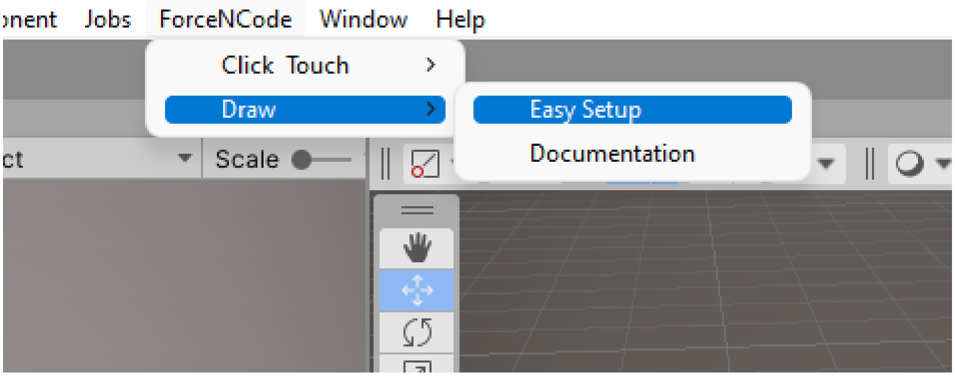
Force Draw Documentation

Initialization:

Force Draw Online Documentation Page: https://forcencode.com/ForceDrawDocumentation

Force Click Package (Simple) is also used in this package, to learn how to set it up please check out this page https://forcencode.com/ForceClickDocumentation



1- Find ForceNCode Bar on the Top Navigation Bar (File, Edit, Assets… etc.)
2- You should see a section as "Draw", if not please be sure your project has no error so that it can compile at least once after importing this asset.
3- Press Easy Setup in the "Draw" Section

   What does Easy Setup do?

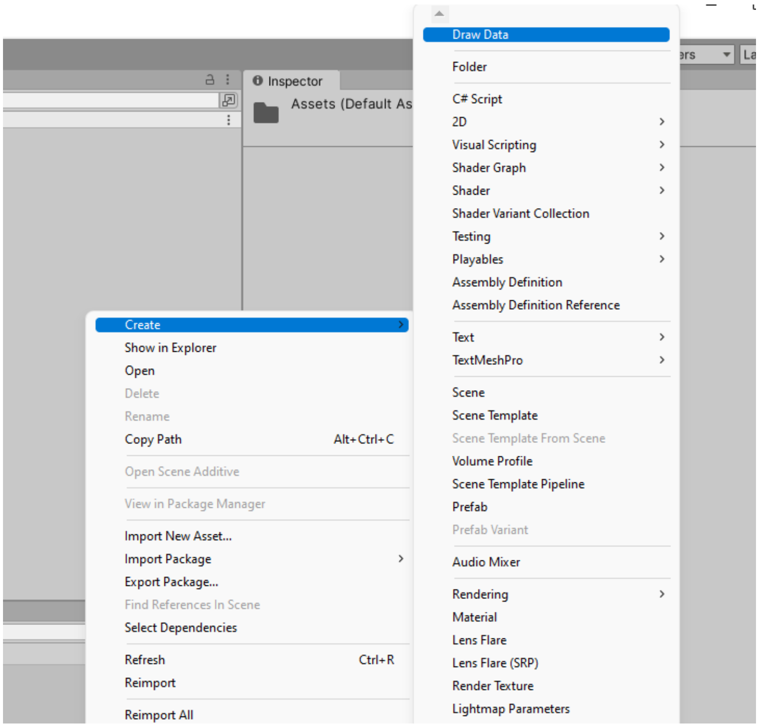
   1- Instantiates a new GameObject
   2- Adds DrawLineController Component to it.

4- Setup is completed.

Overview & Simple Usage:

First of all you need to create **Draw Data** to arrange drawing settings such line size etc.

To Create Draw Data:



After Right Clicking in any folder in Project Tab, press “Draw Data” under “Create” section.

To have detailed data about generated scriptable object “Draw Data” check out **Draw Data in Methods & Classes & Scriptable Objects**

Please notice the prefab Under Prefabs Folder (/Prefabs):

-LinePrefab

LinePrefab is used for drawing each fragment of the line.

Put **LinePrefab** in the **Draw Data** you have generated.

Find DrawLineController generated in Initialization part (or create an object and add DrawLineController Component),

and put **Draw Data** you have generated to “Draw Data” of DrawLineController Component.

Components & Scriptable Objects:

DrawLineController:

Fields:

string Id: Indicates Line Id, if this is empty IsDrawing might be wrong. If 2 DrawLineController have same Line Id, their end might connect when too close.

Transform LineParent: Indicates the transform where Lines are going to be generated. If null this is set as self transform.

DrawData drawData: Drawing and Line Information(Settings) that is required to draw line. To have this press Right Click to your Assets and press Draw Data.

Transform StartPosition: Indicates the transform where line should started if CanOnlyStartFromStartPosition is true in drawData. If null this is set as self transform.

List<Line> Lines: List of Line elements that are generated.

bool IsDrawing: This is true if Line is being drawn.

bool IsActivated: Cannot be drawn if not active.

bool IsAvailable: Cannot be drawn if not available.

Events:

OnStartedToAddLine: Fired when started to generate a line fragment

OnLineAdded: Fired after generated a line fragment.

OnActivated: Fired when IsActivated becomes true.

OnNotActivated: Fired when IsActivated becomes false.

OnAvailable: Fired when IsAvailable becomes true.

OnNotAvailable: Fired when IsAvailable becomes false.

OnDrawingSessionStarted: Fires when user started drawing a line.

OnDrawingSessionEnded: Fires when user ends drawing a line.

Methods:

void SetIsActivatedTrue(): Sets IsActivated as true

void SetIsActivatedFalse(): Sets IsActivated as false

void SetIsAvailableTrue(): Sets IsAvailable as true

void SetIsAvailableFalse(): Sets IsAvailable as false

bool GetIsAvailable(): Gets IsAvailable

bool GetIsActive(): Gets IsActivated

void FlushLines(): Clears Line list but not destroys lines

void DestroyAllLines(): Clears Line list and destroys all lines

Draw Data:

Fields:

GameObject LinePrefab: Single Line fragment generated by DrawLineController

Color LineColor: Indicates color of line generated

float LineSize: Indicates size(thickness) of line generated

int MaxLineAmount: Indicates Maximum line fragment that can be generated. If less than 3 this will be considered as unlimited.

bool ClearLinesIfAmountPassesMaxLine: If This is true, lines are cleared when amount passes MaxLineAmount, If false, instead of cleaning user cannot draw anymore.

float DrawOffset: Indicates the range of click that checks availability of draw line fragment

float MinDrawOffset: If click range is below this range, you cant draw line so that if user dont move click or touch it wont draw a line

float DistanceBetweenLines: Distance Between Each Line Fragment, even if user click somewhere away than this distance, line fragment instantiated at position with this distance from previous line fragment.

DrawEnums.DrawPlaneAxis DrawPlane: Draw line to given Draw Plane. For Example: if Y is selected, line can propagate in XZ plane only

float DrawPlaneOffsetValue: Only applicable if DrawPlane is not All (DrawEnums.DrawPlaneAxis.All). If this is selected rather than zero line is drawn with offset of this value.

DrawEnums.DrawDimensions Dimension: Changes if line is 3D or 2D

float DrawLineRefreshTimer: Indicates the duration of setting value of IsDrawing as false, after removing click or touch

bool CanOnlyStartFromStartPosition: If this is true, Start Position of line can only be position of given StartPosition Transform. If false, it can start from anywhere

bool CanHaveMoreThanOneStart: If this is false, user can only 1 line and continue it by continue drawing end of the line. If true, user can draw as much as wanted.

bool CanOnlyDrawInSingleHold: If this is true, line only can be edited in draw session, line cannot be continued

Line:

Fields:

GameObject Capsule: Capsule element of LinePrefab

GameObject Sphere: Sphere element of LinePrefab

GameObject Box: Box element of LinePrefab

GameObject Cylinder: Cylinder element of LinePrefab

List<MeshRenderer> ExtraColorMeshes: If somehow it is required to add or remove elements to LinePrefab that is required to be colored same as line color set those components to this list

int Order: Adding Order of lines

Line NextLine: Line added to DrawLine->Lines list after this line

DrawLineController DrawLine: The DrawLineController that generates this line

bool IsLast: This is true if this Line is end part of a drawn line fragment

Events:

OnUnlasted: Fired when IsLast becomes false

OnLasted: Fired when IsLast becomes true

OnColored: Fired When color changed

OnOrderSet: Fired When Order Set

OnSetNextLine: Fired when next line set

Methods:

Line GetNextLine(): Gets Line added to DrawLine->Lines list after this line.

void ActivateLine(bool StartOrEnd,DrawEnums.DrawDimensions Dimension): Activates Required Inner GameObject

void SetDrawLineController(DrawLineController controller): Sets DrawlineController as controller parameter

DrawLineController GetDrawLineController(): The DrawLineController that generates this line

void SetLastness(bool IsLast): Sets IsLast True or false

bool GetIsLast(): This is true if this Line is end part of a drawn line fragment

void SetColor(Color color): Colors this line element

void SetOrder(int Order): Sets Order of this element

int GetOrder(): Adding Order of lines