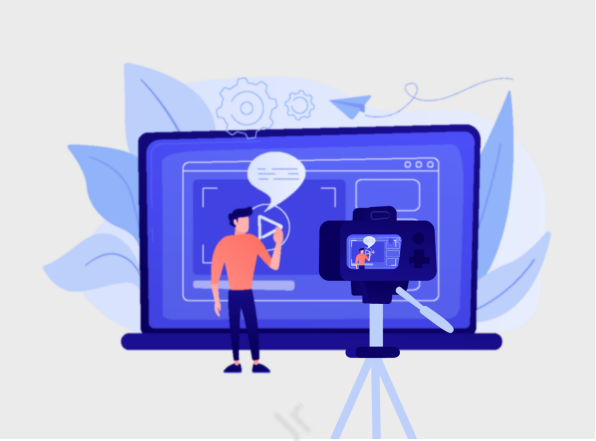
PRO-C161

CAMERA DIRECTION



What is our GOAL for this MODULE?

We learned about the relationship between A-Frame and Three.js. We learned how to access Three.js objects and methods using A-Frame entities to get the camera direction in A-Frame.

What did we ACHIEVE in the class TODAY?

- Learned how to access Three.js objects using A-Frame entities.
- Learned to use Three.js to get the camera direction in A-Frame.

Which CONCEPTS/CODING BLOCKS did we cover today?

- document.querySelector('#camera').object3D
- THREE.Vector3(), .getWorldDirection(vectorVariable)
- .addEventListener(), setAttribute(), getAttribute(), .registerComponent() methods

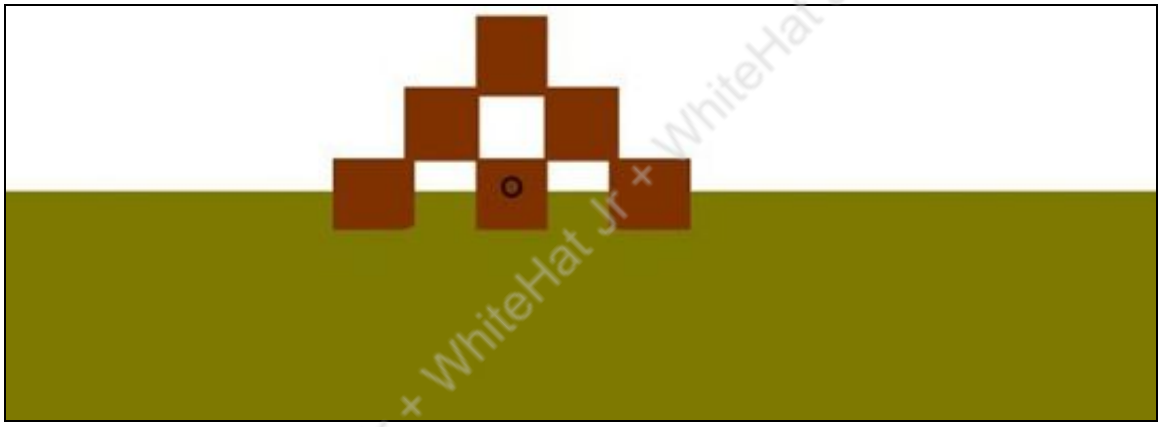
How did we DO the activities?

1. Create a basic 3D scene.

    - Add a light and camera element to the scene.
    - Attach the cursor element to the camera.
    - Add a plane for the ground; and
    - A few boxes to shoot at.

```
<!--Light-->
<a-entity light="type: ambient; color: yellow; intensity:0.5"></a-entity>

<!--Camera and Cursor-->
<a-entity
  id="camera"
  camera
  position="0 1.6 0"
  wasd-controls
  look-controls="pointerLockEnabled: true"
>
  <a-cursor></a-cursor>

</a-entity>

<!--Ground-->
<a-plane
  id="ground"
  repeat="5 5"
  position="0 1 0"
  rotation="-90 0 0"
  height="200"
  width="200"
  color="#FBF2D4"
  static-body
  visible="true"
>
</a-plane>
```

```
<!--Boxes-->
<a-box position="-2 1.5 -10" color="tomato" depth="1" height="1" width="1"></a-box>

<a-box position="0 1.5 -10" color="tomato" depth="1" height="1" width="1"></a-box>

<a-box position="2 1.5 -10" color="tomato" depth="1" height="1" width="1"></a-box>

<a-box position="-1 2.5 -10" color="tomato" depth="1" height="1" width="1"></a-box>

<a-box position="1 2.5 -10" color="tomato" depth="1" height="1" width="1"></a-box>

<a-box position="0 3.5 -10" color="tomato" depth="1" height="1" width="1"></a-box>
</a-scene>
```



2. Create a component, called “bullets”, which will create a bullet every time the “z” key is pressed.

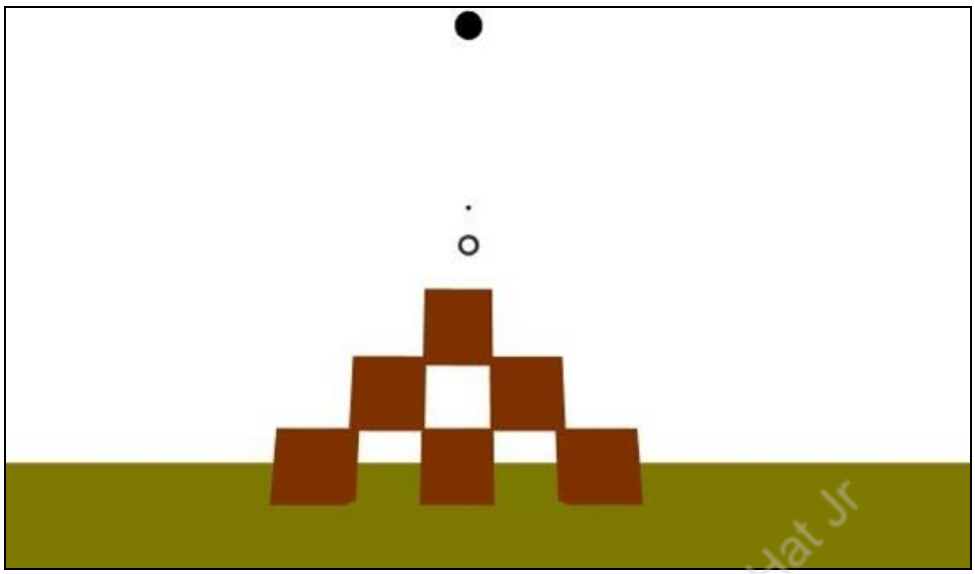
```
AFRAME.registerComponent("bullets", {

  init: function () {
    this.shootBullet();
  },
  shootBullet: function () {
    window.addEventListener("keydown", (e) => {
      if (e.key === "z") {

        var bullet = document.createElement("a-entity");

        bullet.setAttribute("geometry", {
          primitive: "sphere",
          radius: 0.1,
        });

        bullet.setAttribute("material", "color", "black");


      }
    });
  },
});
```

3. Create a function to shoot the bullet from the position of the camera.

```
shootBullet: function () {
  window.addEventListener("keydown", (e) => {
    if (e.key === "z") {
      var bullet = document.createElement("a-entity");

      bullet.setAttribute("geometry", {
        primitive: "sphere",
        radius: 0.1,
      });

      bullet.setAttribute("material", "color", "black");

      var cam = document.querySelector("#camera");

      pos = cam.getAttribute("position");

      bullet.setAttribute("position", {
        x: pos.x,
        y: pos.y,
        z: pos.z,
      });

      cam.appendChild(bullet);
    }
  });
}
```

4. Give velocity to the bullet in the negative Z-direction to move the bullet into the screen.

```
bullet.setAttribute("position", {
  x: pos.x,
  y: pos.y,
  z: pos.z,
});

bullet.setAttribute("velocity", { x: 0, y: 0, z: -1 });

cam.appendChild(bullet);
```



5. Bullet follows the camera even after the shoot, to avoid that, instead of making a bullet as the child of the camera, we should make it a child of the scene.

```
bullet.setAttribute("velocity", {x:0, y:0, z:-1});


var scene = document.querySelector("#scene");

scene.appendChild(bullet);
```

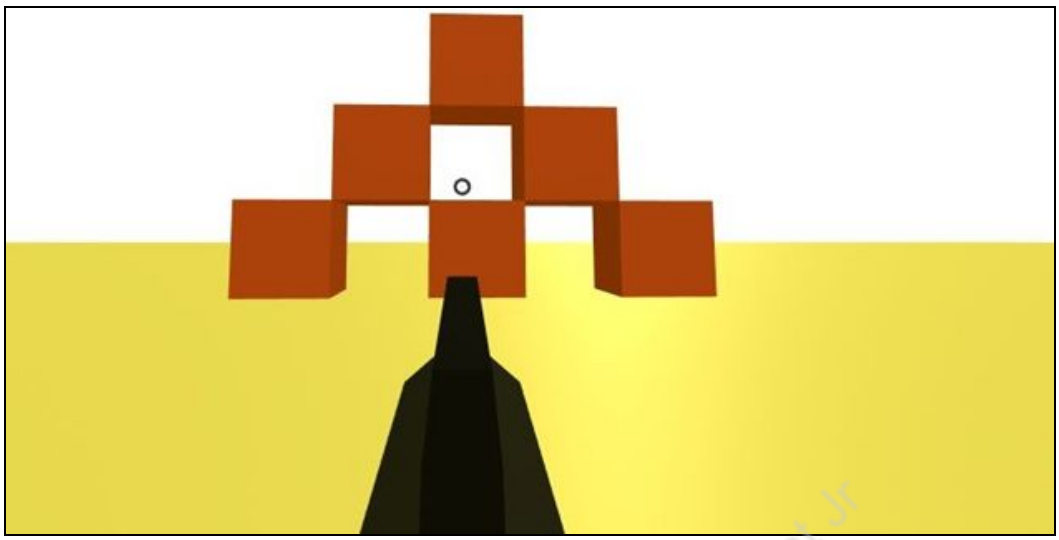
6. Get the camera's direction and use it as the velocity direction for the bullet by accessing the Three.js scene.

```
var camera = document.querySelector("#camera").object3D;

//get the camera direction as Three.js Vector
var direction = new THREE.Vector3();
camera.getWorldDirection(direction);

//set the velocity and it's direction
bullet.setAttribute("velocity", direction.multiplyScalar(-10));

var scene = document.querySelector("#scene");

scene.appendChild(bullet);
```

7. Add the gLTF model of the gun shooter in the scene as the child entity of the camera.

```
<!--Assets-->
<a-assets>
  <a-asset-item
    id="shooter"
    src="./models/shooter/gun.gltf"
  ></a-asset-item>

</a-assets>
```

```
<!--Camera and Cursor-->
<a-entity
  id="camera"
  camera
  position="0 1.6 0"
  wasd-controls
  look-controls="pointerLockEnabled: false"
>
  <a-entity
      id="weapon"
      gltf-model="#shooter"
      position="0 -4.4 3"
      rotation="0 180 0"
      scale="0.35 1 1"
    >
  </a-entity>
  <a-cursor></a-cursor>
</a-entity>
```



We have successfully learned to follow the camera direction in A-Frame using Three.js objects.

What's NEXT?

In the next class, we will be learning how to remove the elements from the A-Frame scene.

EXTEND YOUR KNOWLEDGE:

- Explore more about A-Frame.
- Explore more about Three.js Camera.