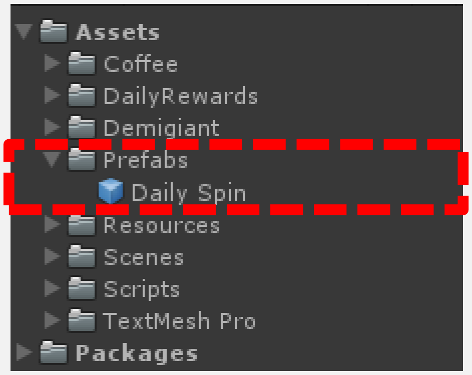
# TABLE OF CONTENTS

## REQUIREMENT

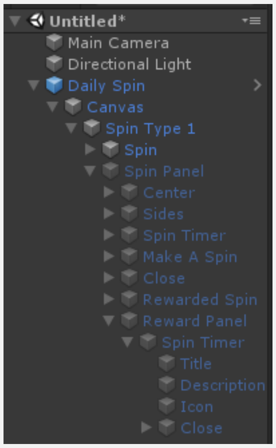
- Daily Reward
- DO Tween
- Coffee UI Shiny
- Text Mesh Pro

## HOW TO USE?

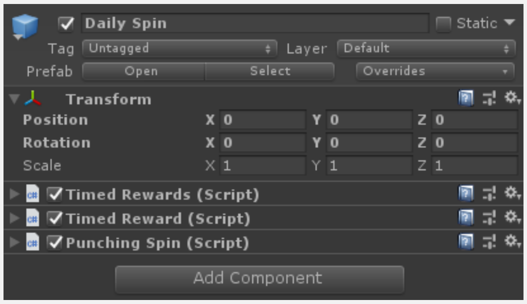
1. Drag and drop **prefab** from Prefabs Folder.



2. After dragging in hierarchy your hierarchy looks like this.



3. Select **Daily Spin**. The Inspector look like this.



☑ Daily Spin □ Static

Tag Untagged Layer Default

Prefab Open Select Overrides

Transform

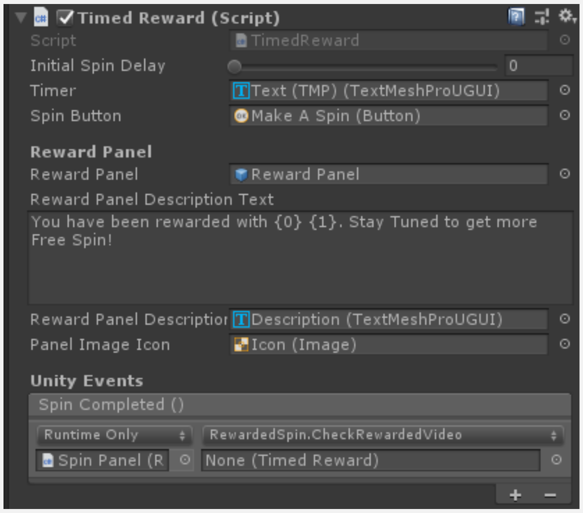
| | | | |
|---|---|---|---|
| Position | X 0 | Y 0 | Z 0 |
| Rotation | X 0 | Y 0 | Z 0 |
| Scale | X 1 | Y 1 | Z 1 |

☑ Timed Rewards (Script)

☑ Timed Reward (Script)

☑ Punching Spin (Script)

4. The timed rewards inspector looks same as daily reward.
5. The timed reward script by Brilliant Gamez look like this.



Timed Reward (Script)

Script TimedReward

Initial Spin Delay 0

Timer Text (TMP) (TextMeshProUGUI)

Spin Button Make A Spin (Button)

Reward Panel

Reward Panel Reward Panel

Reward Panel Description Text

You have been rewarded with {0} {1}. Stay Tuned to get more Free Spin!

Reward Panel Descriptior Description (TextMeshProUGUI)

Panel Image Icon Icon (Image)

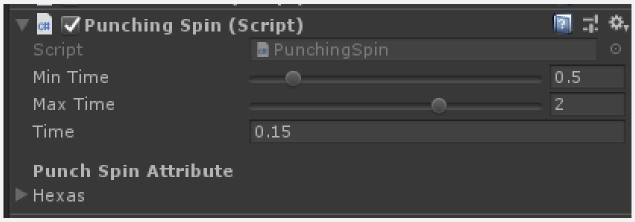
Unity Events

Spin Completed ()

Runtime Only RewardedSpin.CheckRewardedVideo

Spin Panel (R None (Timed Reward)

6. The next script is Punching Spin. This script handles the animation of spinner.



Punching Spin (Script)

Script PunchingSpin

Min Time 0.5

Max Time 2

Time 0.15

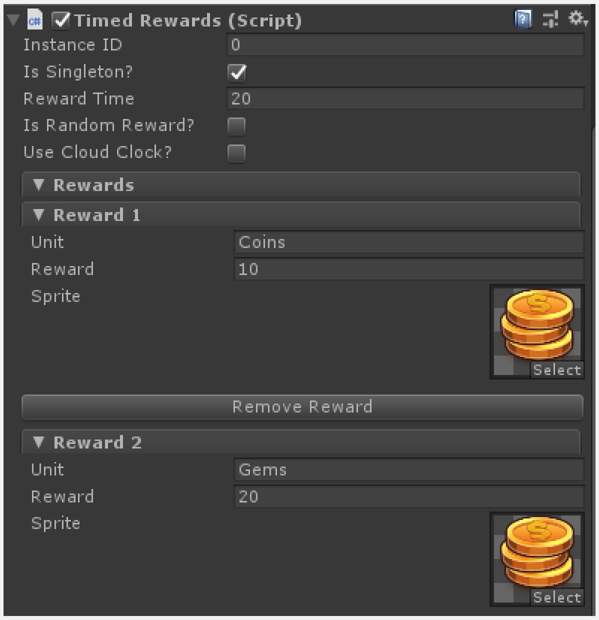
Punch Spin Attribute

   a. This script tackle punching animation of spinner.
   b. You can view that in Scenes -> DailySpin Demo.
   c. You can remove this script and add new spin animation type. Follow the Link.

7. When you play a game you will see that the prefab will go to Don't DestroyOnLoad. So cross reference will throw exception. REMINDER.

# MANUAL

## TIMED REWARDS



☑ Timed Rewards (Script)

Instance ID: 0

Is Singleton?: ☑

Reward Time: 20

Is Random Reward?: □

Use Cloud Clock?: □

▼ Rewards

▼ Reward 1

Unit: Coins

Reward: 10

Sprite

▼ Reward 2

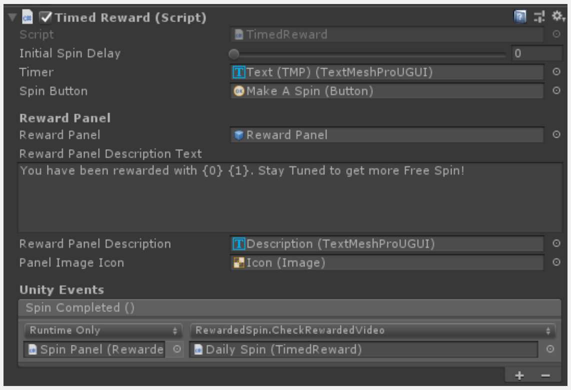
Unit: Gems

Reward: 20

Sprite

- **Reward Time**: Time for next reward/Spin.
- **Rewards**: These contains the information of reward.
    - **Unit**: The identifier used in code to give reward.
    - **Reward**: Amount given to user. Can be amount, Player index anything.
    - **Sprite**: Icon going to show on reward given panel screen.

## TIMED REWARD



☑ Timed Reward (Script)

Script: TimedReward

Initial Spin Delay: 0

Timer: Text (TMP) (TextMeshProUGUI)

Spin Button: Make A Spin (Button)

**Reward Panel**

Reward Panel: Reward Panel

Reward Panel Description Text

You have been rewarded with {0} {1}. Stay Tuned to get more Free Spin!

Reward Panel Description: Description (TextMeshProUGUI)

Panel Image Icon: Icon (Image)

**Unity Events**

Spin Completed ()

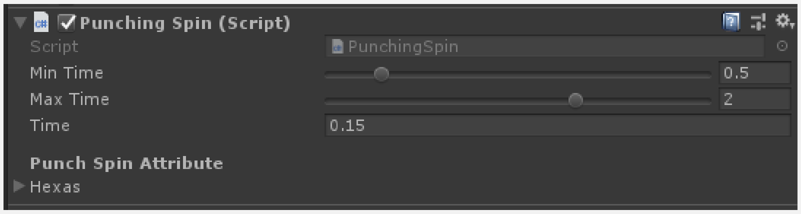
Runtime Only: RewardedSpin.CheckRewardedVideo

Spin Panel (Rewarde: Daily Spin (TimedReward)

- **Initial Spin Delay**: The time to wait before spin. Useful when new window popup with spinner run automatically.

- **Timer**: The text component of timer. NOTE: can be switched on Unity Text.
- **Spin Button:** The reference of button which make spinner to animate.
- **Reward Panel:** Panel which opens when spinner end and show user the reward.
- **Description Text:** The text shown in the reward panel description.
    - **{0}:** The amount of reward provided in Timed Rewards.
    - **{1}:** The name of reward provided in Timed Rewards.
- **Reward Panel Description:** The description text component UI. NOTE: can be switched on Unity Text.
- **Panel Image Icon:** The icon will show on panel reward. This image data will update by the Timed Rewards.
- **Spin Completed:** This Unity event which will calls on the spin completed. As you can see the argument passed in this event is check the rewarded video for spin.

## PUNCHING SPIN



☑ Punching Spin (Script)

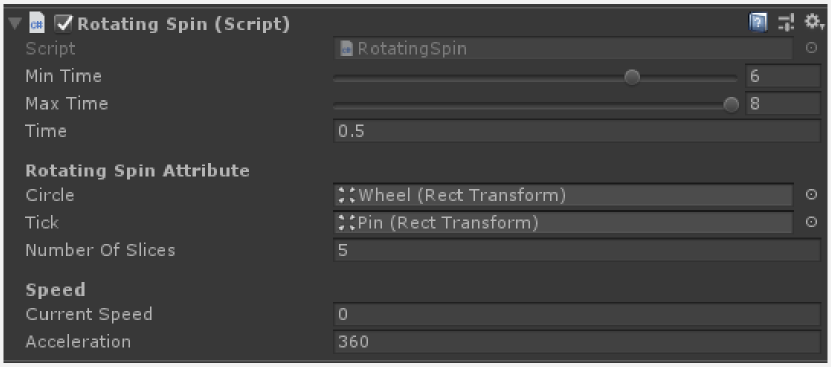
| | |
|---|---|
| Script | PunchingSpin |
| Min Time | 0.5 |
| Max Time | 2 |
| Time | 0.15 |

Punch Spin Attribute

▶ Hexas

- **Min Time:** Minimum time to spin.
- **Max Time:** Maximum time to spin.
- **Time:** Time of punch animation.
- **Hexas:** These are the position of spin. i.e. side position of spinner. These hexas will do punching animation. For clearance try demo scene.

## ROTATING SPIN



Rotating Spin (Script)

Script: RotatingSpin

Min Time: 6

Max Time: 8

Time: 0.5

**Rotating Spin Attribute**

Circle: Wheel (Rect Transform)

Tick: Pin (Rect Transform)

Number Of Slices: 5

**Speed**

Current Speed: 0

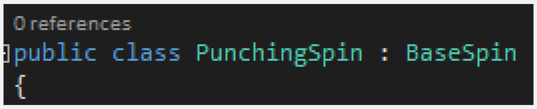
Acceleration: 360

- **Min Time:** Minimum time to spin.
- **Max Time:** Maximum time to spin.
- **Time:** Time of punch animation.
- **Circle:** The Rotating wheel of spinner. For further explanation check the prefab and Demo scene.
- **Tick:** The pin pointing at the top of Wheel.
- **Number of Slices:** The slices in the spinning wheel.
- **Current Speed:** The current speed of the wheel. Only used for debugging.
- **Acceleration:** The speed of wheel in which a wheel is going to rotate.

# SCRIPTING API

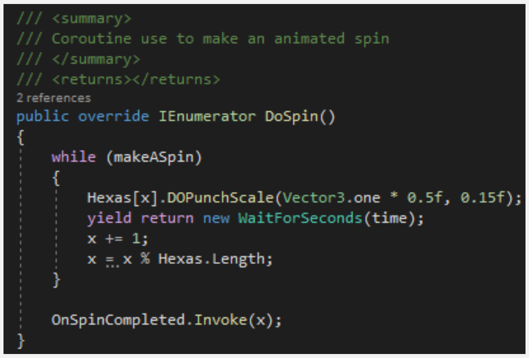
## HOW TO CREATE NEW SPIN ANIMATION?

- To create new spin animation, make a new script.
- I will give you the example of punching animation. When script is created open in code editor.
- Extend this class with Base Spin.



```
0 references
public class PunchingSpin : BaseSpin
{
```

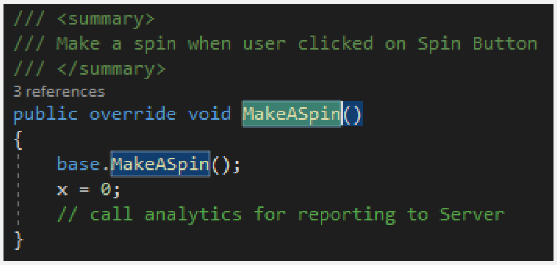
- Implement abstract method named as `DoSpin().`
  - This method is responsible of handling all the animations. You can write your own **custom animation behavior** is this function.



```
/// <summary>
/// Coroutine use to make an animated spin
/// </summary>
/// <returns></returns>
2 references
public override IEnumerator DoSpin()
{
    while (makeASpin)
    {
        Hexas[x].DOPunchScale(Vector3.one * 0.5f, 0.15f);
        yield return new WaitForSeconds(time);
        x += 1;
        x = x % Hexas.Length;
    }

    OnSpinCompleted.Invoke(x);
}
```

  - In this you can see that DoSpin is IEnumerator. Which means this needs a delay to tackle with.
  - The while loop has a variable makeASpin which is base class variable and gets **true** when user click on Spin Button and **false** when timer of spin ends.
  - I the while loop the tweening is happening.
  - After loop **OnSpinCompleted Event** called, which shows that the spin finished and now ready to do anything on spin end e.g. Check Rewarded Video.
- To give **analytics** it's better to implement a virtual function named as `MakeASpin().`



```
/// <summary>
/// Make a spin when user clicked on Spin Button
/// </summary>
3 references
public override void MakeASpin()
{
    base.MakeASpin();
    x = 0;
    // call analytics for reporting to Server
}
```

- The base.MakeASpin(); line is necessary as this call the basic functionality of spinner. Under that line you can write anything you want to do. Like in this case I make my X variable to zero. And comment also says you **report analytics to server**.

## HOW TO ADD NEW REWARD?

- Open **Timed Reward** script in Code Editor which contains information of panel, description, and Unity Event.
- Go to `OnSpinCompleted(int index).`
- **Integration Timed Reward** script is attached to the main prefab. Open that script and check the `OnClaimPrizeTimedRewards(int index)` contains few commented lines.
- NOTE: DON'T CHANGE ANYTHING ELSE IN THIS CODE ELSE YOU UNDERSTAND EACH AND EVERYTHING.