Blueprints Tips

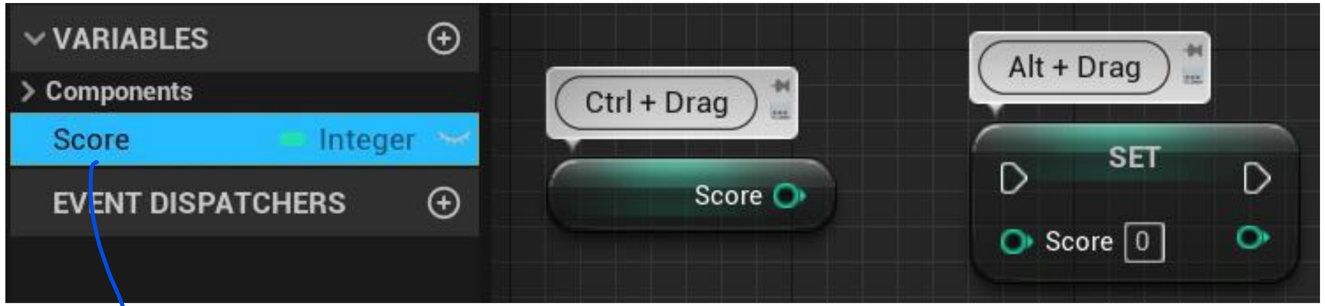
9th Week, 2022

Blueprint Editor shortcuts (1)

› Shortcuts to create **GET** and **SET** nodes

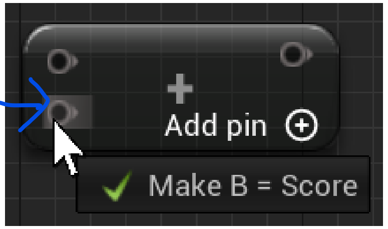
단축키

- – Ctrl + Drag: to create **GET** node
- – Alt + Drag: to create **SET** node



< Shortcuts to create GET and SET nodes >

- – Dropping a variable on an input parameter pin: to create a **GET** node

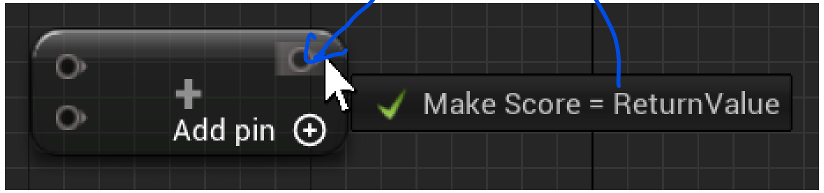


자동으로 get 생성

< Dragging a variable and dropping it on an input pin to create a GET node >

Blueprint Editor shortcuts (2)

- Dropping a variable on an output parameter pin: to create a **SET** node

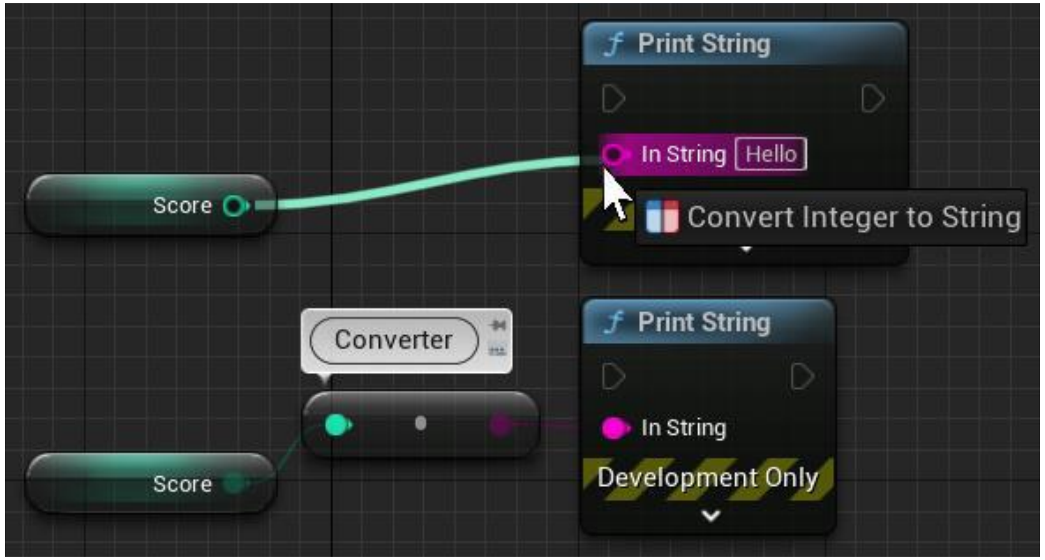


✓ Make Score = ReturnValue

자동으로 Set생성

< Dragging a variable and dropping it on an output pin to create a SET node >

› The Blueprint Editor has an automatic type conversion system.



Convert Integer to String

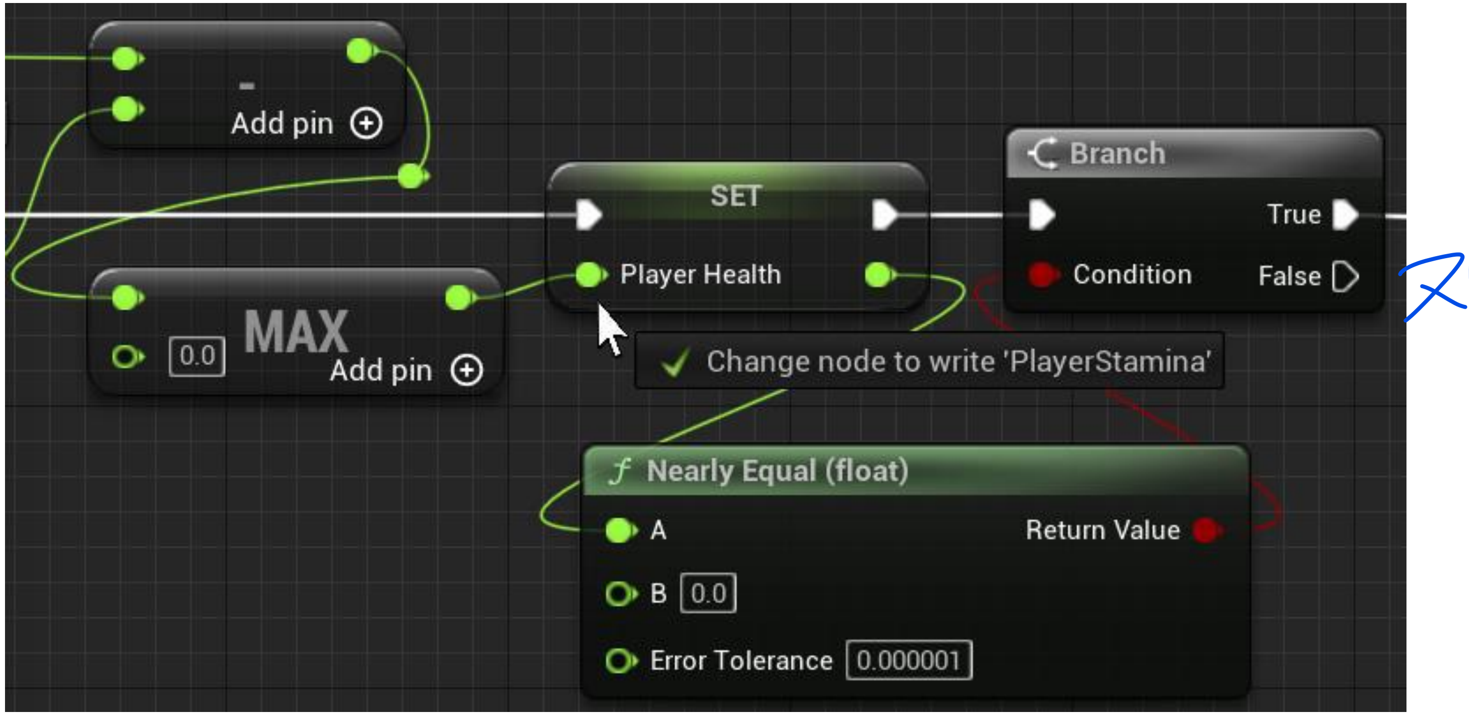
자동 변환

< Creating a converter node >

Blueprint Editor shortcuts (3)

› In the Blueprint Editor, it is possible to change an existing node for another node that uses the same variable type without breaking the connections.

같은 타입일땐 자유롭게

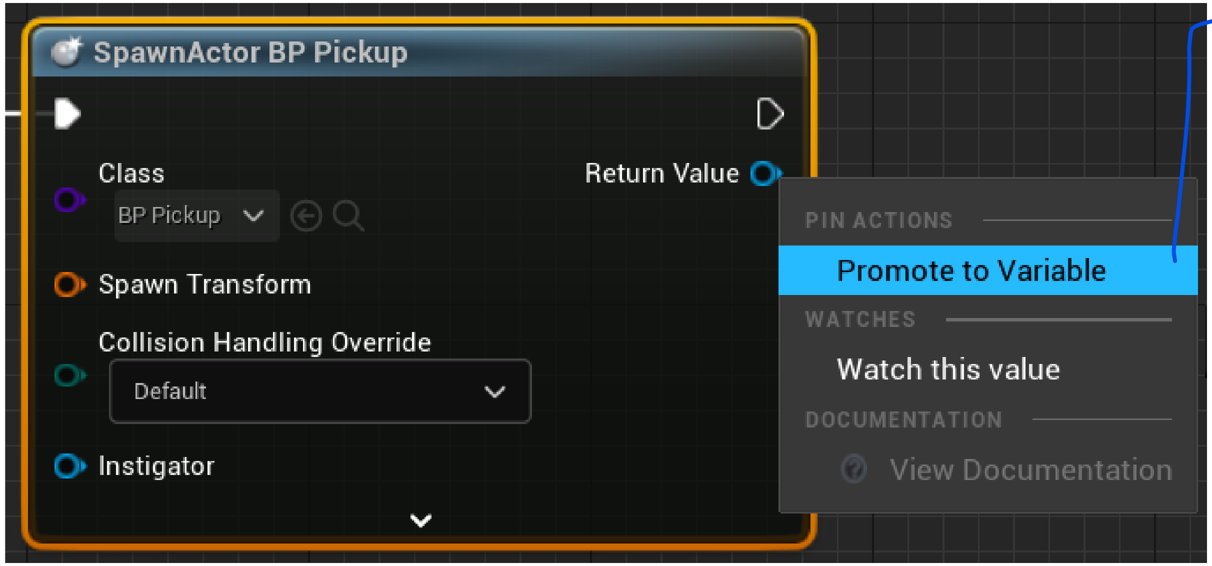


✓ Change node to write 'PlayerStamina'

< Changing a node and keeping all connections >

Blueprint Editor shortcuts (4)

- The **Promote to Variable** option: A shortcut to create variable based on the type of an input or output pin of a node.



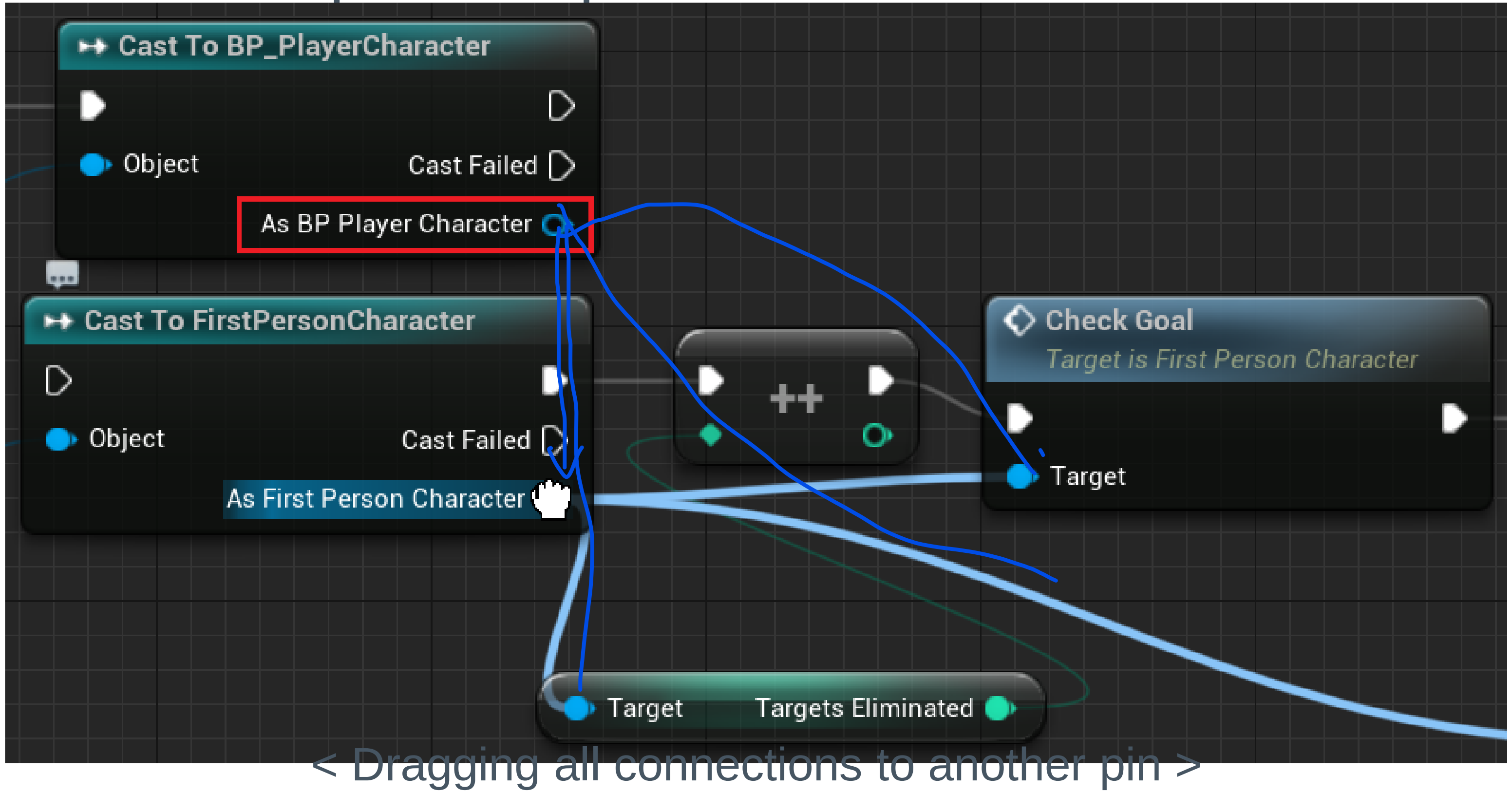
→ 자동으로 변수 생성

< Promoting the return value to a variable >

Blueprint Editor shortcuts (5)

- Alt + Click: to break all the connections of a pin
- Ctrl + Drag: to move all the connections of a pin to another compatible pin

모든 연결을 이동시킴

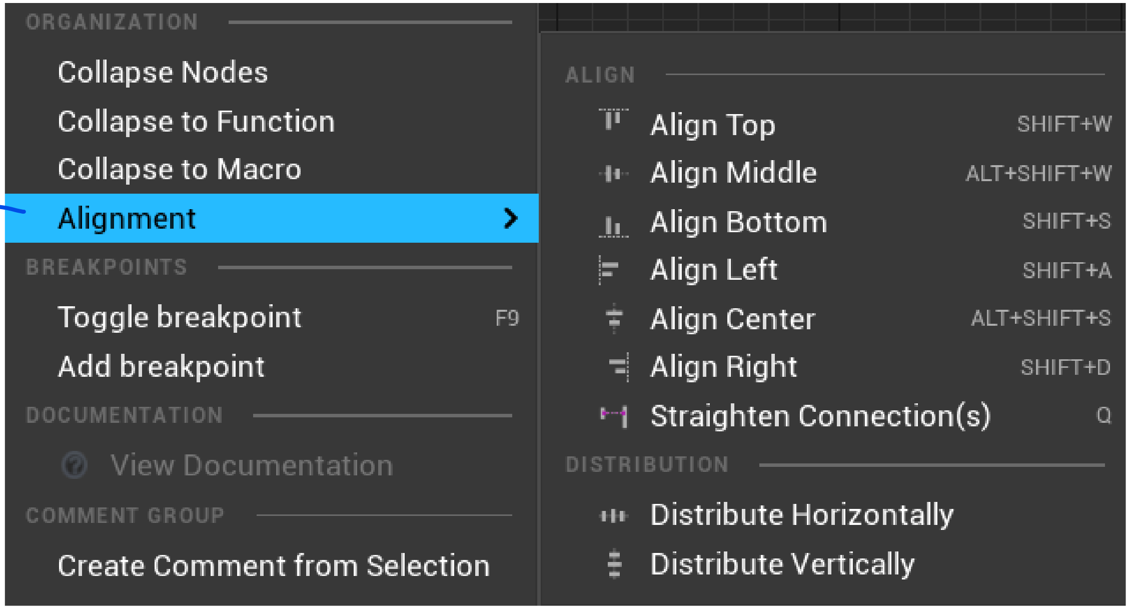

Cast To BP_PlayerCharacter
Object
Cast Failed
As BP Player Character
Cast To FirstPersonCharacter
Object
Cast Failed
As First Person Character
++
Check Goal
Target is First Person Character
Target
Target
Targets Eliminated

< Dragging all connections to another pin >

Blueprint Editor shortcuts (6)

- The Blueprint Editor offers several options for node alignment.

노드 정렬



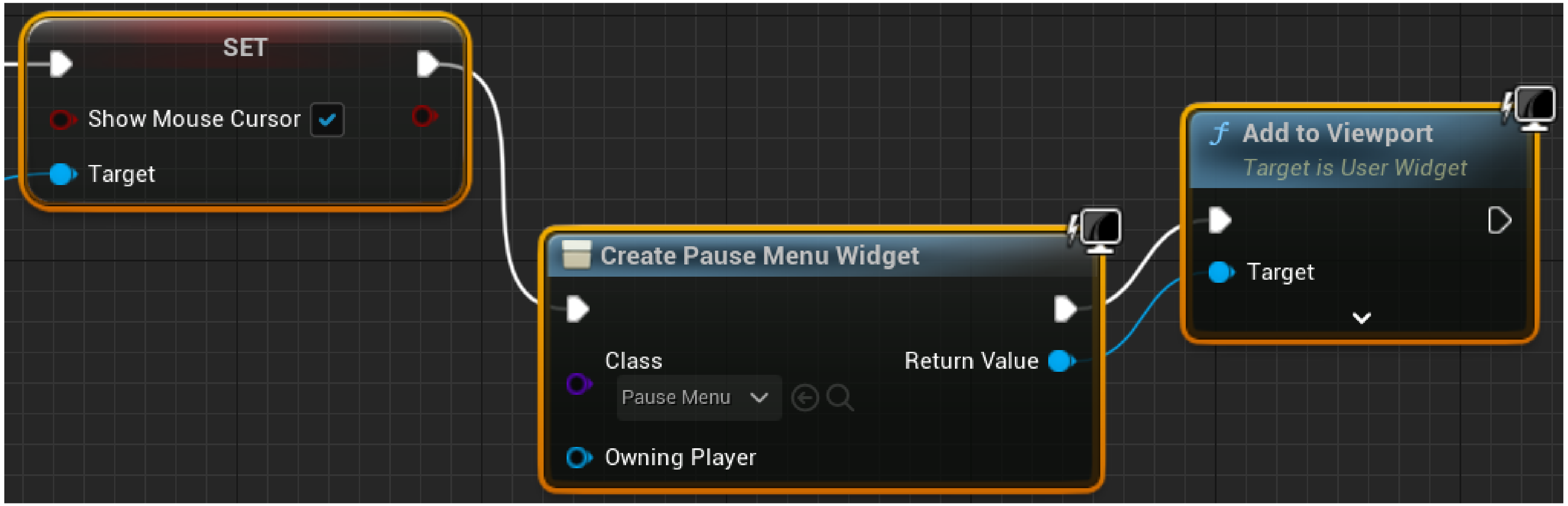
ORGANIZATION
Collapse Nodes
Collapse to Function
Collapse to Macro
Alignment
BREAKPOINTS
Toggle breakpoint F9
Add breakpoint
DOCUMENTATION
View Documentation
COMMENT GROUP
Create Comment from Selection

ALIGN

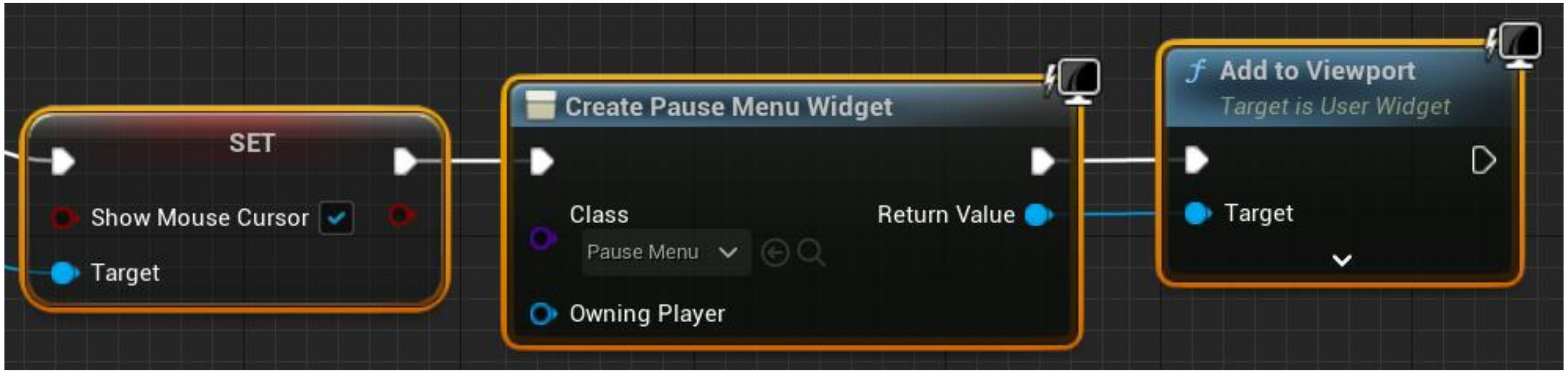
Align Top	SHIFT+W
Align Middle	ALT+SHIFT+W
Align Bottom	SHIFT+S
Align Left	SHIFT+A
Align Center	ALT+SHIFT+S
Align Right	SHIFT+D
Straighten Connection(s)	Q

DISTRIBUTION
Distribute Horizontally
Distribute Vertically

< The Alignment options >



< These nodes will be aligned >



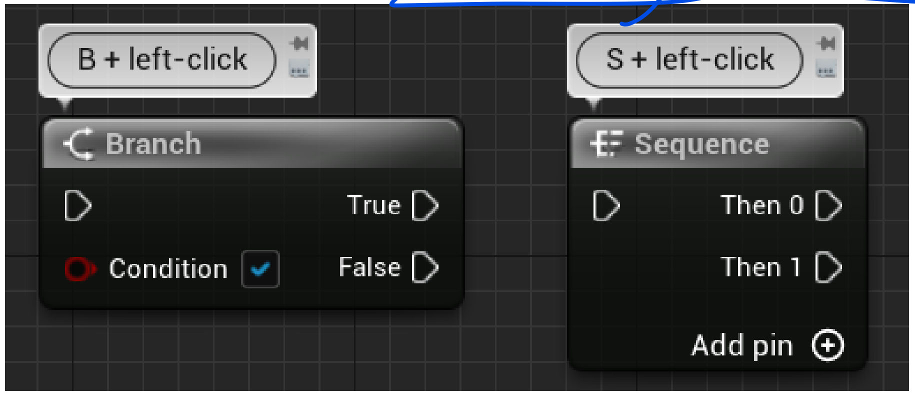
< The nodes after applying Straighten Connection(s) >

Blueprint Editor shortcuts (7)

- Shortcut keys to create some common nodes in Blueprints
  - B + left-click: to create a **Branch** node
  - S + left-click: to create a **Sequence** node

b + 클릭

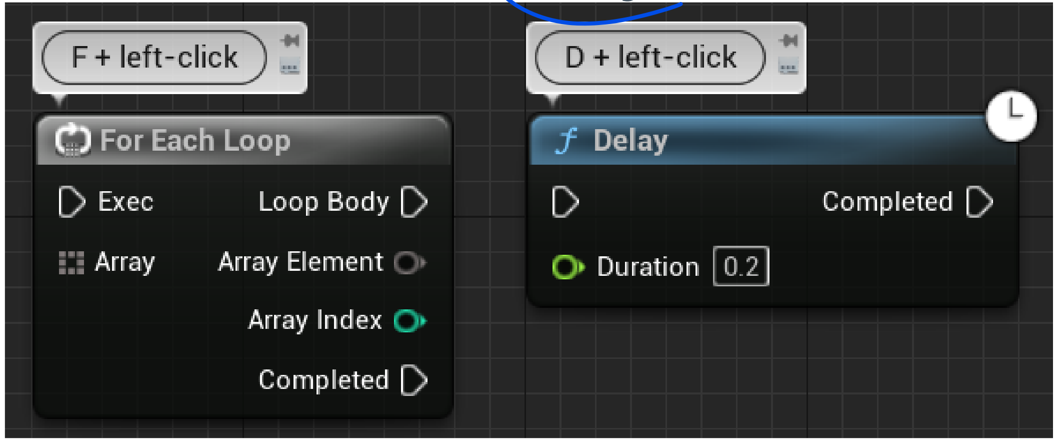
S + 클릭



< Shortcuts for Branch and Sequence nodes >

Blueprint Editor shortcuts (8) F 클릭

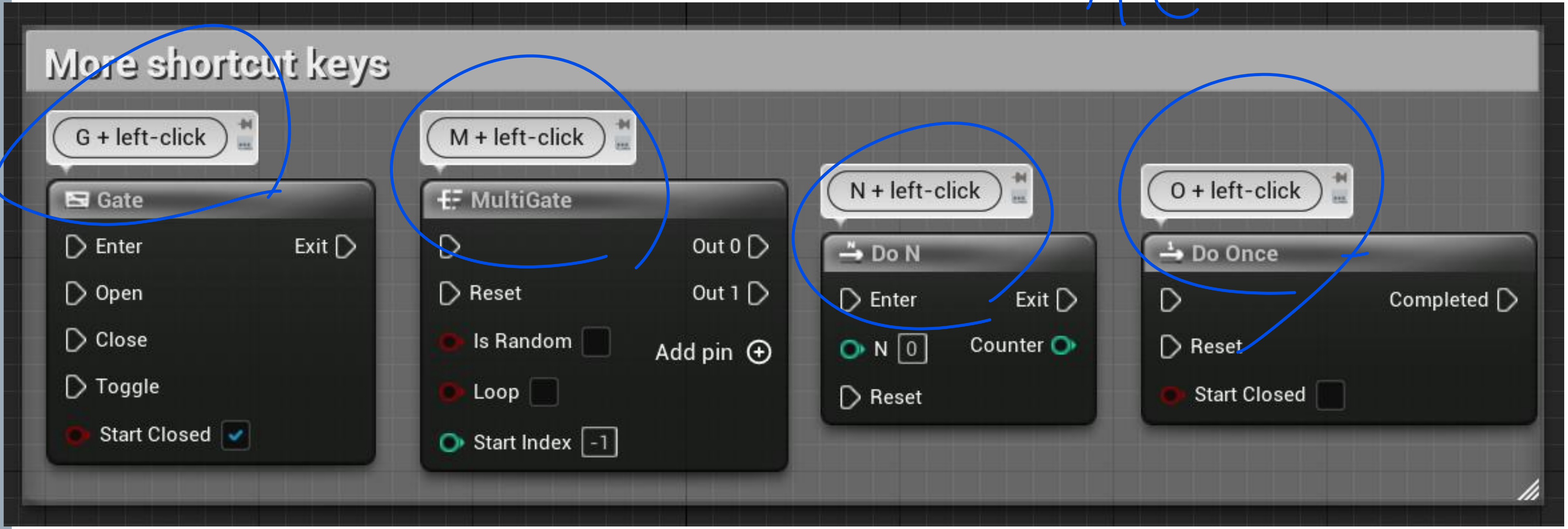
- F + left-click: to create a **For Each Loop** node D 클릭
- D + left-click: to create a **Delay** node



< Shortcuts for the For Each Loop and Delay nodes >

Blueprint Editor shortcuts (9)

- To create comment box around some nodes, first select the nodes, then right-click on one of the selected nodes and select the **Create Comment** option from **Selection**, or you can just press the C key. 단축키 | C

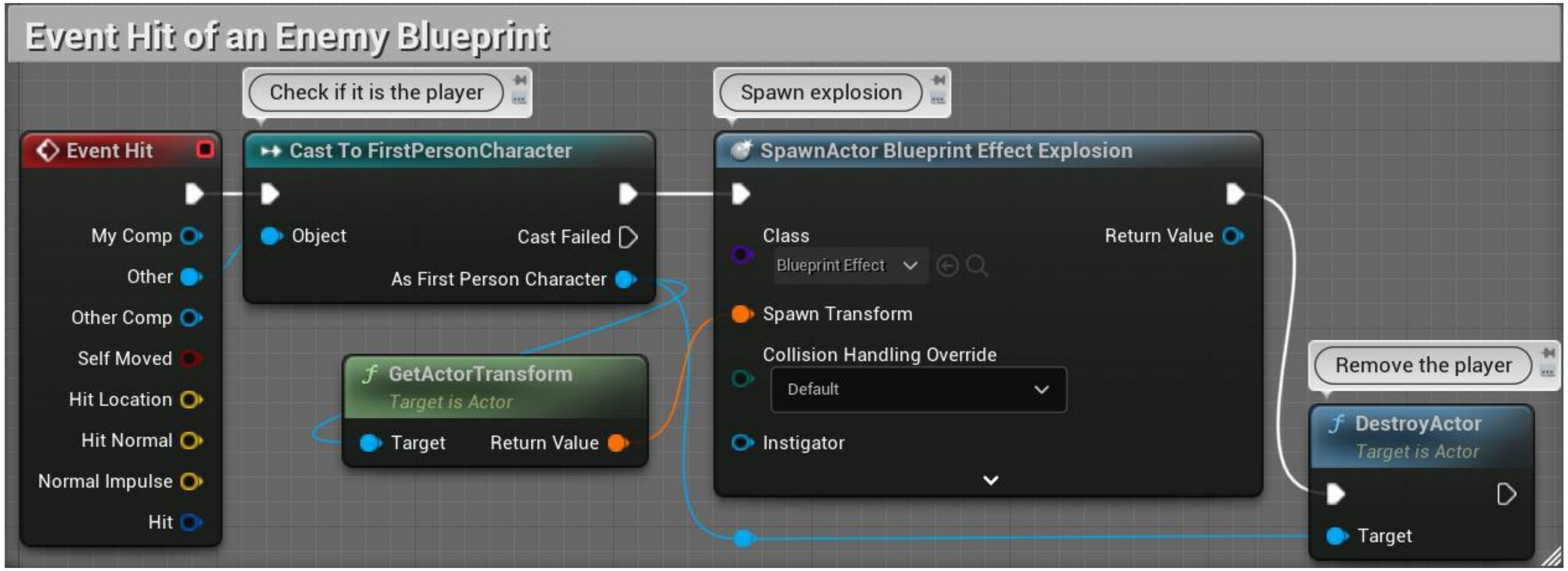


< Shortcuts for some flow control nodes >

Blueprint best practices (1)

- Blueprint responsibilities
  - When creating a Blueprint, you need to decide what its responsibilities will be.
  - This refers to what it will do and what it will not do.
  - You need to make the Blueprint as independent as possible.
  - A Blueprint must be responsible for its internal state.

Blueprint best practices (2)

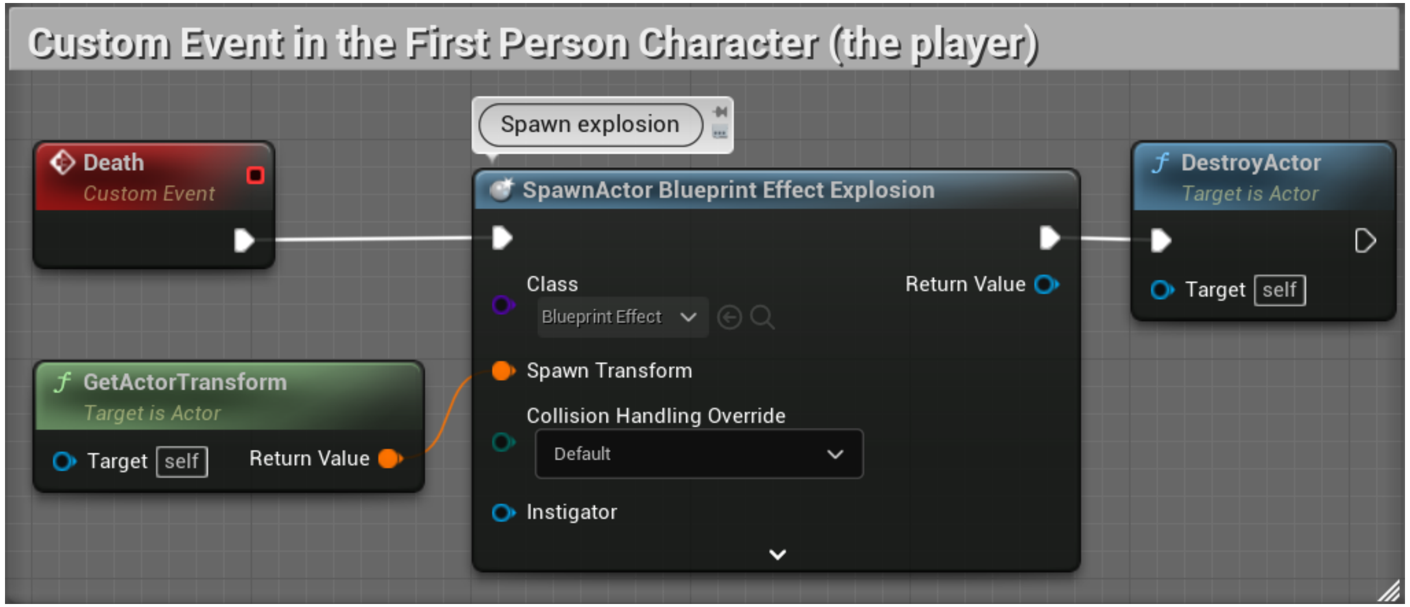


Class: Blueprint Effect

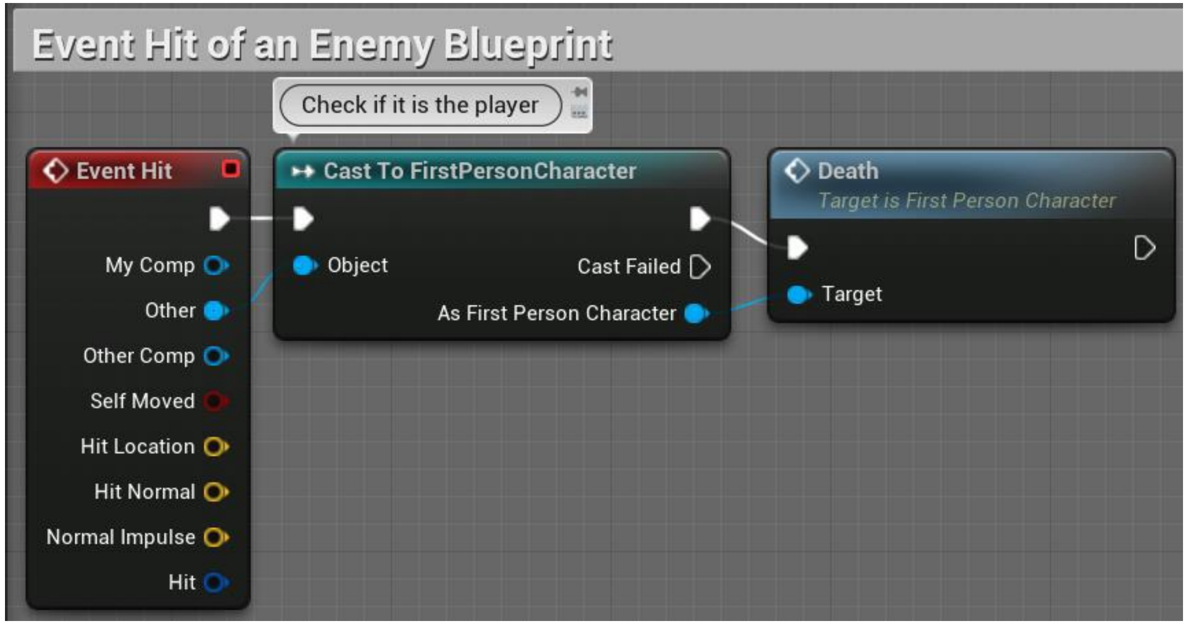
Collision Handling Override: Default

< Event hit o an enemy Blueprint >

– But you decide to change the way the player dies.



< Creating the Death event in the FirstPersonCharacter Blueprint >


Event Hit of an Enemy Blueprint
Check if it is the player
Event Hit
My Comp
Other
Other Comp
Self Moved
Hit Location
Hit Normal
Normal Impulse
Hit
Cast To FirstPersonCharacter
Object
Cast Failed
As First Person Character
Death
Target is First Person Character
Target

< New Version of Event Hit of Enemy Blueprint >

룰이 이동한
게임위치

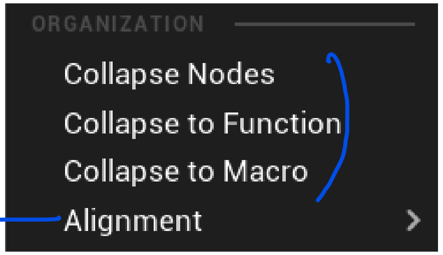
Blueprint best practices (3)

- A Level Blueprint must be used only for logic and situations specific to one Level.
  - If your game rules logic changes, then you will need to all the Level Blueprint of the new Level.
  - A better place to implement game rules logic is in a GameMode Blueprint class.
  - The logic for other actors should be implemented in Blueprint class rather than being implemented in the Level Blueprint.

해당 레벨에서만
사용하는 로직만
넣는다

Blueprint best practices (4)

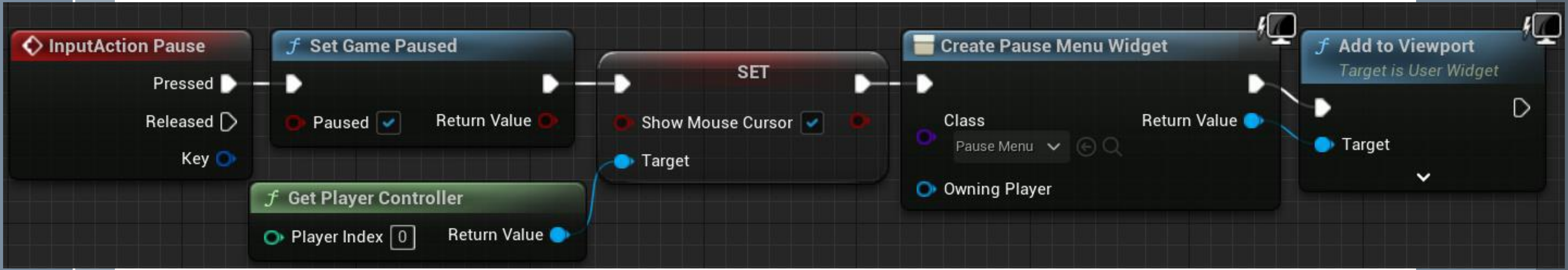
- Managing Blueprint complexities
  - A Blueprint **EventGraph** can become very complex and scary.
  - **Abstraction** is used to handle complexities by hiding low-level details, allowing the developer to focus on a problem at a high abstraction level.
  - A simple way to apply abstraction: to select a group of nodes and convert them into a collapsed graph, Function, or Macro.
    - To convert the nodes, right-click on the selected nodes.



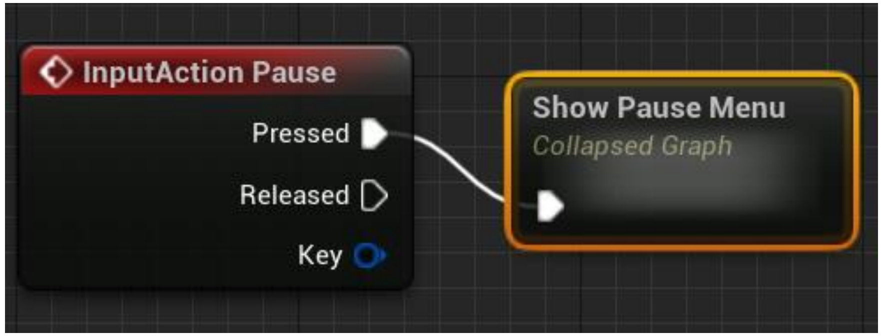
< Collapse options >

정렬

복잡한 노드를
압축한다
가독성 위해



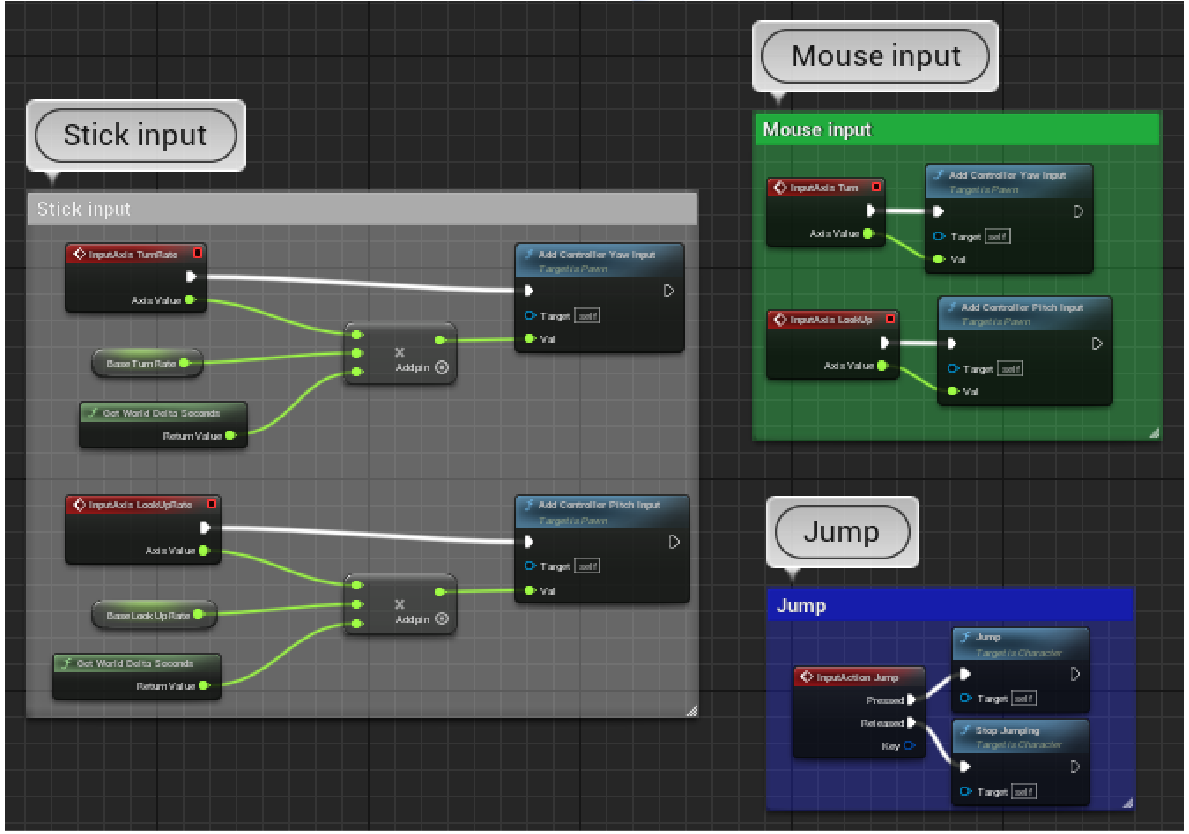
< Node used to show the Pause Menu >



< The nodes were converted into a collapsed graph >

Blueprint best practices (5)

– Another handy tool that can increase the readability of a complex **EventGraph** is comment box



< The comments are visible when the EventGraph is zoomed out >

Blueprint best practices (6)

Blueprint best practices (7)

Blueprint best practices (8)

Blueprint best practices (9)