```
using System.Collections;
using System.Collections.Generic;
using UnityEngine;

public class PlayerControl : MonoBehaviour
{
   Vector3 movement;

   CharacterController controller;

   private float gravity;
   private float jumpSpeed;
   private float speed;

   public bool collided;

   private bool canDoubleJump;

   void Start()
   {
      this.controller = GetComponent<CharacterController>();
      this.jumpSpeed = 7.5f;
      this.speed = 4f;
      this.gravity = -9.8f;

      this.movement = new Vector3(0, 0, 0);

      this.canDoubleJump = false;

      collided = false;
   }

   void Update()
   {
      if (canDoubleJump && Input.GetKeyDown(KeyCode.Space))
      {
         movement.y += jumpSpeed / 2;
         Mathf.Clamp(movement.y, -20, 10);
         canDoubleJump = false;
       }

      if (Input.GetKey(KeyCode.RightArrow))
      {
         movement.x = Vector3.right.x * speed;
```

```
    }

    if (Input.GetKey(KeyCode.LeftArrow))
    {
        movement.x = Vector3.right.x * -speed;
    }

    //DON'T APPLY GRAVITY WHEN IT IS GROUNDED
    if (controller.isGrounded)
    {
      movement.y = 0f;
      canDoubleJump = false;
      if (Input.GetKey(KeyCode.Space))
      {
          movement.y += jumpSpeed;
          Mathf.Clamp(movement.y, -20, 10);
          canDoubleJump = true;

          //transform.Translate(Vector3.up * (Time.deltaTime * jumpSpeed));
      }
    }
    else
    {
        movement.y  = movement.y + (gravity * Time.deltaTime);

        if (Input.GetKey(KeyCode.DownArrow))
        {
            movement.y -= jumpSpeed / HelperClass.FAST_FALL_MAGNITUDE;
        }
    }

    transform.up = Vector3.up;
    transform.forward = Vector3.forward;
    transform.right = Vector3.right;
    transform.position = new Vector3(transform.position.x, transform.position.y, 0);


   controller.Move(movement * Time.deltaTime);
}

void OnCollisionEnter(Collision collision)
{
    if (collision.gameObject.name.StartsWith("Obstacle")) {
        movement.x = 0f;
```

```
            collided = false;
        }
    }
}
```