* Exercise:
  + Create a new scene
    - Add in a **Plane**, and a **Cylinder**
  + Exercise:
    - Derive the vector plane equation
    - Set the **Cylinder** to point in the normal direction of the plane
    - Question: how do you know your equation is correct?
    - Answer
      * Vecctor plane equation: P dot Vn = D
        + **Vn = Plane.transform.up**
        + **D = Vector3.Dot(Plane.transform.localPosition, Vn)**
      * Plane normal vector:
        + **Cylinder.transform.up = Vn**
      * Try rotating the plane see the cylinder moving with it
  + Exercise
    - Define two positions, P and Pon, project P onto the plane as Pon
    - Math
      * **P dot Vn = d**
      * **Pon = P – (d-D) \* Vn**
    - Again rotate the plane to see Pon sticks
  + Exercise
    - Define a flattened sphere (5, 0.5, 5) on the plane
    - How do you keep the sphere always flattened on the plane?
    - Answer:
      * **Sphere.transform.localPosition = Plane.transform.localPosition + 0.5 \* Vn;**
      * **Sphere.transform.up = Vn**
    - Break the answer
      * Rotate the Plane
        + along the Y-axis
        + then along any other axis!
      * Updated solution:
        + **Sphere.transform.localRotation = Plane.transform.localRotation**

**Solution**

// Update is called once per frame

void Update()

{

// Vector Plane Equation

Vn = Plane.transform.up;

D = Vector3.Dot(Plane.transform.localPosition, Vn);

// Normal vector orientation

Normal.transform.up = Vn;

// Better solution would be:

// Normal.transform.localRotation = Plane.transform.localRotation;

// Projecting the point P onto the Plane and show as Pon

float d = Vector3.Dot(P.transform.localPosition, Vn);

Pon.transform.localPosition = P.transform.localPosition - (d - D) \* Vn;

// Sticking the flattened sphere on the plane

Sphere.transform.localPosition = Plane.transform.localPosition + 0.5f \* Vn;

// Sphere.transform.up = Vn; Bad answer

Sphere.transform.localRotation = Plane.transform.localRotation;

}