GAME CONTROLLER INPUTS – FOR UNITY 3D

Here's a list of all properties accessable by any script to test for inputs from a game controller.

Active Input

// Returns TRUE when any input is active (Can be used to block lines of code from executing if there is no input from the controller)

public bool IsInputActive

Left and Right Directionals

// Returns an analog value from -1 (Left Directional completely to the LEFT) to 1 (Left Directional completely to the RIGHT), returning 0 when in the middle.

public float LeftDirectional_Horizontal

// Returns TRUE while Left Directional is moved to the LEFT (past the threshold). Returns FALSE while released.

public bool LeftDirectional_asLeftButton

// Returns TRUE while Left Directional is moved to the RIGHT (past the threshold). Returns FALSE while released.

public bool LeftDirectional_asRightButton

// Returns an analog value from -1 (Left Directional completely to the TOP) to 1 (Left Directional completely to the BOTTOM), returning 0 when in the middle.

public float LeftDirectional_Vertical

// Returns TRUE while Left Directional is moved to the TOP (past the threshold). Returns FALSE while released.

public bool LeftDirectional_asUpButton

// Returns TRUE while Left Directional is moved to the BOTTOM (past the threshold). Returns FALSE while released.

public bool LeftDirectional_asDownButton

// Returns an analog value from -1 (Right Directional completely to the LEFT) to 1 (Rightt Directional completely to the RIGHT), returning 0 when in the middle.

public float RightDirectional_Horizontal

// Returns TRUE while Right Directional is moved to the LEFT (past the threshold). Returns FALSE while released.

public bool RightDirectional_asLeftButton

// Returns TRUE while Right Directional is moved to the RIGHT (past the threshold). Returns FALSE while released.

public bool RightDirectional_asRightButton

// Returns an analog value from -1 (Right Directional completely to the TOP) to 1 (Right Directional completely to the BOTTOM), returning 0 when in the middle.

public float RightDirectional_Vertical

// Returns TRUE while Right Directional is moved to the TOP (past the threshold). Returns FALSE while released.

public bool RightDirectional_asUpButton

// Returns TRUE while Right Directional is moved to the BOTTOM (past the threshold). Returns FALSE while released.

public bool RightDirectional_asDownButton

Triggers

// Returns an analog value from -1 to 1 that is a combination from the Left Trigger (0 to 1) and the Right Trigger (0 to -1). It returns 0 if both triggers are released (0 + 0) or fully pressed (-1 + 1).

public float BothTriggers

// Returns an analog value from 0 (Trigger released) to 1 (Trigger fully pressed).

public float LeftTrigger

// Returns an analog value from 0 (Trigger released) to 1 (Trigger fully pressed).

public float RightTrigger

// Returns TRUE while the Left Trigger is being pressed (past the threshold). Returns FALSE while released.

public bool LeftTrigger_asButton

// Returns TRUE while the Right Trigger is being pressed (past the threshold). Returns FALSE while released.

public bool RightTrigger_asButton

DPADs

// Returns TRUE while the DPad's Left direction is being pressed. Returns FALSE while released.

public bool DPad_Left

// Returns TRUE while the DPad's Right direction is being pressed. Returns FALSE while released.

public bool DPad_Right

// Returns TRUE while the DPad's Up direction is being pressed. Returns FALSE while released.

public bool DPad_Up

// Returns TRUE while the DPad's Down direction is being pressed. Returns FALSE while released.

public bool DPad_Down

ABXY buttons

> *An quick explanation about **Buttons**: When the end of the variable ends in* **_hold***, it means that the input will return TRUE every frame that the button is held down. All variables that ends in* **_asButton** *or* **_as(direction)Button** *will behave the same way. Variables that ends in _down will return TRUE only on the frame that the press happened, even if the the button continues to be pressed afterwards. Variables ending in* **_up** *will only return TRUE on the frame that the user released the button. In this context* **_down** *and* **_up** *is not related to the up and down direction of a movement.*

// Returns TRUE while the A button is being pressed. Returns FALSE while released.

public bool A_button_hold

// Returns TRUE while the B button is being pressed. Returns FALSE while released.

public bool B_button_hold

// Returns TRUE while the X button is being pressed. Returns FALSE while released.

public bool X_button_hold

// Returns TRUE while the Y button is being pressed. Returns FALSE while released.

public bool Y_button_hold

// Returns TRUE during the frame the A button was pressed.

public bool A_button_down

// Returns TRUE during the frame the B button was pressed.

public bool B_button_down

// Returns TRUE during the frame the X button was pressed.

public bool X_button_down

// Returns TRUE during the frame the Y button was pressed.

public bool Y_button_down

// Returns TRUE the first frame the A button was released.

public bool A_button_up

// Returns TRUE the first frame the B button was released.

public bool B_button_up

// Returns TRUE the first frame the X button was released.

public bool X_button_up

// Returns TRUE the first framer the Y button was released.

public bool Y_button_up

Bumbers

// Returns TRUE while the Left Bumper is being pressed. Returns FALSE while released.

public bool LB_hold

// Returns TRUE during the frame the Left Bumper was pressed.

public bool LB_down

// Returns TRUE the first frame the Left Bumper was released.

public bool LB_up

// Returns TRUE while the Right Bumper is being pressed. Returns FALSE while released.

public bool RB_hold

// Returns TRUE during the frame the Left Bumper was pressed.

public bool RB_down

// Returns TRUE the first frame the Right Bumper was released.

public bool RB_up

Directional Press

// Returns TRUE while the Left Directional is being pressed down (click). Returns FALSE while released.

public bool LeftDir_press_hold

// Returns TRUE during the frame the Left Directional was pressed down (click).

public bool LeftDir_press_down

// Returns TRUE the first frame the Left Directional press (click) is released (click).

public bool LeftDir_press_up

// Returns TRUE while the Right Directional is being pressed down (click). Returns FALSE while released.

public bool RightDir_press_hold

// Returns TRUE during the frame the Right Directional was pressed down (click).

public bool RightDir_press_down

// Returns TRUE the first frame the Right Directional press (click) is released (click).

public bool RightDir_press_up

Start and Back Buttons

// Returns TRUE while the Start button is being pressed. Returns FALSE while released.

public bool Start_hold

// Returns TRUE during the frame the Start button was pressed down.

public bool Start_down

// Returns TRUE the first frame the Start button press is released.

public bool Start_up

// Returns TRUE while the Back button is being pressed. Returns FALSE while released.

public bool Back_hold

// Returns TRUE during the frame the Back button was pressed down.

public bool Back_down

// Returns TRUE the first frame the Back button press is released.

public bool Back_up