2D Top Down Movement Tutorial

Coding

* ***Setting up the Game:***
* Open up a new project in Unity Hub by selecting New, **2D** in Unity Hub (make sure you select **2D** instead of 3D).
* Once you are inside of Unity, go to the Hierarchy (the menu to the left hand side of the screen, and above the Project menu) and right click. Select 3D Object, Sphere. This will create a sphere in your Scene (the centre screen where you can edit your scene). Rename the sphere “Player” by clicking on the sphere in the Hierarchy to rename it.
* In the Inspector (the menu to the right hand side of the screen), and while selecting the Player in your Hierarchy, scroll down and delete the Sphere Collider by clicking on the cog wheel (on the far right hand side of the Sphere Collider tag) and select Remove Component. This is because the collider is 3D, not 2D. Also, in order to centre your player, find the Transform menu in the Inspector (it should be the first menu down), right click on the Transform text and select Reset. This will default the player to the centre of your screen.
* *While we’re here, it’s always nice to stay organised and to know where everything is so everything you create doesn’t get cluttered. We need to create come folders to stay organised. To do this, head to Project (the menu below the Scene screen) and right click, select Create, and Folder. Click on the folder you have created to rename it. I would recommend always creating a folder for Scripts and Materials.*
* We should now create a Material so we can make our Player stand out more. To do this, right click in the Project menu and select Create, Material. Rename the material “Player” and move it into your Material folder. To change the colour of your material, double click on your Player material, and in the Inspector, find Albedo (under Main Maps), and click on the white colour box. Select whatever colour you like (apart from blue as the game background is already blue). *I would choose red as a nice base colour.*
* To attach this material to your Player, go back to the Hierarchy, select your Player, and drag your player Material to a box that says “Element 0” (under Mesh Renderer, Materials). Your player should now be whatever colour you selected in the scene view.
* ***Creating the Movement:***
* Time to start coding! Inside the Inspector menu (while selecting your Player in the Hierarchy), click Add Component at the bottom of the menu and type “PlayerController.” Click new script, create and add, and you should see your new script attached to your player automatically (in the Project menu, click Assets and drag your new script into your Scripts folder, just so the menu doesn’t get cluttered). Double click the script to open it up in Visual Studio.
* Once you are in Visual Studio, delete lines 7 to 17 so you are left with this:
* public class PlayerController : MonoBehaviour  
  {  
    
  }
* Once you have done this, in between the {}’s, type public float speed; to give the player a speed variable that will determine how fast the player will move.
* Below this, type private Rigidbody2D rb; (rb is short for Rigid Body).
* Below this, type void Start (), add { and press Enter. Your code should now look something like this:
* public class PlayerController : MonoBehaviour  
  {  
   public float speed;  
    
   private Rigidbody2D rb;  
    
   void Start()  
   {  
    
   }  
    
  }
* In between the two {}’s under void Start, type rb = GetComponent<Rigidbody2D>(); (this is saying that rb is equal to the Rigid Body component that we will add to our player).
* Once you have done this, save your code and head back into Unity and under the Inspector menu while selecting the Player in the Hierarchy, select Add Component and type Rigid Body 2D (it must be **2D** as a regular Rigid Body is 3D, and we don’t want that for the purpose of this tutorial). Under the Rigid Body 2D menu, select Body Type (it should be the first option down) and change it from Dynamic to Kinematic. This means that the player won’t be affected by gravity or other external forces.
* Head back into Visual Studio and add an Update function, which is basically what you did to create the Start function above. Place this function below the Start function so your code should look like this:
* public class PlayerController : MonoBehaviour  
  {  
   public float speed;  
    
   private Rigidbody2D rb;  
    
   void Start()  
   {  
   rb = GetComponent<Rigidbody2D>();  
   }  
    
   void Update()  
   {  
    
   }  
    
  }
* Inside the Update function (aka between { and }), type Vector2 moveInput = new Vector2(Input.GetAxis(“Horizontal”), Input.GetAxis(“Vertical”)); this basically detects where the player wants to move.
* Above the Start function and below private Rigidbody2D rb;, type private Vector2 moveVelocity; and then inside the Update function and below the line of code you wrote in the previous step, type moveVelocity = moveInput.normalised \* speed;
* Below the entire Update function, create a FixedUpdate function, exactly like how you created your Start and Update functions from before. (Note that all of the code that will adjust physics must go inside here). Your code should now look like this:
* public class PlayerController : MonoBehaviour  
  {  
   public float speed;  
    
   private Rigidbody2D rb;  
   private Vector2 moveVelocity;  
    
   void Start()  
   {  
   rb = GetComponent<Rigidbody2D>();  
   }  
    
   void Update()  
   {  
   Vector2 moveInput = new Vector2(Input.GetAxis("Horizontal"), Input.GetAxis("Vertical"));  
   moveVelocity = moveInput.normalized \* speed;  
   }  
    
   void FixedUpdate()  
   {  
    
   }

}

* Inside your new Fixed Update function, type rb.MovePosition(rb.position + moveVelocity \* Time.fixedDeltaTime);
* Once you have done this, save your code and head back into Unity. Click on your Player either inside the Scene or the Hierarchy and head to the Inspector. Underneath your PlayerMovement script, you should see a value for Speed. Set this speed to 10 as an example. You should now be able to move your character once you play your game! :D
* The final code should look like this:
* public class PlayerController : MonoBehaviour  
  {  
   public float speed;  
    
   private Rigidbody2D rb;  
   private Vector2 moveVelocity;  
    
   void Start()  
   {  
   rb = GetComponent<Rigidbody2D>();  
   }  
    
   void Update()  
   {  
   Vector2 moveInput = new Vector2(Input.GetAxis("Horizontal"), Input.GetAxis("Vertical"));  
   moveVelocity = moveInput.normalized \* speed;  
   }  
    
   void FixedUpdate()  
   {  
   rb.MovePosition(rb.position + moveVelocity \* Time.fixedDeltaTime);  
    
   }  
  }
* ***Extra:***
* *Depending on what you want for your game, you may find that the movement is too floaty, and the player may take too long to stop and takes too long to accelerate. If these settings are fine for you, then you can stop the tutorial here. If you want to change this so that the player reaches max speed immediately and stops immediately, it is a very simple change. In Visual Studio, head to the first line of your Update function, which should look like this:*
* *(Vector2 moveInput = new Vector2(Input.GetAxis("Horizontal"), Input.GetAxis(“Vertical”));*
* *All you need to do is change “GetAxis” to “GetAxisRaw” before both Horizontal and Vertical. It’s that simple! :D*
* *Video that this tutorial was inspired by:*
* <https://www.youtube.com/watch?v=CeXAiaQOzmY&t=19s>