# Bethany Woodruff

## Personal Improvement

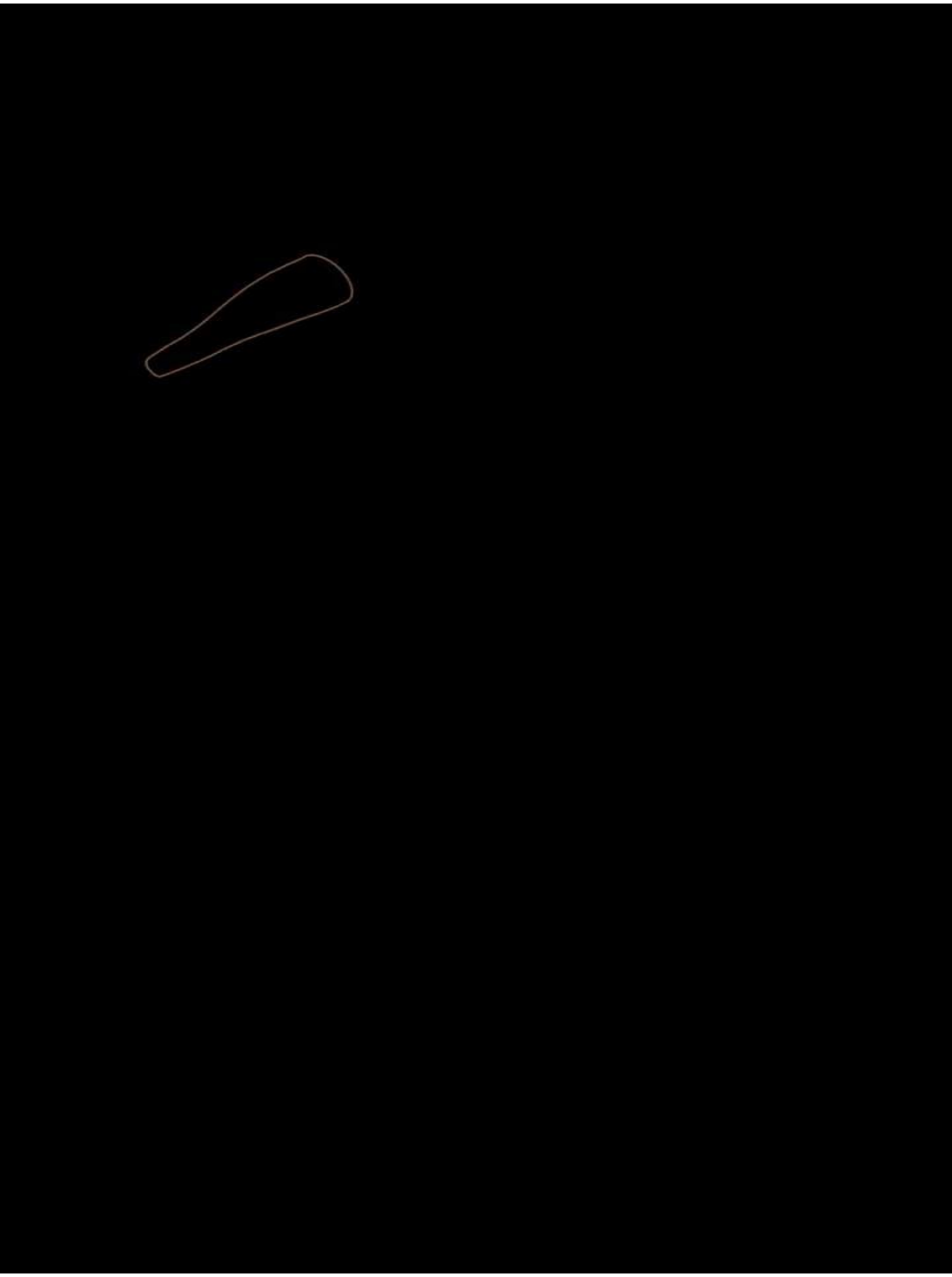
Or “Woodruf” according to Teams...

Or “Ylst” according to my marriage license...

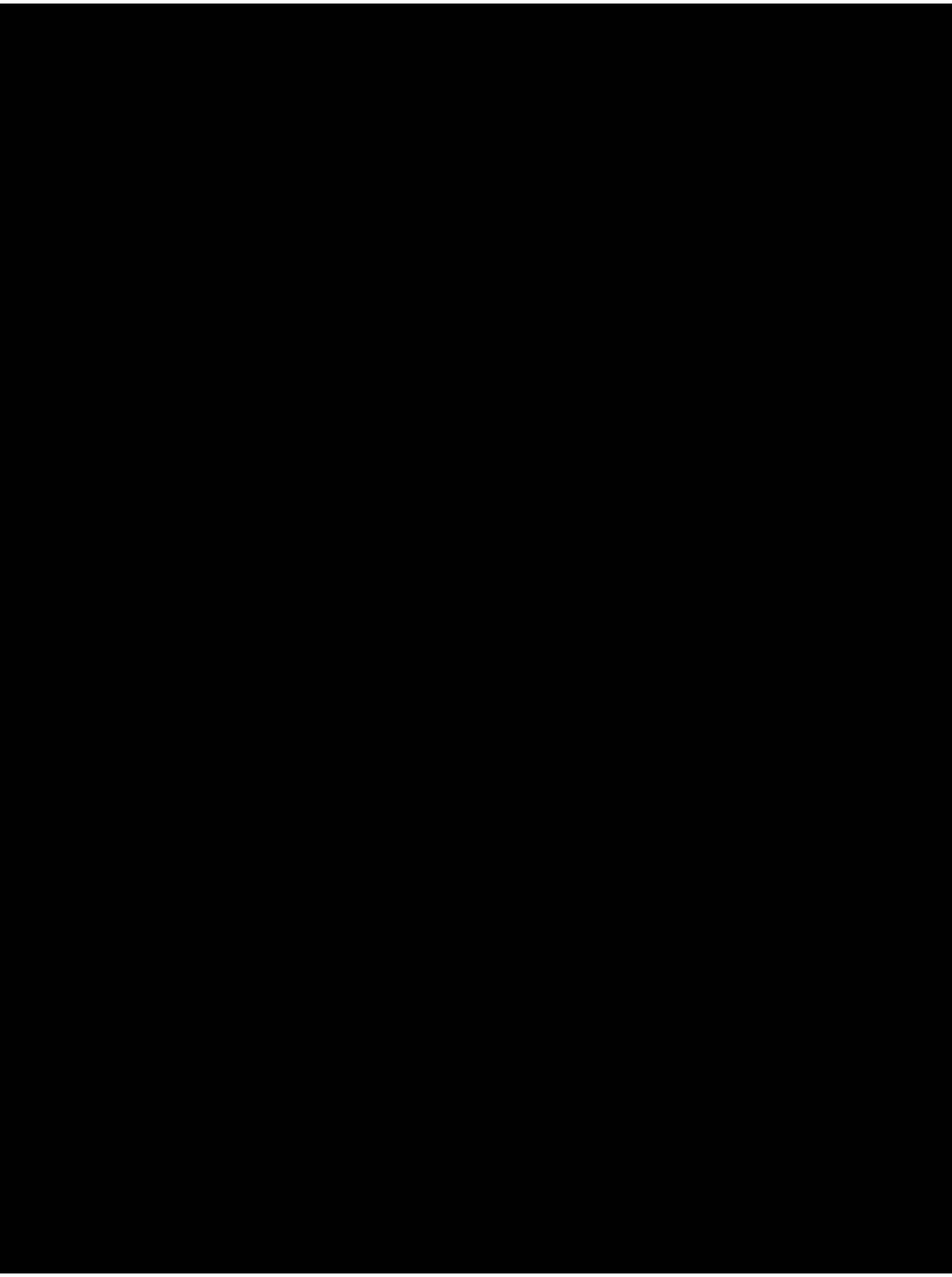
# Animation II

## DGM 2060

# Flame Animation

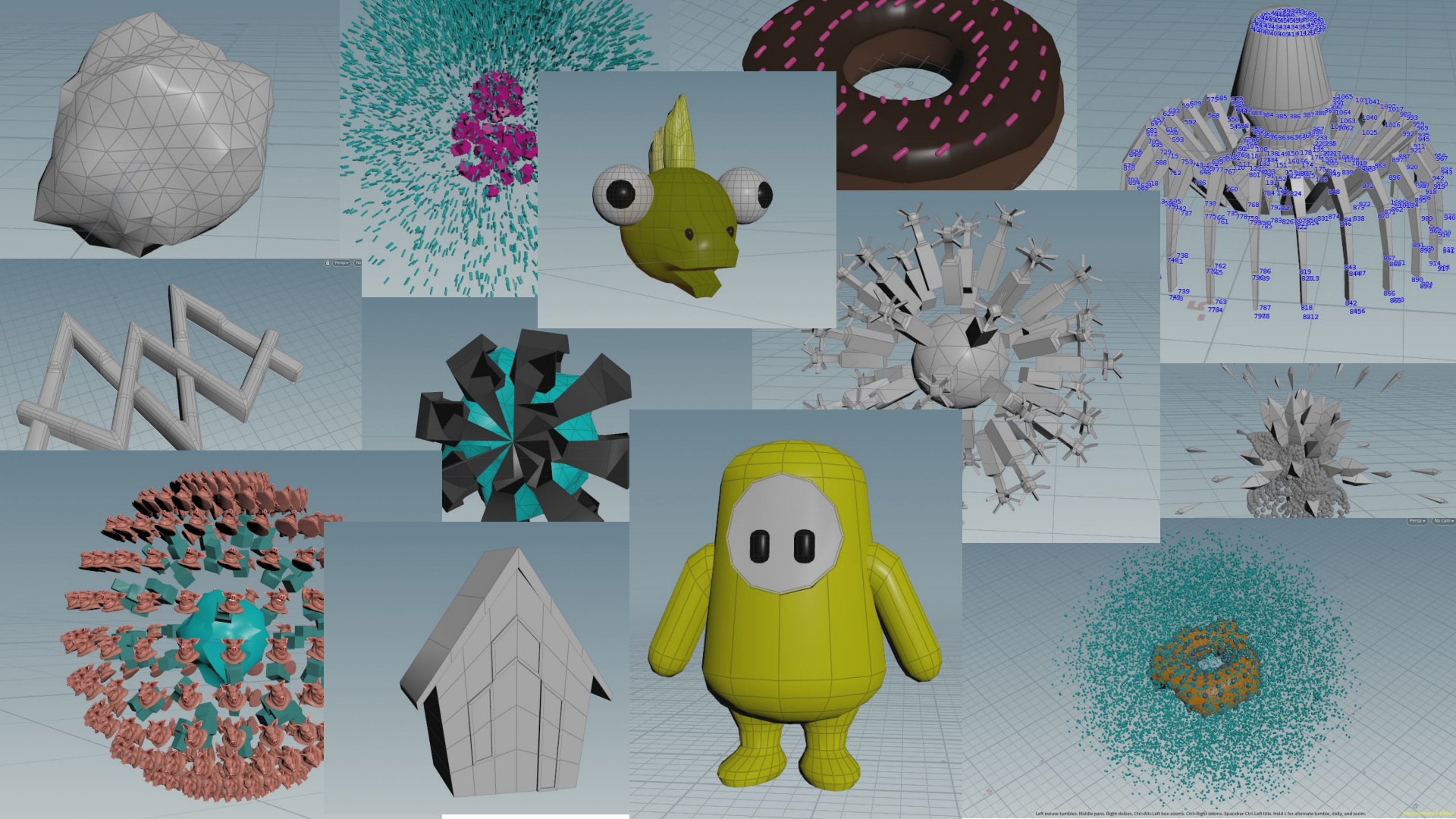


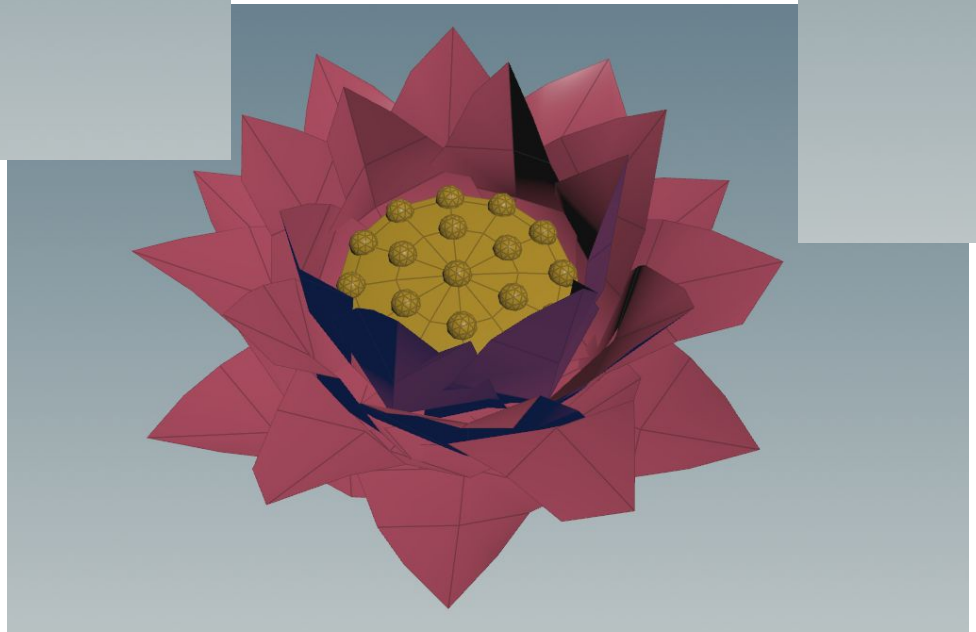
# Weightlifter Animation

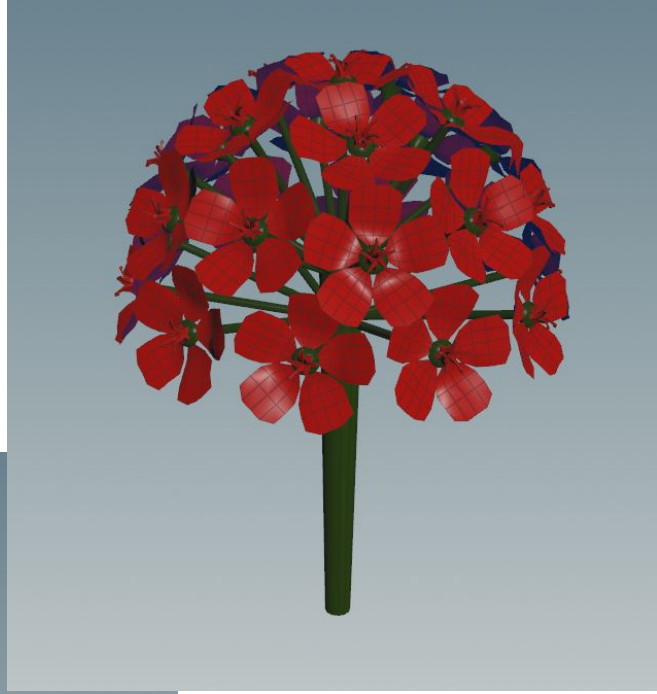


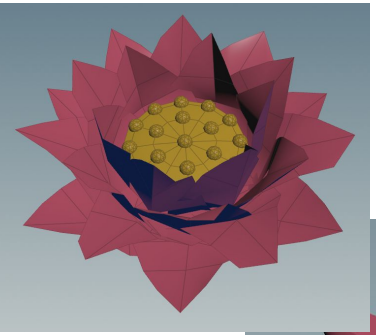
# 3D Modeling

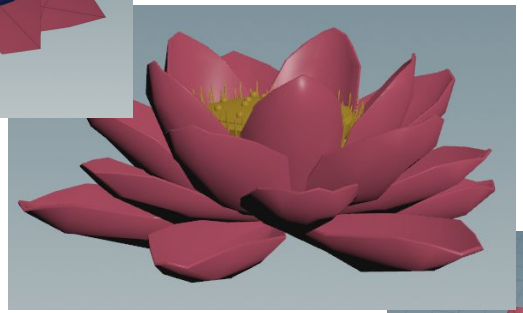
## DGM 2210

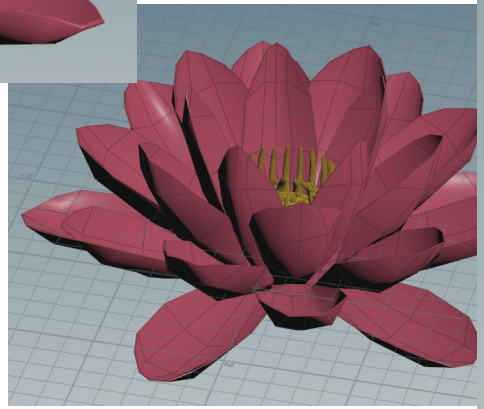












# Scripting II

## DGM 2670

```
public class ChangeWeapon : MonoBehaviour
{
    public string compareTag;  "Player"
    public UnityEvent changeWeaponEvent;  4 methods
    public FloatData originalWeaponDmg;  WeaponDamage.asset
    public float newWeaponDmg;  Changed in 1 asset
    public GameObject weapon;  Changed in 1+ assets
    public Material startColor;  Serializable

    Event function  Bethany Ylst
    public void Start()
    {
        weapon.GetComponent<MeshRenderer>().material = startColor;
        originalWeaponDmg.value = .05f;
    }

    Event function  Bethany Ylst
    private void OnTriggerEnter(Collider other)
    {
        if (other.CompareTag(compareTag))
        {
            originalWeaponDmg.value = newWeaponDmg;
            changeWeaponEvent.Invoke();
        }
    }
}
```

```
public class SceneManager : MonoBehaviour
{
    public string gameOver, playerWon, notEnough;  "NeedMore"

    public FloatData playerHealth, bossHealth;  Dr.Bentley.asset
    public IntData birdCount;  BirdCount.asset

    Event function  Bethany Ylst
    public void Update()
    {
        if (playerHealth.value <= 0 && (birdCount.value <= 0) && (bossHealth.value > 0))
        {
            UnityEngine.SceneManagement.SceneManager.LoadScene(gameOver);
        }

        if (bossHealth.value <= 0 && (birdCount.value >= 3) && (playerHealth.value > 0))
        {
            UnityEngine.SceneManagement.SceneManager.LoadScene(playerWon);
        }

        if (bossHealth.value <= 0 && (birdCount.value < 3) && (playerHealth.value > 0))
        {
            UnityEngine.SceneManagement.SceneManager.LoadScene(notEnough);
        }
    }
}
```

```
public class WeaponDamage : MonoBehaviour
{
    public FloatData enemyHealth, weaponDamage;  WeaponDamage.asset
    public string compareTag;  "Weapon"

    Event function  Bethany Ylst
    private void Update()
    {
        if (enemyHealth.value <= 0)
        {
            Destroy(gameObject);
        }
    }

    Event function  Bethany Ylst
    public void OnTriggerEnter(Collider other)
    {
        if (other.CompareTag(compareTag))
        {
            var newHealth :float  = enemyHealth.value - weaponDamage.value;
            enemyHealth.value = newHealth;
```

```
public class PlayerDeath : MonoBehaviour
{
    public FloatData playerHealth;  PlayerHealth.asset
    public WaitForSeconds wfs = new WaitForSeconds(2f);
    private MeshRenderer meshRenderer;
    public UnityEvent playerEvent;  2 methods
    public Vector3Data playerRespawnPosition;  CheckpointLocation.asset
    public GameObject playerObj;  Changed in 2 assets
    public Image img;  Health (MonoBehaviour)
    public IntData birdCount;  BirdCount.asset


    Event function  Bethany Ylst
    public void Start()
    {
        playerHealth.value = 1f;
    }

    Event function  Bethany Ylst
    public void Update()
    {
        if (playerHealth.value <= 0 && (birdCount.value > 0))
        {
            RespawnPlayer();
            img.fillAmount = playerHealth.value;
        }
    }

    Frequently called  1 usage  Bethany Ylst
    private void RespawnPlayer()
    {
        playerObj.transform.position = playerRespawnPosition.value;
        playerHealth.value = .75f;
        birdCount.value--;
        playerEvent.Invoke();
    }
```

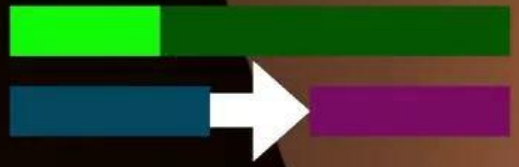
```
public class PlayerController : MonoBehaviour
{
    public FloatData flightAmount;  Changed in 2 assets
    public FloatData flyCount, gravity;  Gravity.asset
    public float maxGlide;  Unchanged
    public IntData walkSpeed;  MoveSpeed.asset

    public Image flyBar;  Fly (MonoBehaviour)
    public Image glideBar;  Glide (MonoBehaviour)

    private float flyHeight = 10f, glideSpeed = -1f;
    private float vInput, hInput, yVar, moveSpeed;
    private CharacterController controller;
    private Vector3 movement;
    private WaitForSeconds wfs = new WaitForSeconds(2f);
    private bool canGlide;

    Event function  Bethany Ylst +1
    private void Start()
    {
        moveSpeed = walkSpeed.value;
        controller = GetComponent<CharacterController>();
        maxGlide = 0;
        flightAmount.value = 40;
        gravity.value = -9.81f;
        flyBar.fillAmount = 0f;
    }
```

```
if (Input.GetKeyUp(KeyCode.LeftShift) && !controller.isGrounded)
{
    yVar = gravity.value;
    print( message: "not gliding");
}

if (controller.isGrounded)
{
    maxGlide = 0f;
}

hInput = Input.GetAxis("Horizontal") * -moveSpeed;
vInput = Input.GetAxis("Vertical") * moveSpeed;
movement.Set( newX: vInput,yVar, newZ: hInput);

Vector3 newPosition = new Vector3( x: -hInput,  y: 0.0f,  z: vInput);
transform.rotation = Quaternion.LookRotation(newPosition);

yVar += gravity.value*Time.deltaTime;

if (controller.isGrounded && movement.y < 0)
{
    flyCount.value = 0;
}

if (Input.GetButton("Jump") && flyCount.value < flightAmount.value)
{
    yVar = flyHeight;
    flyCount.value++;
}

controller.Move( motion: (movement) * Time.deltaTime);
```

```
private void Update()
{
    flyBar.fillAmount = flyCount.value;
    glideBar.fillAmount = maxGlide;

    if (flyCount.value == flightAmount.value)
    {
        flyBar.fillAmount = 0;
    }

    if (maxGlide == 100f)
    {
        glideBar.fillAmount = 0;
    }

    if (Input.GetKey(KeyCode.LeftShift) && !controller.isGrounded && maxGlide < 100f)
    {
        yVar = glideSpeed;
        maxGlide++;
        moveSpeed = walkSpeed.value;
        print( message: "gliding");

        if (maxGlide >= 100f)
        {
            yVar = gravity.value;
            print( message: "falling");
        }
    }
```

# Game Essentials

## DGM 2221



UVU Department of DIGITAL MEDIA
UTAH VALLEY UNIVERSITY

Baby Birds: 1 / 10