Jump Man!

Jump Man is a simple game where the player controls ‘Jump Man’ who must jump over a series of crates as he skateboards along a road.

The game is controlled by pressing a key on the keyboard. Jump Man skates automatically towards the crates and the player must time the jumps to avoid the crates. Hitting the crates decreases Jump Man’s health. Points are awarded for the amount of crates that are avoided. The game ends when Jump Man’s health is depleted.

Open Flash and Start a new Flash File (Actionscript 3).

In the Timeline, double click on the name of Layer 1 and rename it to ground.

Draw a green coloured rectangle that covers the bottom half of the stage. This will represent the ground.

Add a new layer and name it jumpman.

Draw a character and his skateboard on this layer. Make its size and position similar to that shown in the screenshot below.
Add a new layer and name it *crates*.

Draw a crate on this layer. Make its size and position similar to that shown in the screenshot below.

![Screenshot of a game with a character and crates]

Select Jump Man, then right-click on it and choose *Convert to Symbol*.

Select Movie Clip type and set the name to *jumpman*. Press OK.

![Convert to Symbol dialog]

Select the crate, then right-click on it and choose *Convert to Symbol*. Name it *crate* and set it to a Movie Clip type. Press OK.

With the crate still selected, look at the properties panel, which will be either at the right or bottom of the screen. If you can’t see the Properties Panel then go to the menu bar and choose *Window – Properties – Properties*.
In the properties panel change the Instance name to \textit{crate\_mc}.

Select jump Man and set his Instance Name in the Properties Panel to \textit{jumpman\_mc}.

Any object that is controlled by Actionscript must have its own instance name. Every object must have its own unique instance name – duplicate names will confuse Flash.

Add a new layer and name it \textit{code}.

Select frame 1 of the code layer and press F9 to open the Actions Panel (or go to the menu bar and choose \textit{Window – Actions}).
Make sure Script Assist is switched OFF (see diagram).

The Actions Panel is where we type the Actionscript code that will make our game work.

Making the crate move

First we will make the crate move. This is done by repeatedly changing the crate’s position on the x axis. The crate will move from right to left, so will be moving in a negative direction. This must happen continuously.

To make something happen continuously in Actionscript we must set up a thing called an Event Listener that will cause a function to run continuously. Type the following line of code, which will create an Event Listener which is attached to the stage. This event listener is a special one called an Enter Frame event listener, which runs continuously. This type of event listener is used a lot in games programming. This event listener will run a function called gameloop, which you will make in a later step

```ActionScript
stage.addEventListener(Event.ENTER_FRAME, gameloop);
```

Next you need to create the function called gameloop. Type the following line of code – note the curly brackets (also known as braces) which indicate the start and end points of the function.

```ActionScript
function gameloop(e:Event):void{
}
```
Any code that is placed within the curly brackets will become part of this function. Add the following line of code between the curly brackets. This line changes the position of the crate on the x axis by 20 pixels (moving in a negative direction, to the left).

\[
\text{crate\_mc.x-=20;}
\]

Close the Actions Panel, then test your Flash movie by pressing Ctrl & Enter at the same time or go to the menu bar and choose Control – Test Movie.

You should see the crate moving from right to left and going off the edge of the screen.

**Making another crate**

You could make another crate and make it move in the same way as the previous one, but we will cheat instead and reuse the crate we already made (reusing objects is common practice in games programming as it makes the programs smaller and faster).

To do this, we will wait until the existing crate moves off the left side of the screen and then we’ll magically move it back to the right of the screen. This way we reuse the same crate over and over, but it will just look like another crate moving across the screen. The player will be unaware that we’ve done this.

Add some extra code to the function, so that it looks like this:

\[
\text{function gameloop(e:Event):void{}
\]

\[
\begin{align*}
\text{    crate\_mc.x=&-20;} \\
\text{    if (crate\_mc.x<-100){} &} \\
\text{        crate\_mc.x=650; &}
\end{align*}
\]

\[
\text{}}}
\]

This code works by checking to see if the crate’s position on the x axis is less than -100 (a position off the left side of the screen). If the crate’s position is less than -100, then the crate is immediately placed at position 650 on the x axis – meaning that the crate is off the right hand side of the screen.

Close the Actions Panel, then test your Flash movie by pressing Ctrl & Enter at the same time or go to the menu bar and choose Control – Test Movie.

You should see a continuous stream of crates move in from the right and go off the left of the screen.

In the next part of this lesson you will learn to make Jump Man jump.

**Exercise**

Think about how you could make the crate move faster or slower. Which part of the code affects the speed of the crate?