AGGP 225  
LAB 2

READINGS:

Flyweight Pattern  
<https://gameprogrammingpatterns.com/flyweight.html>

Scriptable Object  
API <https://docs.unity3d.com/ScriptReference/ScriptableObject.html>  
Unity Learning <https://learn.unity.com/tutorial/introduction-to-scriptable-objects>  
Brackys Tutorial Video <https://www.youtube.com/watch?v=aPXvoWVabPY>  
Architecting With <https://unity.com/how-to/architect-game-code-scriptable-objects>

Lab Assignment:

Design:

Create a Card game that is based on Rock-Paper-Scissors. But with more options.

There are two ways to build your gameplay:

* Each Card has an equal number of other cards that it beats or is beaten to
  + Examples <https://www.umop.com/rps.htm>
* Each Card has a Super Variant that will beat 2 others, but is beaten by one other.
  + Example:
    - The Bomb bets Rock and Paper, but loses to Scissors

You are required to have a minimum 6 different cards in your game.

We are going to use the Term “Challenge” to determine the result of one card being played against the other.

Required Components:

* Create a Enum called “ChallengeResult”
  + It will have 3 values:
    - Wins
    - Loses
    - Draw
* Each Card is defined with a Scripted Object
  + Name this class “CardDefination”
  + Each card should a list that defines who it loses to and who it wins to.
  + Add a Method
    - Public ChallengeResult PerformChallenge(Scripted Object)
    - This is where you’ll put the gameplay logic to determine who wins or loses.

* Create a Prefab called “CardObject”
  + This is a Game Object that will be in the scene
  + Write a method that sets up the card object based on the data in the Scriptable Object.
* Create a Prefab called “CardUI”
  + This is a game object rendered on a canvas.
  + Write a method that sets up the card object based on the data in the Scriptable Object.
* Once you have you created these required elements, Create a simple mini game between a human player and the CPU.
* Keep track of the number of Wins per player and number of Draws.
  + This should be displayed on Screen.

EXTRA CREDIT:

There are many different ways expand upon this structure.

Examples

Make an Enum for your card types (Rock, Paper, Scissors, ect) and add a variable in your CardDefination. Then Create Muiltiple card for each type.

Expand the Card Defination with extra rules

When this card is played… ect

When this card loses… ect