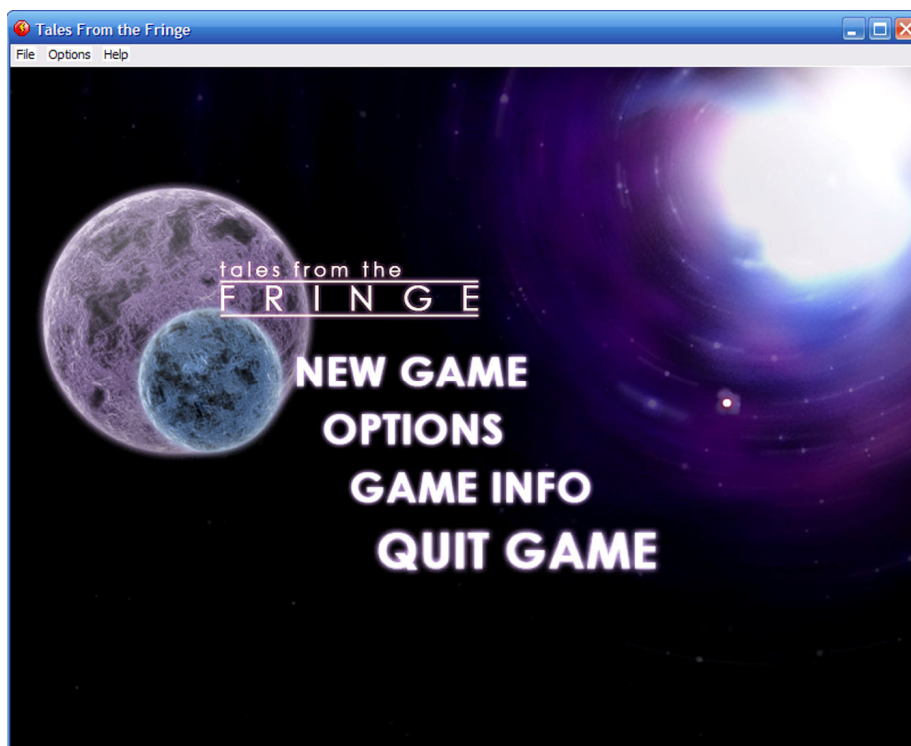


HOW TO CREATE
A GREAT LOOKING

MENU

...AND LIVE TO TELL THE TALE.

PART ONE THE TITLE SCREEN



You may not use this tutorial for any other purpose than learning, working or having fun... In other words: You can use this PDF tutorial for anything you'd like, as long as it doesn't involve both a hammer and a squirrel.

Koobare
marchewkowy@gmail.com

Welcome to another one of Koobare's little tutorials, teaching you – as always – how to effectively and efficiently use the best multimedia authoring tool ever – [Multimedia Fusion 2](#) by Clickteam! This tutorial is meant for novices, people who just started to explore the fascinating world of MMF2, yet you should have a basic grasp of Fusion's interface and functions before getting yourself familiar with this little document (you should read the "Interface Guide", "Glob Wars" and "Smelly Claw" tutorials first).

Enhancing the Feel...

Don't know if you remember, but some time ago I've written two tutorials about implementing weather effects into your game projects – nothing too fancy, but they explained how to easily add nice rain, snow and wind effects to any type of a side-viewed game (actually, they could as well be used in most top-down games as well). These two tuts (oh boy, were those created ages ago...) were just a part of a longer series, intended to last, showing you how to easily improve the looks and the overall "feel" of your games, no matter if your goal is to build a strategy game, a side-scrolling platformer or an action-packed multiplayer shooter.

I named this series "**Enhancing the Feel**". Try to remember that, since it's here to stay! From time to time I'll be releasing new tutorials from this series, on how to almost effortlessly add new content to your games – content helping you not only to enhance the gameplay experience for each player, but also teaching you a few tricks that might come in handy somewhere during your game development process. You'll see – it'll prove helpful!



the good old times...

Whether you'd like to add a nice weather effect to your game, enhance it with a day & night system, or just create a great looking fireball effect – stay tuned, we'll get there sooner or later!

How to create a great looking title screen and live to tell the tale...

So, we now know what the whole series of “Enhancing the Feel” tutorials will try to achieve – but what about this one specific tutorial that you’re reading right now? Well, this won’t be a shocker if you already read the title (and I’m assuming that you’re one of those people who start reading from page *numero uno*), but we’re going to create a really cool-looking main menu and a title screen for a science-fiction game that I’ll turn into another tutorial one day.

When you’re creating the title and main menu screens for your game, always remember: this is the first thing your player is going to see. Sure, this isn’t as important as gameplay features or a good story, but it still can help to get your player really attracted to the game itself, excited about playing your creation. A well constructed menu and title screens can really help the player to FEEL the world you’ve created.



That’s why this tutorial will be devoted to creating an attention-grabbing main menu, options and title screens. **This part will be all about the title screen** – after completing it, turn to part two, where you’ll find the main menu and options. I decided to divide this tutorial into two parts to keep the usual length of about 18-20 pages, and to keep it small, easier to download and to send by e-mail. We will use sounds, music, transitions and a couple of moving active objects to create a nice mood of interstellar travels, mysterious uncharted worlds and time-warping black holes... Welcome to the world of the “Tales from the Fringe”, cadet! Arm your laser sidearm, get yourself into one of those spacesuits and say farewell to this galaxy... ‘Cause we’re leaving!

- 📄 If you have any problems with this tutorial, or notice that there are some mistakes present, please, contact me and I'll do my best to help you and replace all the errors with correct information.

Contact me at: marchewkowy@gmail.com

- 📄 **Note:** I've been receiving some reports that not all e-mails get to me for some reason. Seems that some of them (quite a lot) end up in my spambox or are blocked out by the server. I dunno why this is happening, so if you're experiencing any difficulties with delivering me a message or haven't received a reply in quite some time, please, send me another e-mail at marchewkowy@wp.pl. I'll do my best to check both these e-mails regularly.

Part I: We're leaving this galaxy.

Time to get our title screen up and ready! Open Multimedia Fusion 2, **create a new application** and save it onto your hard drive (a little reminder: it's a good thing to have the *Autobackup* option of MMF2 turned on – check your *Preferences*). Rename it to “**Tales from the Fringe**”. Now, go to your application's **Properties window** (if it didn't open up by itself, right click on your application's name in the workspace toolbar and select *Properties* from the drop-down menu), and select the **Window** tab (second from the left). Set the window size to **800x600**. If MMF2 asks you if you'd

like to modify the size of the existing frame as well, select “Yes”. When that's done, create two more frames (we want three of them in our application) and name them respectively: **Title Screen**, **Main Menu** and **Options**. Make sure that

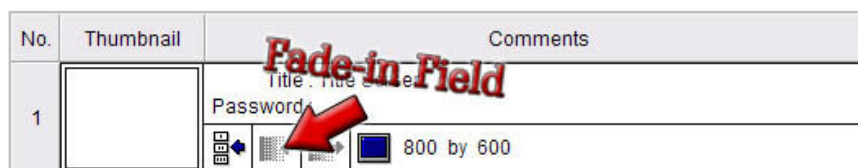
all three of them are set to a **800x600** size. If you wish, you may brand our game – return to the **Properties window** of your application (you may need to click the name of your application in the *Workspace toolbar*), select the **About** tab (first from the right), change the “Description” field to “Tales From the Fringe”.

No.	Thumbnail	Comments
1		Title : Title Screen Password : 800 by 600
2		Title : Main Menu Password : 800 by 600
3		Title : Options Password : 800 by 600

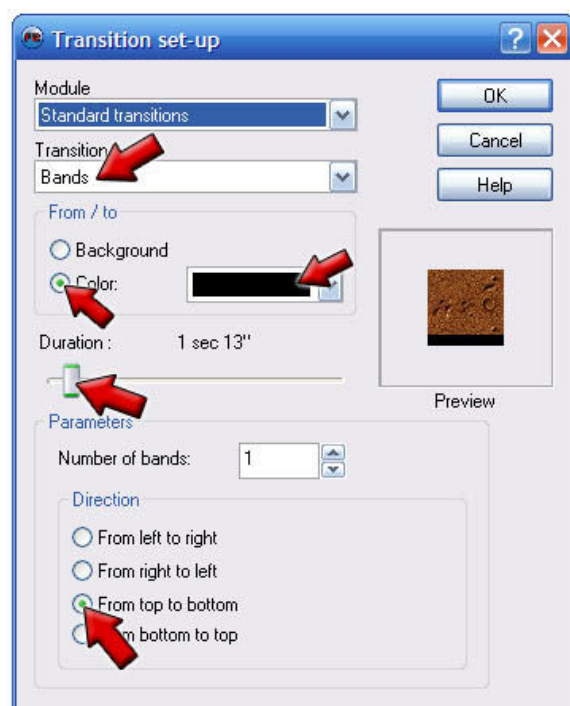


Once you're done with that, open up the **Values** tab (third one from the right, the one with the "A-Z" symbol), which will enable us to edit the **Global Values** for our application. Create three new Global Values and name them respectively: **Music**, **Menu Animation** and **Safety**. "Music" and "Menu Animation" will be responsible for controlling our game options (shutting the music on/off and stopping the optional movement of the menu), whereas "Safety" will be used as a safety switch just to make sure that everything goes well within a single event loop. **Set Music to 1, Menu Animation to 1 and Safety to 0.** This will be pretty important in part two of this here tutorial, so don't mess it up...

Part II: Transitions.



It's now time to edit the **transition effects** that will appear during the game, between our

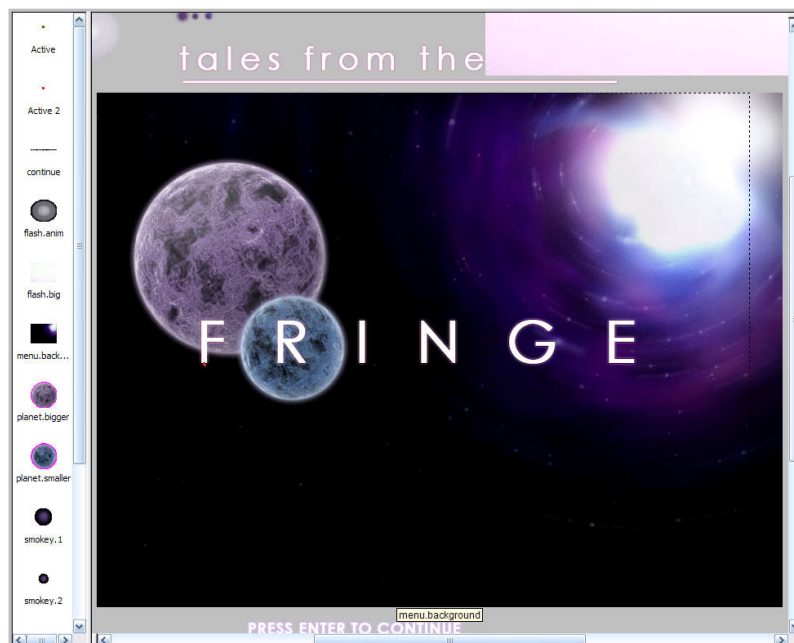


frames. Open up the **Transition setup** for the fade-in effect of the first frame (right-click on the fade-in field and choose "Transition setup" from the context menu), set it to **Bands**, change the duration to **1.13** of a second, change the direction to "**From top to bottom**" and – finally – set it to appear from **black**. Change the fade-out transition of the first frame to **Fade**, set the duration to **1.35** of a second, make sure that it also appears from **black**. Now, for the second frame: both fade-in (duration: **1.01**, from black) and fade out (duration: **1.35**, from black) set as the **Fade effect**. Exactly the same transitions should be set up for the third frame.

Part III: To import, or not to import – that is the question...

Once we're done with transitions, it's time to import all the needed objects into our frames... This will be a bit boring, so... Hey, you know what? Let's just skip this. We'll cheat a bit, but it's for a good cause – to spare you all that irritating “copy, paste, reposition” stuff. Just close what you've done so far and open up the **menu-tutorial-part-one.mfa** file (should be in the same directory as this tutorial, since they were packed together into the same archive).

Please note: most of the objects use alpha channels, a feature that is unavailable in Games Factory 2 (TGF2 users should use basic library objects or create their own graphics instead – I generally would advise you to upgrade to MMF2 as soon as possible, since you're missing out on some really good stuff, a lot of quite impressive features).



Now, once we've opened **menu-tutorial-part-one.mfa**, open up the first frame (“Title Screen”) and take a look around. There are quite a lot of different objects laying around, right? Two planets, a background picture (which actually is a moving active object), some strange actives... Don't be worried, though, we'll know all there is to know about them in a jiffy!

Part IV: Knowing what's what.

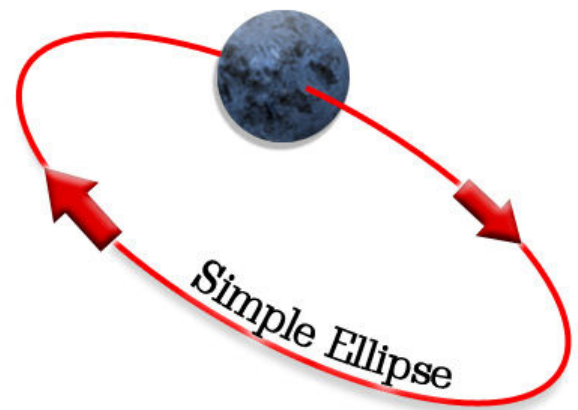
It's now time to learn something about the thingies that are laying around in our first frame. Below you can find an alphabetical list of all the objects, with their short description, their purpose and some characteristic properties all written down:

Object's name:	So... What is it?
continue.text	<p><u>Active object - a bitmapped text saying "Press Enter to continue".</u></p> <p>MOVEMENT</p> <p>What's worth noticing is that its movement is set to Simple Ellipse (with Angular Velocity at 120, Radius X at 25, Radius Y at 12, Centre X at 400, Centre Y at 550, Starting Angle and Offset Angle both set to 0. This means it orbits around the given X-Y Centre.</p> <p>ANIMATION</p> <p>If you'll open up this object in the Animation Editor, you'll notice that it has a "pulsing" animation, played and looped over a spread of 21 frames.</p>
flash.anim	<p><u>A "spaceflash" animation, depicting a sudden burst of light in a distance.</u></p> <p>TRANSITIONS</p> <p>It has it's Fade-in and Fade-out transitions set to Zoom (the Fade-out animation lasts a bit longer than the Fade-in one).</p> <p>ANIMATION</p> <p>Open it up in the Animation Editor to check out how it's 39-frame animation looks like – it basically is a whitey big dot stretching itself to imitate the growing of a very distant burst of light.</p>
flash.big	<p><u>Just a big white rectangle, capable of covering the whole screen,</u></p> <p>TRANSITIONS</p> <p>It has it's Fade-in and Fade-out transitions set to the Fade effect, with the Fade-out lasting a lot longer (3.5 seconds) than the very short Fade-in (0.22 of a second).</p> <p>WHY IT'S HERE?</p> <p>It will be used to "blind" the player (cover the screen with a white veil for a few seconds), when the distant flash of light arrives at his location.</p>
menu.background	<p><u>A moving active object used as the background for our frame.</u></p> <p>MOVEMENT</p> <p>I've set it up with a Simple Ellipse movement (Angular Velocity at 180, Radius X and Radius Y both set at 15, Centre X and Centre Y both set at -15, Starting Angle and Offset Angle both set at 0), so it rotates frantically, as if the player looked at the screen from an ever-revolving spaceship.</p>
planet.bigger	<p><u>A big purple planet, spinning in the background.</u></p> <p>MOVEMENT</p> <p>Another object that I equipped with a Simple Ellipse movement (Settings: Angular Velocity at 170, Radius X at 55, Radius Y at 100, Centre X at 200, Centre Y at 230, Starting Angle and Offset Angle both set at 0), to make it rotate in orbit, in the distance. When placed on a rotating background, it really adds to a nice overall effect.</p>

planet.smaller	<p><u>A blue planet, spinning in front of the bigger, purple one.</u></p> <p>MOVEMENT</p> <p>It is equipped with a Simple Ellipse movement (Angular Velocity at 170, Radius X at 95, Radius Y at 135, Centre X at 270, Centre Y at 284, Starting Angle and Offset Angle both set at 0).</p>
seeder.1	<p><u>Object used to seed the “smokey” objects (to create the purple smoke).</u></p> <p>WHY IT'S HERE?</p> <p>The purpose of this object will become pretty clear later on, for now you just need to know that it's used to create a nice smoke-related graphical effect, which will follow the shape of the “Fringe” part of our game's logo.</p> <p>MOVEMENT</p> <p>Path movement, with the path following the silhouette of the “Fringe” text.</p>
seeder.2	<p><u>Active object used to “seed” the “smokey” objects.</u></p> <p>WHY IT'S HERE?</p> <p>Used for the same stuff as “seeder.1” object, it also has a similar Path movement, yet the path is a bit different (check it out for yourself).</p>
smokey.1 (smokey.2 and smokey.3)	<p><u>Three different actives used to create a nice purple smoke effect.</u></p> <p>WHY IT'S HERE?</p> <p>They are created at the current position of the “seeder.1” and “seeder.2” objects, and are destroyed almost instantly. When vanishing, they go through a Zoom or Fade transition, creating an eye-pleasing graphical effect that resembles purple smoke.</p>
title.fringe	<p><u>Active object – a part of the game's logo (“Fringe”).</u></p> <p>WHY IT'S HERE?</p> <p>It is a part of the game's title (“Tales from the Fringe”), that appears from the depths of space. It's Fade-in transition is set to Zoom, so it basically zooms into the view of the player, over a span of 8.47 seconds.</p>
title.line	<p><u>Another part of the game's logo (line).</u></p> <p>WHY IT'S HERE?</p> <p>A line that is used under and over the “Fringe” part of the game's logo. It has a very slow Fade-in transition (Fade effect, set to 8.47 seconds).</p>
title.tales	<p><u>A crucial part of our game's logo (“Tales from the”).</u></p> <p>WHY IT'S HERE?</p> <p>Another part of the game's logo. It has a very slow Fade-in transition (Fade effect, set to 7.91 seconds). It is created onto the game frame at 6th second after the start of the frame.</p>

Simple Ellipse movement

As you can see, many of our title screen's objects were given the “**Simple Ellipse**” movement, a neat little movement ideal for making an object orbit around a certain point. It is controlled via a set of properties, such as: **Angular Velocity** (the speed of movement), **X-Y Radius** (the length of movement in the X and Y scale), **X-Y Centre** (centre of the movement – the rotating object will revolve around this point), the Starting angle and Offset angle.



Smoke effects

One of the easiest ways to create a good looking **smoke effect** in MMF2 is by using an alpha blended object with blended corners (for example, created in Photoshop, saved as a .PNG and then imported into Multimedia Fusion 2), setting it's **Fade-out transition** to either a fast **Zoom** or **Fade**, then spawning it every fraction of a second to the position of a fast-moving object (player's cursor, another active) and – what's important – instantly destroying it with an event based on the “Always” condition. You'll see how this is going to work in a jiffy.

Part V: Time for a bit of programming.

It's time for some fun! Save your project (always remember to save it from time to time!) and open up the **Event Editor**. If you're new to my tutorials, let me introduce you to the event-recording system that I use. If you know it already – just skip this frame below and quickly move on to the coding part:

Koobare's MMF-to-paper coding system

IF (Condition): [Object for the condition] > Condition group > Condition

THEN (Action): [Object for the action] > Action group > Action

Seems simple, right? Well, that's just because IT IS simple. All the conditions are marked in red, while actions are written in fancy blue.

Object names are always put in [square brackets]. The final condition/action is always in *Italic*. If we'll have a multi-condition event, then it'll be like this:

IF (Condition 1): [Object for condition 1] > Condition group 1 > *Condition 1*

IF (Condition 2): [Object for condition 2] > Condition group 2 > *Condition 2*

THEN (Action): [Object for the action] > Action group > *Action*

Whereas a multi-action event looks like this:

IF (Condition): [Object for condition] > Condition group > *Condition*

THEN (Action 1): [Object for the action 1] > Action group 1 > *Action 1*

THEN (Action 2): [Object for the action 2] > Action group 2 > *Action 2*

If you'll have to input anything by keyboard, it will be indicated by coloring the text green and using < angle brackets >, like this:

< Set the Global Value A to 32 >

Additional comments, instructions and info will be put in << double angle brackets >>, using a different color:

<< Select any wave sound from the MMF2's sound library >>

From time to time I'll also use this style to throw in some extra tips and tricks about MMF2 and more advanced coding techniques.

All you have to do is to go step-by-step through all the listed events and keep one eye on your Event Editor, and the second one on this tutorial...

Let's make our title screen work...

1) Firstly, let's start off with the conventional **"Start of frame"** event, which I usually create at the very beginning of the events list. This event – triggered when someone opens up our game – will destroy three objects that I've created earlier in the frame, but are not yet needed (I want them to be created later on, at a specific time):

IF: **[Storyboard Controls] > Start of frame**
 THEN: **[continue.text] > Destroy**
 THEN: **[seeder.1] > Destroy**
 THEN: **[seeder.2] > Destroy**

And that's that – we've got our first event ready. Thanks to this little gizmo we can be sure that neither those two seeders, nor the "Press Enter to continue" text will spring up to the screen before they are needed.



2) Time for some more events, this time sound-related... Here's eight simple events that will play the samples chosen by me in the given time. Be sure to choose the same samples as I did to achieve the maximal effect – you can find them all on the "Bonus CD" of your Multimedia Fusion 2 pack, in Clickteam's multimedia libraries (if I remember correctly they are all from the "Science Fiction" directory):

IF: **[The Timer Object] > Is the timer equal to a certain value?**
 << Set the timer to 0.50 of a second >>
 THEN: **[Sound Object] > Samples > Play sample**
 << Choose the "Auto hatch.wav" sound >>

Now, here are events three to nine:

IF: **[The Timer Object] > Is the timer equal to a certain value?**
 << Set the timer to 2 seconds >>
 THEN: **[Sound Object] > Samples > Play sample**
 << Choose the "Cloud City.wav" sound >>

IF: **[The Timer Object] > Is the timer equal to a certain value?**
 << Set the timer to 3.80 seconds >>
 THEN: **[Sound Object] > Samples > Play sample**
 << Choose the "Cargo hold.wav" sound >>

IF: **[The Timer Object] > Is the timer equal to a certain value?**
 << Set the timer to 4.00 seconds >>
 THEN: **[Sound Object] > Samples > Play sample**
 << Choose the "WHOOSH09.wav" sound >>

IF: [The Timer Object] > *Is the timer equal to a certain value?*

<< Set the timer to 5.50 seconds >>

THEN: [Sound Object] > Samples > *Play sample*

<< Choose the “Craft land.wav” sound >>

IF: [The Timer Object] > *Is the timer equal to a certain value?*

<< Set the timer to 10.50 seconds >>

THEN: [Sound Object] > Samples > *Play sample*

<< Choose the “Law and Order.wav” sound >>

IF: [The Timer Object] > *Is the timer equal to a certain value?*

<< Set the timer to 13.00 seconds >>

THEN: [Sound Object] > Samples > *Play sample*

<< Choose the “Law and Order.wav” sound >>

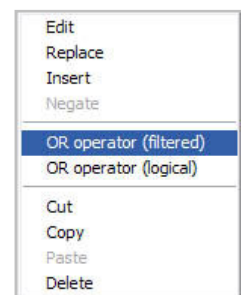
IF: [The Timer Object] > *Is the timer equal to a certain value?*

<< Set the timer to 16.50 seconds >>

THEN: [Sound Object] > Samples > *Play and loop sample*

<< Choose the “London 2098.wav” sound, loop it 0 times, that means infinitely >>

3) Got it? Great! Time to move on to the next event, enabling the player to skip this frame when he presses either the **Enter** key or the **Space bar**, or – alternatively – clicks with the **left mouse button**... Note that we’re using the “**OR**” operator here, enabling us to create alternative conditions in one event – you can add it by right clicking on a condition and then choosing “**OR operator (filtered)**” from the context menu:



IF: [Keyboard & Mouse Object] > The Keyboard > *Upon pressing a key*

<< press ENTER on your keyboard >>

<< OR operator (filtered) >>

IF: [Keyboard & Mouse Object] > The Keyboard > *Upon pressing a key*

<< press SPACE BAR on your keyboard >>

<< OR operator (filtered) >>

IF: [Keyboard & Mouse Object] > The Mouse > *User clicks*

<< select: LEFT BUTTON, SINGLE CLICK >>

THEN: [Storyboard Controls] > *Next frame*

Thanks to the “OR” operator we don’t have to create a separate event for all of these three conditions – no matter whether the player clicks with LMB or presses ENTER, it’ll work.

5) And here we've got three time-conditioned events that are responsible for controlling the "spaceflash" effect:

IF: [The Timer Object] > *Is the timer equal to a certain value?*

<< Set the timer to 1 second >>

THEN: [Create New Objects] > *Create Object*

<< Select the [flash.anim] object >>

<< Set the coordinates to x=400, y=306 >>

IF: [The Timer Object] > *Is the timer equal to a certain value?*

<< Set the timer to 5.15 seconds >>

THEN: [Create New Objects] > *Create Object*

<< Select the [flash.big] object >>

<< Set the coordinates to x=0, y=0 >>

IF: [The Timer Object] > *Is the timer equal to a certain value?*

<< Set the timer to 6 seconds >>

THEN: [flash.big] > *Destroy*

Usually I would suggest merging the event above (the one beginning with the "timer equals 6 seconds" condition) with the identically conditioned event in point 5, but let's keep it this way this time, just to show you that it is possible to make it in such way, even if it's a bit untidy.

6) And here's another event to create...

IF: [The Timer Object] > *Is the timer equal to a certain value?*

<< Set the timer to 12.50 seconds >>

THEN: [Create New Objects] > *Create Object*

<< Select the [continue.text] object >>

<< Set the coordinates to x=400, y=550 >>

7) And another one... This one brings our "seeder.1" and "seeder.2" objects into the frame, enabling them to start spawning all that purple smoke all over the place:

IF: [The Timer Object] > *Is the timer equal to a certain value?*

<< Set the timer to 10.50 seconds >>

THEN: [Create New Objects] > *Create Object*

<< Select the [seeder.1] object >>

<< Set the coordinates to x=132, y=334 >>

THEN: [Create New Objects] > Create Object
 << Select the [seeder.2] object >>
 << Set the coordinates to x=132, y=334 >>
 THEN: [seeder.1] > Visibility > Make object invisible
 THEN: [seeder.2] > Visibility > Make object invisible

8) And here are the events responsible for the creation of this purple smoke:

IF: [The Timer Object] > Is the timer greater than a certain value?
 << Set the timer to 10.50 seconds >>
 IF: [The Timer Object] > Every
 << Set the timer to 0.08 of a second >>
 THEN: [Create New Objects] > Create Object
 << Select the [smokey.1] object >>
 << Set the coordinates to x=0, y=0, relative to [seeder.1] object >>
 THEN: [Create New Objects] > Create Object
 << Select the [smokey.2] object >>
 << Set the coordinates to x=2, y=1, relative to [seeder.2] object >>

<ul style="list-style-type: none"> • Timer is greater than 10"-50 • Every 00"-08 					✓														
--	--	--	--	--	---	--	--	--	--	--	--	--	--	--	--	--	--	--	--

As you can see above – every 0.08 of a second (but only if the overall game timer is greater than 10.50 seconds) a new “smokey.1” and “smokey.2” objects are created at coordinates near “seeder.1” or “seeder.2”...Let’s continue, shall we?

IF: [The Timer Object] > Is the timer greater than a certain value?
 << Set the timer to 10.50 seconds >>
 IF: [The Timer Object] > Every
 << Set the timer to 0.11 of a second >>
 THEN: [Create New Objects] > Create Object
 << Select the [smokey.2] object >>
 << Set the coordinates to x=3, y=0, relative to [seeder.1] object >>
 THEN: [Create New Objects] > Create Object
 << Select the [smokey.3] object >>
 << Set the coordinates to x=0, y=-2, relative to [seeder.2] object >>
 THEN: [Create New Objects] > Create Object
 << Select the [smokey.1] object >>
 << Set the coordinates to x=-1, y=4, relative to [seeder.2] object >>

And here's another one, almost identical...

```
IF: [The Timer Object] > Is the timer greater than a certain value?
<< Set the timer to 10.50 seconds >>
IF: [The Timer Object] > Every
<< Set the timer to 0.14 of a second >>
THEN: [Create New Objects] > Create Object
<< Select the [smokey.3] object >>
<< Set the coordinates to x=-2, y=1, relative to [seeder.1] object >>
THEN: [Create New Objects] > Create Object
<< Select the [smokey.2] object >>
<< Set the coordinates to x=-1, y=-4, relative to [seeder.2] object >>
THEN: [Create New Objects] > Create Object
<< Select the [smokey.1] object >>
<< Set the coordinates to x=4, y=5, relative to [seeder.2] object >>
```

Yet another very similar event... Unfortunately, we have to get through with this if we want to have a nice little smoke effect without playing with math:

```
IF: [The Timer Object] > Is the timer greater than a certain value?
<< Set the timer to 10.50 seconds >>
IF: [The Timer Object] > Every
<< Set the timer to 0.16 of a second >>
THEN: [Create New Objects] > Create Object
<< Select the [smokey.2] object >>
<< Set the coordinates to x=2, y=5, relative to [seeder.1] object >>
THEN: [Create New Objects] > Create Object
<< Select the [smokey.2] object >>
<< Set the coordinates to x=-4, y=4, relative to [seeder.2] object >>
THEN: [Create New Objects] > Create Object
<< Select the [smokey.1] object >>
<< Set the coordinates to x=0, y=6, relative to [seeder.2] object >>
```

You know the drill, just keep it tight and we'll be through with this as soon as possible...

```
IF: [The Timer Object] > Is the timer greater than a certain value?
<< Set the timer to 10.50 seconds >>
IF: [The Timer Object] > Every
<< Set the timer to 0.18 of a second >>
THEN: [Create New Objects] > Create Object
```

```

<< Select the [smokey.1] object >>
<< Set the coordinates to x=-2, y=6, relative to [seeder.1] object >>
THEN: [Create New Objects] > Create Object
<< Select the [smokey.2] object >>
<< Set the coordinates to x=-2, y=2, relative to [seeder.2] object >>
THEN: [Create New Objects] > Create Object
<< Select the [smokey.1] object >>
<< Set the coordinates to x=4, y=1, relative to [seeder.2] object >>

```

This one is a little bit longer, more smokeys created at once...

```

IF: [The Timer Object] > Is the timer greater than a certain value?
<< Set the timer to 10.50 seconds >>
IF: [The Timer Object] > Every
<< Set the timer to 0.21 of a second >>
THEN: [Create New Objects] > Create Object
<< Select the [smokey.1] object >>
<< Set the coordinates to x=0, y=1, relative to [seeder.1] object >>
THEN: [Create New Objects] > Create Object
<< Select the [smokey.2] object >>
<< Set the coordinates to x=4, y=-2, relative to [seeder.1] object >>
THEN: [Create New Objects] > Create Object
<< Select the [smokey.3] object >>
<< Set the coordinates to x=3, y=3, relative to [seeder.1] object >>
THEN: [Create New Objects] > Create Object
<< Select the [smokey.2] object >>
<< Set the coordinates to x=5, y=2, relative to [seeder.2] object >>
THEN: [Create New Objects] > Create Object
<< Select the [smokey.1] object >>
<< Set the coordinates to x=0, y=-2, relative to [seeder.2] object >>

```

We're almost through with this, so don't give up now, cadet!

```

IF: [The Timer Object] > Is the timer greater than a certain value?
<< Set the timer to 10.50 seconds >>
IF: [The Timer Object] > Every
<< Set the timer to 0.24 of a second >>
THEN: [Create New Objects] > Create Object
<< Select the [smokey.2] object >>
<< Set the coordinates to x=3, y=-3, relative to [seeder.1] object >>

```

THEN: [Create New Objects] > *Create Object*
 << Select the [smokey.3] object >>
 << Set the coordinates to x=-1, y=-3, relative to [seeder.1] object >>
 THEN: [Create New Objects] > *Create Object*
 << Select the [smokey.1] object >>
 << Set the coordinates to x=2, y=5, relative to [seeder.2] object >>
 THEN: [Create New Objects] > *Create Object*
 << Select the [smokey.2] object >>
 << Set the coordinates to x=2, y=-5, relative to [seeder.2] object >>

And here's the last event from this batch:

IF: [The Timer Object] > *Is the timer greater than a certain value?*
 << Set the timer to 10.50 seconds >>
 IF: [The Timer Object] > *Every*
 << Set the timer to 0.36 of a second >>
 THEN: [Create New Objects] > *Create Object*
 << Select the [smokey.1] object >>
 << Set the coordinates to x=0, y=6, relative to [seeder.1] object >>
 THEN: [Create New Objects] > *Create Object*
 << Select the [smokey.2] object >>
 << Set the coordinates to x=-3, y=2, relative to [seeder.1] object >>
 THEN: [Create New Objects] > *Create Object*
 << Select the [smokey.3] object >>
 << Set the coordinates to x=-1, y=1, relative to [seeder.1] object >>
 THEN: [Create New Objects] > *Create Object*
 << Select the [smokey.3] object >>
 << Set the coordinates to x=-3, y=4, relative to [seeder.2] object >>
 THEN: [Create New Objects] > *Create Object*
 << Select the [smokey.2] object >>
 << Set the coordinates to x=4, y=1, relative to [seeder.2] object >>

9) Two more events and we'll finish this part of the tutorial... This one destroys all our "smokey" objects on every loop:

IF: [Special Object] > *Always*
 THEN: [smokey.1] > *Destroy*
 THEN: [smokey.2] > *Destroy*
 THEN: [smokey.3] > *Destroy*

9) Here's the last event we'll create in this part of this tutorial... It destroys both of the "seeder" objects when they are no longer needed:

IF: [The Timer Object] > *Is the timer equal to a certain value?*

<< Set the timer to 16 seconds >>

THEN: [seeder.1] > Destroy

THEN: [seeder.2] > Destroy

Always remember – when an object is no longer needed in your frame, it may be wise to destroy it. Think about adding an additional event that'll destroy the given gizmo once its functions are obsolete. This will save up a bit of memory for every useless object not laying around the frame, which can make a difference in bigger games, with lots of effects, explosions, ship fragments flying around, etc. – the less memory your game uses, the better for you and your players.

Aaaaand... We're done here! End of part one!

That's it, cadet! Congratulations! You've made it through part one of this tutorial! You've got your main title screen up and ready, pulsing, rotating and glowing in all its glory! If you wish, play with it a bit, but be sure to save an unmodified copy of your tutorial file somewhere on your hard drive – it will be needed to successfully complete part two of this here tutorial. We'll see each other pretty soon then, huh?

Thanks for your time and see you again soon!

Cheers!

Koobare

*If you have any questions, suggestions or just need help –
mail me at marchewkowy@gmail.com*

END of PART ONE
SEE YOU IN PART TWO!

You have been reading...

HOW TO CREATE
A GREAT LOOKING

MENU

...AND LIVE TO TELL THE TALE.

PART ONE THE TITLE SCREEN

brought to you by
Koobare

Clickteam's funkiest
~~mercenary~~
menu builder



Created for Multimedia Fusion 2 & Multimedia Fusion 2: Developer

Always be sure to have your MMF2 up-to-date!