# **Visionaire Basics for beginners**

Video 10 – IF Statements

## **Introduction – IF statement**

In this section we will take a closer look at IF statements.

So what are IF statements? IF statements are simply a way to create branching actions within your game. At it's core it's a statement that have a different end result depending on the condition we use.

Look at the below where a character will have two different responses when we look at an object depending on whether we have already looked at our Fish tank object.

IF Already\_Looked\_at\_object = false

Then character says "Hey, It's a fishtank"

### ELSE

Then character says "That fishtank has a crack in it!"

So in the above the character looks at the fishtank object the first time and then has a response. If the player looks at the object a second time there will be a more detailed response.

## **Our first IF statement**

The first thing here is to define where this action will occur. In our case we will have these responses occur when we LOOK at the FISHTANK object.

1. On your Fishtank object add a new action with command LOOK AT

Objects Way systems Actions			
$+ - \vee \wedge \diamond \uparrow$			
KeyObjectOnScreen 🔀	8 - A - A - A - A - A - A - A - A - A -		
Door			
PlayerStart Position			
FishTank			
	Properties Effects Actions Anim		
	$+ - \land \lor \diamond$	Execution type	Executed command on object
	'Look' executed	Command	Look
		ltem	[empty]
		+ -	

2. Add a new Action part called IF CONDITION

Select action part		x		
Most frequently used action parts	СНАВАСТ	FR		
Display text	COENIC			
Change condition	SCENE			
Change scene	CONDITION, IF	, VALUE		
End if	INTERE	<del>If character has</del> tem		
Fise		If lua result		
Show scene/menu	ANIMAT	Wait until value was changed		
Add/Remove item	ACTION, S	Wait until condition was changed		
Set cursor		Start tween on value		
Start dialog/dialog-layer	MOLTIMI	Change condition		
Pause	SAVEGA	Set value		
	MISCELLAN	Set vardem value		
	L			
		If value		
		If character is on scene		
		If character is current character If character is aligned to		

3. We now have an IF statement that needs something to check. Refer back to the original example.

## IF Already\_Looked\_at\_object = false

Right now we need to create the *Already\_Looked\_at\_object* condition to check. This we then add in the [EMPTY] field.

Execution type	Executed command on object
Command	Look
	[empty]
$+ - ^{-}$	
🗏 If condition "[e	mpty]' is true

- a. Create a new condition under the CONDITIONS tab on your object.
- b. Set it to FALSE. The reason here is that we have not yet looked at the FishTank object when the scene just loaded so this should be FALSE by default.

Properties Effects Actions Anim	ations Conditions Values Action Action areas Character	
+ <b>-</b> ^	Variable Ocombined condition	
Already_looked_at_object	🔵 True 🚺 False	
	+ -	

- 4. Go back to ACTIONS page.
- 5. Click on IF CONDITION action part and note the properties we have.

Properties Effects Actions Anim	nations Condition	s Values Action Action areas Ch	arad	ter size Pre	view
$+ - \wedge \vee \Diamond$		Executed command on object	-	Action part	If condition 🗸 🥒
'Look' executed		Look	•	Condition	[empty] 🛛 🝙 🗙
		[empty]	Ŧ	Equals	True
	$+ - \wedge \vee$				False
	■ If condition '[e	mpty]' is true			Elself

- a. In the Condition area we set which condition should be checked.
- b. The equals sets what the IF statement should evaluate to continue the process.
- 6. Click on the small icon to the right of the CONDITION field and select the condition you created and set the equals to FALSE.

Execution type	Executed command on object	Ŧ	Action part	If condition	•	Ø
Command	Look	•	Condition	Already_looked_at_object	9	×
Item	[empty]	•	Equals	True		
$+ - \wedge \vee$			False			
■ If condition 'Al	ready looked at object is false			Elself		

As per our original example we have now completed the following;

IF Already\_Looked\_at\_object = false

7. Now, let's add the next section which is the the display text.

IF Already\_Looked\_at\_object = false

Then character says "Hey, It's a fishtank"

To do this, simply add a display text with the relevant text.

Execution type	Executed command on object	-	Action part	Display text 🗸 🗸	
Command	Look	-	Character	MainCharacter	-
ltem	[empty]	-	Text	Hey, It's a fishtank	
$+ - ^{-}$	$\vee$				
E If condition 'Al	ready_looked_at_object' is false				
MainCharacter	: Hey, It's a fishtank				
				▶ ■ ··· × Q 🖬	
				Show as background text	

8. Next we need to set what is said when *Already\_Looked\_at\_object* condition is set to true.

IF Already\_Looked\_at\_object = false Then character says "Hey, It's a fishtank"

ELSE

Then character says "That fishtank has a crack in it!"

9. This means we need to add an 'ELSE' statement. Click and add a new Action part called ELSE



#### 10. Our setup looks as follows;



11. Add a second display text action part with a different text.

Properties Effects Actions Anin	nations Condition	Values Action Action area	eas Character size Preview
$+ - \wedge \vee \Diamond$		Executed command on object	
'Look' executed		Look	
		[empty]	
	$+ - ^{1}$		
	<ul> <li>If condition 'Al MainCharacter</li> <li>Else</li> <li>MainCharacter</li> </ul>	eady_looked_at_object' is false Hey, It's a fishtank That fishtank has a crack in it!	

12. There is one more VERY IMPORTANT step we need to add. We need to tell Visionaire where the IF STATEMENT ends. We do this by adding a END IF action part right at the end of our statement. Add a new action part as per below.

Select action part		×
Most frequently used action parts	CHARACTER	
Display text Change condition		
Change scene End if	CONDITION, IF, V	Higheracter has item
		If lua result
Else Show scene/menu		Wait until value was changed
Add/Remove item		Wait until condition was changed
Set cursor Start dialog/dialog-layer		Change condition
Pause		
X		
LUUN		If value
+ - ^ ~		
If condition 'Already_looked_at_o		
Fise		
MainCharacter: That fishtank has a	crack in it!	
		Else
		End if

## 13. Our IF statement looks as follows;

 	,					
$+ - \land \lor \Diamond$	Execution type	Executed command on object	-	Action part	End if 🔶 👻	
		Look				
		[empty]				
	$+ - ^{-}$					
	<ul> <li>If condition 'Already_looked_at_object' is false</li> <li>MainCharacter: Hey, It's a fishtank</li> <li>Else</li> </ul>					
	MainCharacter End if	: That fishtank has a crack in it!				

14. This will still not work and we will explain why in a moment. For now play your scene and note what happens when you LOOK at the FISHTANK object.

Notice that it doesn't matter how many times we LOOK at our object it always shows the first display text and never the next text. Why is this?

Simply it means the condition Already\_looked\_at\_object always evaluates to FALSE. The reason is that nowhere did we specify that this should change to TRUE!

Obviously when we look at an object the first time then this object should change to TRUE.

#### 15. Add a new ACTION part called CHANGE CONDITION.

Select action part		×			
Most frequently used action parts	CHAR	ACTER			
Display text Change condition	SCENE				
Change scene End if	CONDITIC	lf character has item			
If condition	INTE	If lua result			
Else Show scene/menu	ANI	Wait until value was changed			
Add/Remove item	ACTIO	Wait until condition was changed			
Set cursor Start dialog/dialog-layer	MUL	Change condition			
Pause	SAV	Set value			
Call/Quitaction 🗙	MISCEL	Set random value			
		If condition			

### 16. It's then added into your action.

Properties Effects Actions Anim	ations Condition	s Values Action Action areas	Chara	cter size Preview		
$+ - \land \lor \diamond$		Executed command on object			Change condition	
'Look' executed		Look				
		[empty]			🔵 True	
	$+ - ^{-}$				False	
	■ If condition 'Al MainCharacter:	ready_looked_at_object' is false : Hey, It's a fishtank ap 'femptyl' to true		- Toggle		
	■ Else MainCharacter: End if	That fishtank has a crack in it!				

17. Notice that if it's placed in the incorrect spot in your statement you can change it using the arrows shown



18. Next set which condition should change and what should change. We need to choose the condition **Already\_looked\_at\_object** to True.



19. Our condition should change when we've look at the object at least once. This is the reason why it's placed in the location shown below.

Properties Effects Actions Anim							
+ <b>-</b> ^ \ Q		Executed command on object				Change condition	- 🖉
		Look				Already_looked_at_object	യ 🗙
	Item [empty]					💿 True	
$+ - \wedge \vee$					False		
	<ul> <li>If condition 'Already_looked_at_object' is false</li> <li>MainCharacter: Hey, It's a fishtank</li> <li>Change condition 'Already_looked_at_object' to true</li> <li>Else</li> <li>MainCharacter: That fishtank has a crack in it!</li> <li>End if</li> </ul>				U loggle		

20. Now play your scene