



DESIGN A SHOOTING STYLE GAME IN FLASH 8

In this tutorial, you will learn how to make a basic arcade style shooting game in Flash 8. An example of the type of game you will create is the game 'Mozzie Blitz' which you can view in the Flash Classroom gallery. This game is designed to build awareness about Dengue mosquitos. These mosquitos are found in Tropical North Queensland and spread Dengue Fever.

The game you will learn to make in this tutorial will have the same features of the Mozzie Blitz game. This includes a title scene, an instructions scene, a scene containing the actual game and a final score scene (see diagrams below).



Fig 1. Title Scene

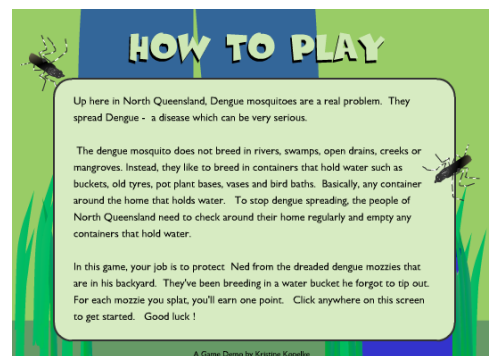


Fig 2. Instructions Scene



Fig 3. Scene involving Game Play



Fig 4. Final Score Scene

To create your game, you will work through the following steps:

1. A planning process to conceptualise your game;
2. The design process where you make your backgrounds and symbols;
3. The authoring process where you add the score, animations and actionscript;
4. The testing process where you test your game.



PLANNING YOUR GAME

It's time for you to start planning the game that you will build by following this tutorial. As this tutorial only covers a certain amount of content, you will need to ensure that the game you are thinking of building initially only has the features listed below. Although, the sky's the limit with what's possible in game development in Flash, this tutorial will only introduce you to game design—so make sure you remember that your game will only have the features below—otherwise you'll be disappointed.

FEATURES OF YOUR GAME

1. A **Title Scene** to introduce the game. The user will click a button on this scene to enter the Instructions Scene. This scene will feature the name of the game, a picture / animation of some sort to engage the user and the name of the author/s.
2. An **Instructions Scene** where you outline how to play the game. Once the user has read the instructions they click on a button to enter the game.
3. A **Game Play Scene** which contains a background, moving objects that must be hit to earn points and a dynamic text box which contains the changing score. Each object that is moving e.g. the mosquito, will need to be on it's own layer. You will be animating these objects using motion tweens.
4. A **Final Score Scene** which contains a copy of the dynamic text box, a message for participants and a replay button that takes the player back to the Title Scene.

Note that once you have completed the game, you may want to enhance it by doing things such as changing the mouse cursor to an object that suits the context of your game e.g. fly swatter, crosshair / and adding sound to your game.

These enhancements aren't covered in this tutorial. To learn how to do these things, complete the Working with Sound in Flash tutorial and the Changing the Mouse Cursor tutorial. Both are available on the Flash Classroom site.

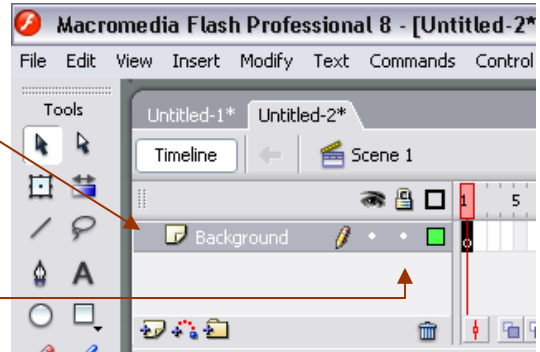
Now that you're aware of the features your game can have, start planning by drawing some storyboards / layouts of each scene on paper. Think of a catchy name for your game and write the instructions for your game. You may even want to do some rough sketches of the characters / objects in your game.

Once you've got all this worked out—you're ready to start creating your game. The following pages will show you step-by-step how to do this.



SETTING UP YOUR TITLE SCENE

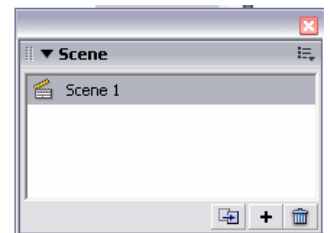
1. Open a new Flash file by selecting **File > New**.
2. Double-click on the text Layer 1 on the first layer of your timeline and type in **Background**.
3. On this layer, **draw** your background picture. Once you are happy with your background, lock this layer by clicking on the dot underneath the lock icon.
4. Make a new layer by clicking on the **Add Layer** button.
5. Rename this layer **Text** by double clicking on the words Layer 2. Add your title and any other text on this layer. **Lock** this layer once you are happy with it.
6. You may want to have a picture of one or more of the games characters / objects on the title page. In the case of my game, 'Mozzie Blitz', I have a picture of one of the mosquitoes. If you do, make a new layer by clicking on the layer button and rename this layer **Characters**. Draw or copy your character onto this layer.



That's nearly everything for the title scene—however, at the moment there isn't any way for the user to get to the next scene—or for Flash to know to stay on the title scene until the user clicks a button. We are going to fix this in the next sections by setting up multiple scenes for the game and adding buttons.

SETTING UP MULTIPLE SCENES

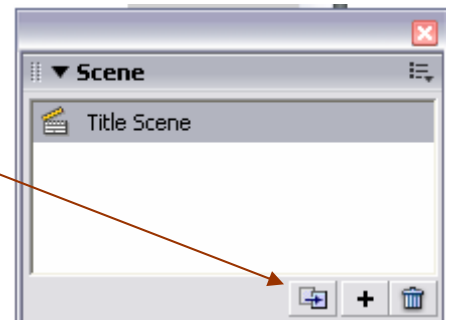
7. To begin with, we're going to open up our Scene panel by holding down **Shift** and **F2**. This will open the panel shown to the right.
8. The first thing we are going to do is change the name of the scene from Scene 1 to **Title Scene**. To do this, double click on the words Scene 1 and type in **Title Scene**.
9. Now let's set up the other scenes for our game. You have a choice at this point based on the design of your game. In my game, 'Mozzie Blitz' I decided to use the same background on each scene. This meant that when I started setting up my scenes, it made sense to duplicate my scenes so I had the background automatically in every scene. In the other scenes, I simply removed, elements from the Title Scene that I didn't need.



However, you may not want to have the same background on every scene and therefore duplicating the first scene doesn't make sense. Take a moment to think which of these options suit you.

9a. Option 1—Duplicate the Title Scene

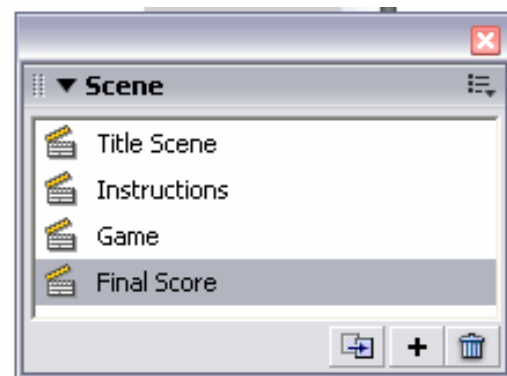
If you've chosen to duplicate the title scene, you can do this by clicking on the title scene in the Scene panel and selecting the **Duplicate Scene** button shown here.



If all of your scenes are to have the same background, repeat this process 3 times.

If not, repeat it until you have the number of scenes you need with that background.

At this point, your Scene Panel should look something like this. Note that I have renamed all the duplicated scenes by double clicking on them and typing in the new names.



I encourage you to use the same names as I have so that you find it easier to follow the actionscript we write later.

If you have set up all your scenes—go to Step 10.

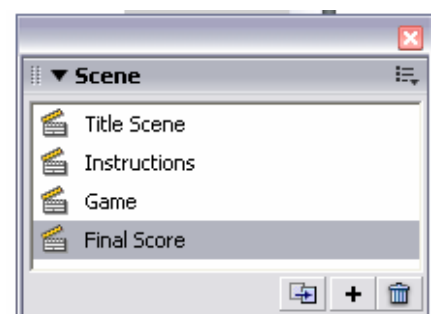
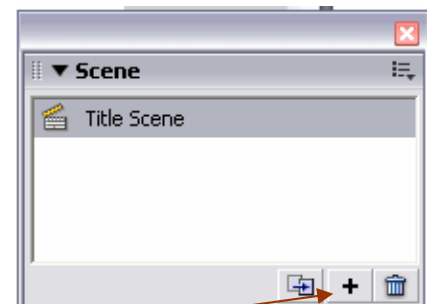
9b. Option 2— Add New Scenes

You may want each scene in your game to have a different look. That's fine too and if you choose to do this, you will now simply add blank scenes that you can design new backgrounds etc for later on.

To **add a new scene**, click on the + button at the bottom of the panel.

Rename the scenes you add by double clicking on their names and typing in your own names. I encourage you to use the same names as I have so that you find it easier to follow the actionscript we write later.

Your scene panel should look like this once you have added all four scenes.





ADDING BUTTONS TO MOVE BETWEEN EACH SCENE

In the game we are creating, we need to add some buttons to enable the player to move between scenes.

In the 'Mozzie Blitz' example, there are 3 buttons which make this possible. These are:

- 2 x Invisible full screen buttons that are located on the top layer named buttons in the **Title Scene** and the **Instructions** scene.
- 1 x Replay button which is located on a layer named buttons in the **Final Score** scene.

This page will walk you through how to create those buttons. The design of the buttons and type of buttons you use is up to you—you may even wish to use a button from the Common Library that Flash provides. To view the buttons in this collection, select **Window > Common Libraries > Buttons**.

The instructions below show you how to make your own buttons.

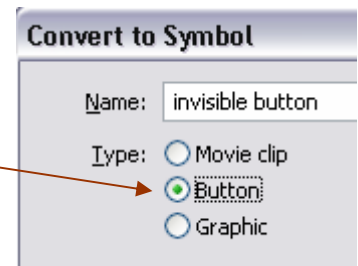
Making Invisible Buttons

10. Make a new layer in the Title Scene and name it Buttons. To do this, click on the **Add New Layer** button at the bottom of the timeline. Double click on the name of the layer and enter **Buttons**.



11. On this new layer which should be at the top of the layers, use the **rectangle drawing tool** to draw a large rectangle over the entire stage. (Note that if your buttons layer isn't at the top, you can click on it and drag it above the other layers).

12. Convert the rectangle to a button by selecting **Modify > Convert to Symbol**. Name your button **Invisible Button** and select the button behaviour.



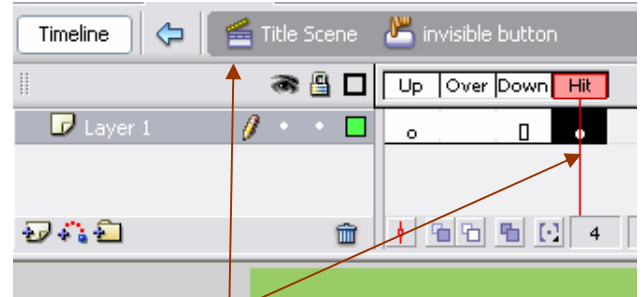
13. To make the button invisible, click on the button and then right click to bring up the context menu. In this menu, select **Edit in Place**.

14. You are now editing the button symbol. You will see that the timeline for this button symbol only has four frames—one for each of the button states.

- The states are
- UP** - when the mouse isn't over the button
 - OVER** - when your mouse is over the button
 - DOWN** - when you click on the mouse
 - HIT** - the active area that is the button.



15. Your button isn't going to have an up, over or down state. It will just have a hit state that contains the rectangle. This will enable the user to simply click their mouse in any location to go to the next scene.



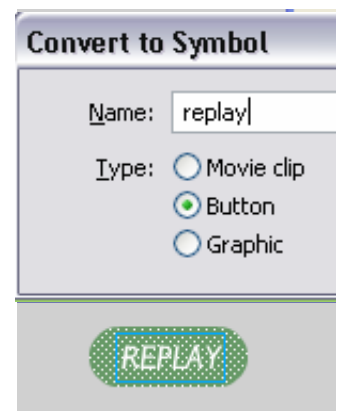
By default, the rectangle will be sitting in the Up state. To move it to the **Hit** state, simply **click** on the keyframe in the Up state and then click again and drag the keyframe into the Hit state. Alternatively, click on the rectangle and cut and paste it into the **Hit** state frame. The timeline within your invisible button symbol should look like the one above.

16. Once you have done this, return to edit the **Title Scene** by selecting the **Title Scene** link just above the timeline.
17. You should now find that your invisible button symbol has turned a transparent aqua colour. This is normal and represents that this button is simply a hotspot where the user will click—it won't actually be visible in the final movie. If you want to edit layers underneath the buttons layer, simply lock this layer and make it invisible by clicking on the dots underneath the lock and eye symbols in the timeline.

You have now made an invisible button. You may wish to use this as I did to link from the Title Scene to the Instructions Scene and from the Instructions scene to the Game Scene. We will look at programming the buttons on the next page. Before we do that, I'll show you how to make a replay button similar to the one in the 'Mozzie Blitz' example.

Making Your Own Button

18. On the grey area to the side of the stage, draw what you want your button to look like when the player does not have the mouse over it. This is the **Up** state.
19. Select your button and select **Modify > Convert to Symbol**. Name your button and select the **Button** behaviour.
20. Click on your button and then right click to bring up the context menu. From this menu select **Edit in Place**.



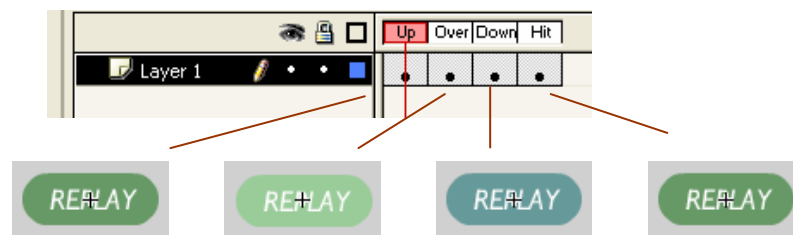


21. You should notice that when editing your button, the timeline will only have a keyframe in the **Up** state. The button will still work when this is the case, however, you may want to change what the button looks like in the other states. For example, it is common for a button to move slightly or change colour when rolled over.

Add **Keyframes** to each of the other button states (**Over**, **Down** and **Hit**) by clicking on each frame and selecting **Insert > Timeline > Keyframe**.

In each keyframe, change what you want the button to look like. The hit state keyframe usually doesn't need to be edited. This state is usually only needed if you want to create an invisible button or say you have an image or text in the other states that is intricate. When this is the case, you may wish to draw a bigger shape over the text or image in this state so that the button is easy for the user to hit. For example, if you didn't do this for a button containing only text, the user would actually have to put their mouse over the lines of the letters to make the button work. Drawing a shape in the hit state makes the button easier to hit.

Below is a diagram that shows the four hit states of the Replay button I created for the 'Mozzie Blitz' game.



22. Once you are happy with your button states, click on the Scene name link at the bottom of the timeline to stop editing the button and return to the current scene.

Programming Your Buttons to Work

It's now time to position the buttons on the button layers in the appropriate scenes and to add the actionscript that will make them go to another scene when clicked. Follow the steps below for each of the buttons to do this.

23. Position the button you want to program in the location you want it on the stage. To make multiple versions of the same button, simply copy and paste or open the Library using **Window > Library** and drag a copy onto the stage.
24. Click on the button you want to add the script to and select **Window > Actions** or press **F9**. This will open the **Actions Panel** where you will type in the script on the following page.



25. Type the following script into the Actions Panel. Note that the scene name has been put in **blue bold** to remind you that you will need to edit this name in the next step.

```
on (release) {  
    gotoAndPlay("Title Scene",1);  
}
```

This script tells Flash that when the player clicks on the button and then releases it, that Flash needs to go to and Play another scene. In this case, the Title Scene. The number 1 indicates that it should go to frame 1 of that scene.

26. Edit the script by changing the name of the scene to the name of the scene your button is going to take the player to. Remember to keep the quotation marks " " around the scene name.

Repeat this process for each of your buttons.

Adding Stop Actions to Scenes

If you were to test your movie now you would find that your movie automatically jumps from one scene to another. This is because Flash naturally plays through scene 1 and then moves to scene 2 and so on, unless it is instructed to stop.

We are going to add **stop actions** to the final keyframe in our **Title Scene**, **Instructions** scene and **Final Score** scene. This will mean that the Flash player will stop at these scenes until the user clicks on the buttons you have added in the previous step. Note that we don't want to add a stop action to the **Game** scene as we want this scene to end by simply jumping to the **Final Score** scene.

27. To add a stop action to the final keyframe of each scene, click on the keyframe and press **F9** to open the **Actions Panel**.

28. Type in the stop action shown below:

```
stop ();
```

This will trap the user in the current scene until they click on the button to go to another scene.

Repeat this process for each scene the user should be stopped at.

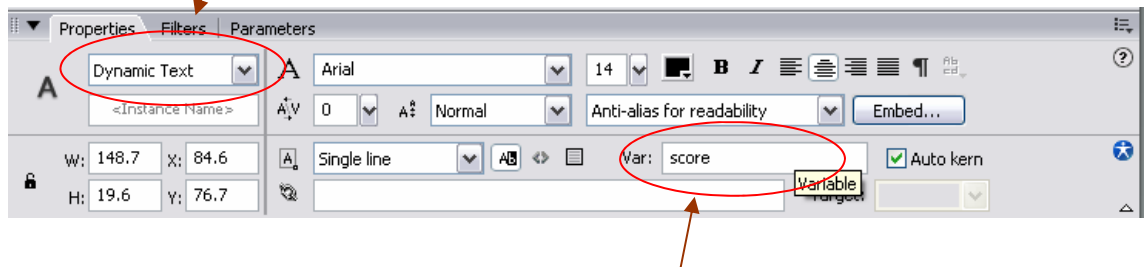
Building Your Game Scene

You have now completed most of the steps involved in setting up the different scenes and the navigation between them. In all of the scenes except for the **Game** scene, you really just need to change the content on the layers to include the text and objects you want.

This section of the tutorial will show you how to create the dynamic text box which will show the score and how to create and animate the objects that when hit will change the value of the score.

Part A— Setting up the Score

29. Add a new layer in the **Game** scene by selecting the **Add New Layer** button at the bottom of the timeline.
30. Rename this layer **Score** by double-clicking on the Layer name and typing in score.
31. **Draw a text box** in the location on the screen where you want your score to appear. In the properties panel, change the font size and colour of the text to make it look how you want it to. Type a zero into this text box.
32. Change the drop down menu shown at the top left corner of the properties panel to the **Dynamic Text** setting (see diagram below).



33. Type the word **score** in the Var (variable) cell located in the bottom right hand corner of the properties panel.

Explanation

What you have just done in the previous two steps is set up a variable for the score that we have called **score**. If you are new to variables, a variable is like a container that contains different values. Our variable or score will change because we will add script to our moving targets (e.g. mozzies) that instruct the Flash player to add a certain amount to the score when clicked.

34. The final scene of the game that you named **Final Score** will also contain the score. At this point, copy this text box and paste it into that scene so it is there for later.
35. You may wish to add a text box above the score variable that contains the word score as I have done. Ensure for this text box you change Dynamic Text back to **Static Text** as this word won't change.



Part B — Making and animating the targets or objects to be hit.

It's now time to make our collection of objects that are going to be hit by the user to earn them points in the game. We are going to put each object that we create on it's own layer and make them move by using motion tweens. If you haven't created a motion tween before, it's a good time to save your game and to complete the motion tween tutorial on this site as this tutorial won't go into motion tweens in detail.

How to do it...

36. In the grey area beside the stage, **draw the object or character** that the user will have to hit to earn points.

Note that if you choose to, you can create more than one object and even objects that can be set to take away points. For example, if you were making a space invaders style game, you could have green ships that are your friends and red ships that are your enemies. In the script that you write, you could make it work that if the user hits a green ship, points are deducted from their score.

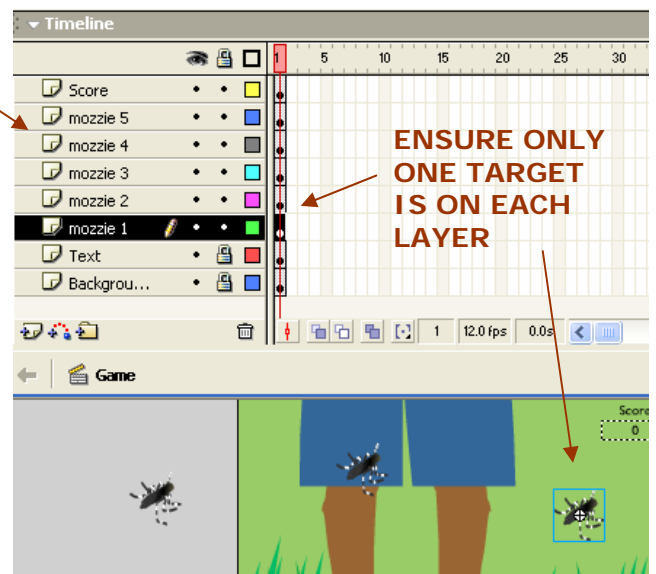
37. Once you are happy with the object or characters you have drawn, convert it/them to button symbols by selecting it/them and choosing **Modify > Convert to Symbol**. Make sure you have the **Button** behaviour checked and give it a name.

38. In an earlier step, you learnt how to **change the button states**. If you wish, you may take the time now to do this for these buttons.

39. Decide on how many of your characters will be moving around the screen and **create a layer** for each of them. You will need a separate layer for every character target that you will be animating. Name each layer—in my game I named my layers Mozzie1, Mozzie2, Mozzie3 etc.

40. Put a **copy** of the button that will be the target on each layer. This button should be the only object on these layers. If there is anything else, Flash will not know what object to tween in the next step.

41. Add the actionscript on the following page to each of the buttons.





Adding the Actionscript to Make the Score Work

42. You will need to add the actionscript below to each target button.

To do this **click on the button** and then select **F9** or **Window > Actions** to open the actions panel.

Type the following script in the panel.

```
on (release) {  
    score = Number((score))+50;  
}
```

This script tells the Flash player that when the user hits and releases the mouse on the button, to add 50 points to the score.

You can change 50 to another value if you choose to. You can also change the + sign to a - sign in the script if points are to be deducted from the score when the object is hit.

43. Check that your script works by selecting **Control > Test Movie**. You should see your score increase each time you hit the button.

Make sure you add this script to every target button in your game.

Animate the targets

In the example, 'Mozzie Blitz', the mozzies are the buttons that the user must hit to increase the score. Each mozzie is a copy of the mozzie button and each one is sitting on it's own layer.

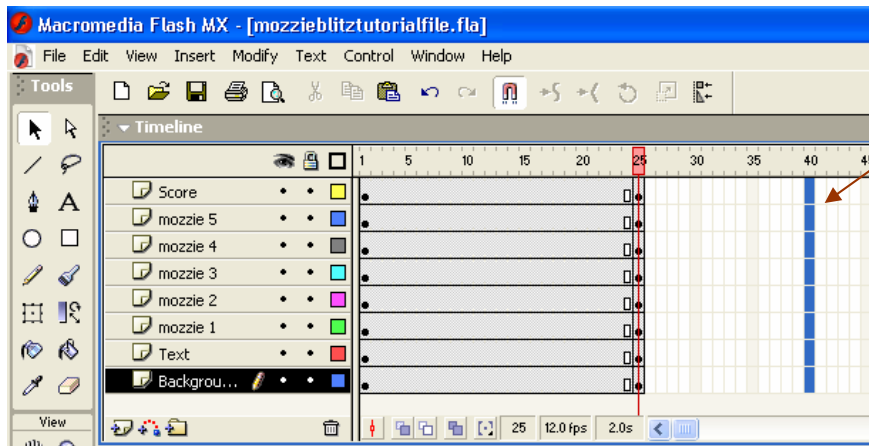
The mozzies enter the stage at different times and fly around. This has been achieved by adding the mozzies at different points in the timeline and creating motion tweens to enable them to fly. In the second version of Mozzie Blitz, the mozzies fly along motion guides. This enables them to move in more interesting ways. However, in this tutorial we'll simply make them move using standard motion tweens. If you know how to use guides, feel free to use them though.

44. Start animating your target buttons by selecting the first one and moving it in **keyframe 1** to the location you want the target to be at the start of the game.
45. Click further along the timeline at around frame 20 and insert a second keyframe by selecting **Insert > Timeline > Keyframe**. Move the target to the location on the stage you want the object to be in before it disappears.
46. Create a **motion tween** by clicking in between these two frames on this layer and changing the tween menu to **Motion** in the Properties Panel.



47. Test that your target object moves from one point to the other. You may notice that the contents of your other layers such as the background disappear when the object is moving. You will need to unlock the other layers and extend them by clicking on them in the final frame of the animation and selecting **Insert > Timeline > Frame**.

Note that you can add frames / keyframes to multiple layers at the same time by clicking on the frame in the highest layer and dragging your mouse down across the layers you want to select. (see figure below)



This figure shows how multiple frames can be selected to convert to keyframes / frames.

49. To make your game challenging, you will need to stagger when your targets come in and out of your game. You can do this by making each target begin and end its motion tween at different points in the timeline.

On each layer, you can drag the keyframe along the timeline to a different starting position. You can also insert keyframes at any frame of the timeline by selecting **Insert > Timeline > Keyframe**. Note that if you end up with unwanted frames, you can scroll across them, right click your mouse and select **Remove frames**.

(If these steps are confusing you, you probably need to take a break from this tutorial and work through the Create a Basic Motion Tween tutorial on this site).

50. Use motion tweens to animate all of your targets. Once you have done this, test your game by selecting **Control > Test Movie**.

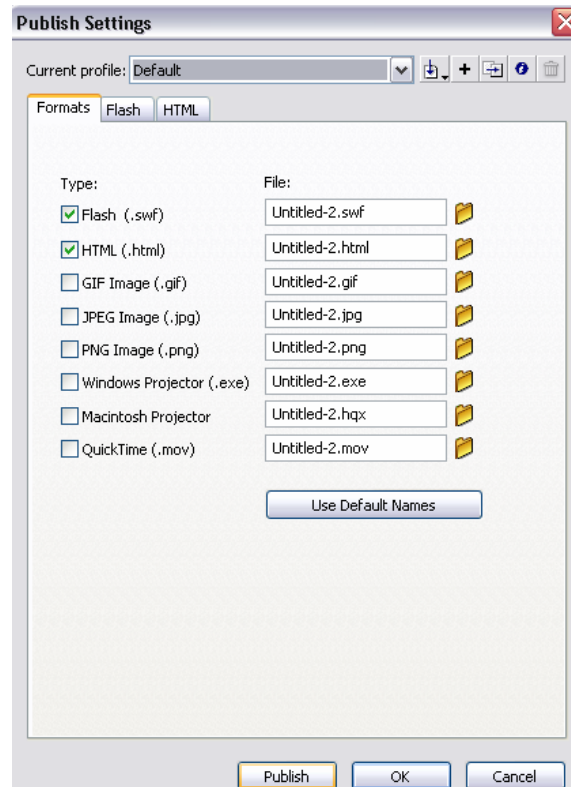
If it works, congratulations—it's time to save and publish your work. If not, go back and check through the steps to ensure you haven't missed anything.

Check out the next page to learn how to Publish your work in different file formats—and to find out how you can make enhancements to your game.



PUBLISH & SHARE YOUR WORK

51. Save your work by selecting **File > Save**.
52. Turn your flash file into a game that can be played on any computer by publishing it in different file formats. To do this select **File > Publish Settings**. The box shown below will appear.



53. Tick the file formats you want and click on the **Publish** button. These files will be saved in the same location you saved your original file. If you want your game to be a standalone file that can be played on Windows or Macintosh machines—ensure you check the Windows Projector (.exe) and Macintosh Projector (.exe) format options.

ENHANCE YOUR GAME

If you've got to this point, why not learn more about game design and how to enhance your game with a customized mouse or sound by completing the following Flash Classroom Tutorials

Working with Sound

Changing the Mouse Cursor tutorial

To learn how to create more complex shooting style games, check out some of the links provided in the Online Links section of our Flash Classrooms site.