



If...then...else and switch:

How an application makes a decision

Applications often need to make decisions. This tutorial introduces you to using 'if...then' to run different scripts depending on the outcome of a decision. We go on to take a look at using switch to choose from a list of possible decisions.

Key topics covered in this tutorial

- Making choices using 'if...then...else'
- Making more complex choices using 'switch...case'
- Making a random choice
- Using comments in scripts

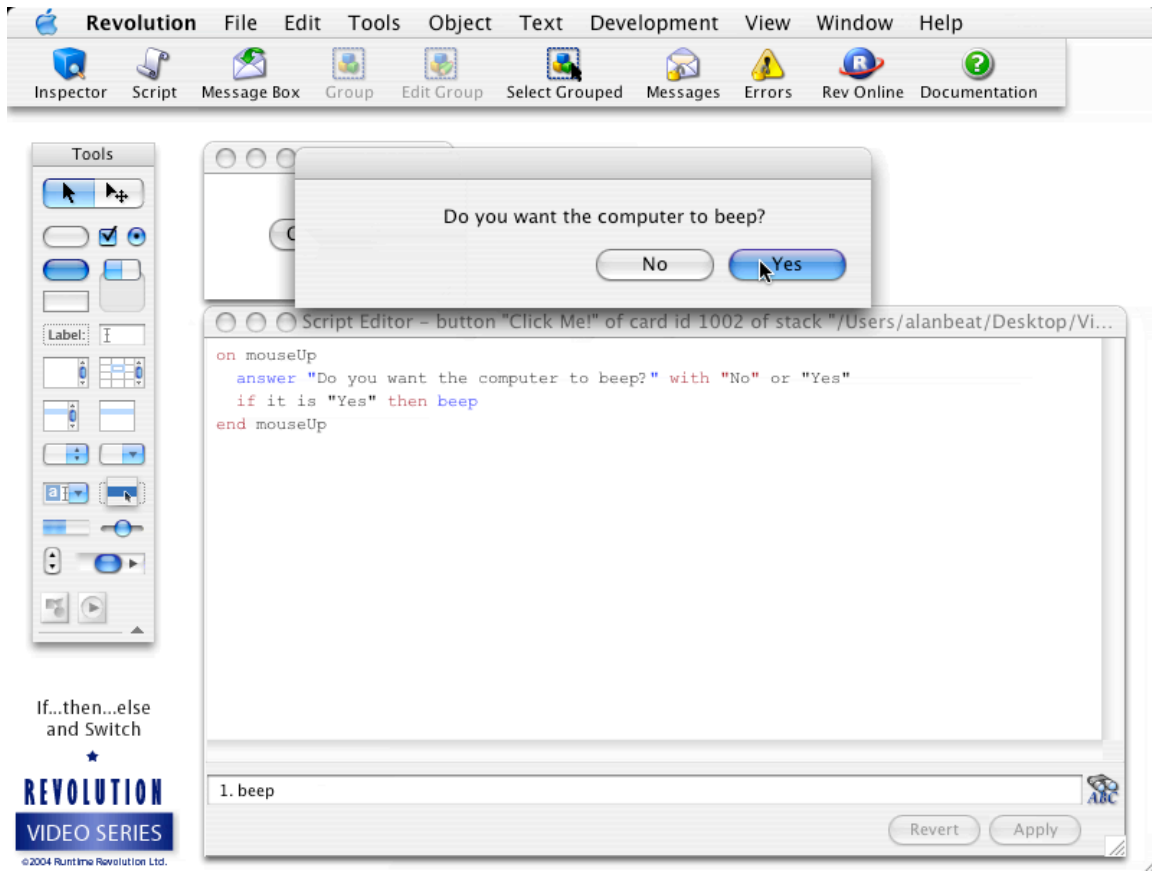
Your application will frequently need to make decisions. Often it will need to run different lines of script depending on what a user does, and sometimes it will need to make decisions when computing something, processing data, or working with text.

Let's try making a script containing a simple decision.

```
answer "Do you want the computer to beep?" with "No" or "Yes"
```

Your application has to make a decision to beep or not, depending on whether the user answers 'No' or 'Yes'.

```
If it is "Yes" then beep
```



And if we try it again and choose 'No' nothing happens.

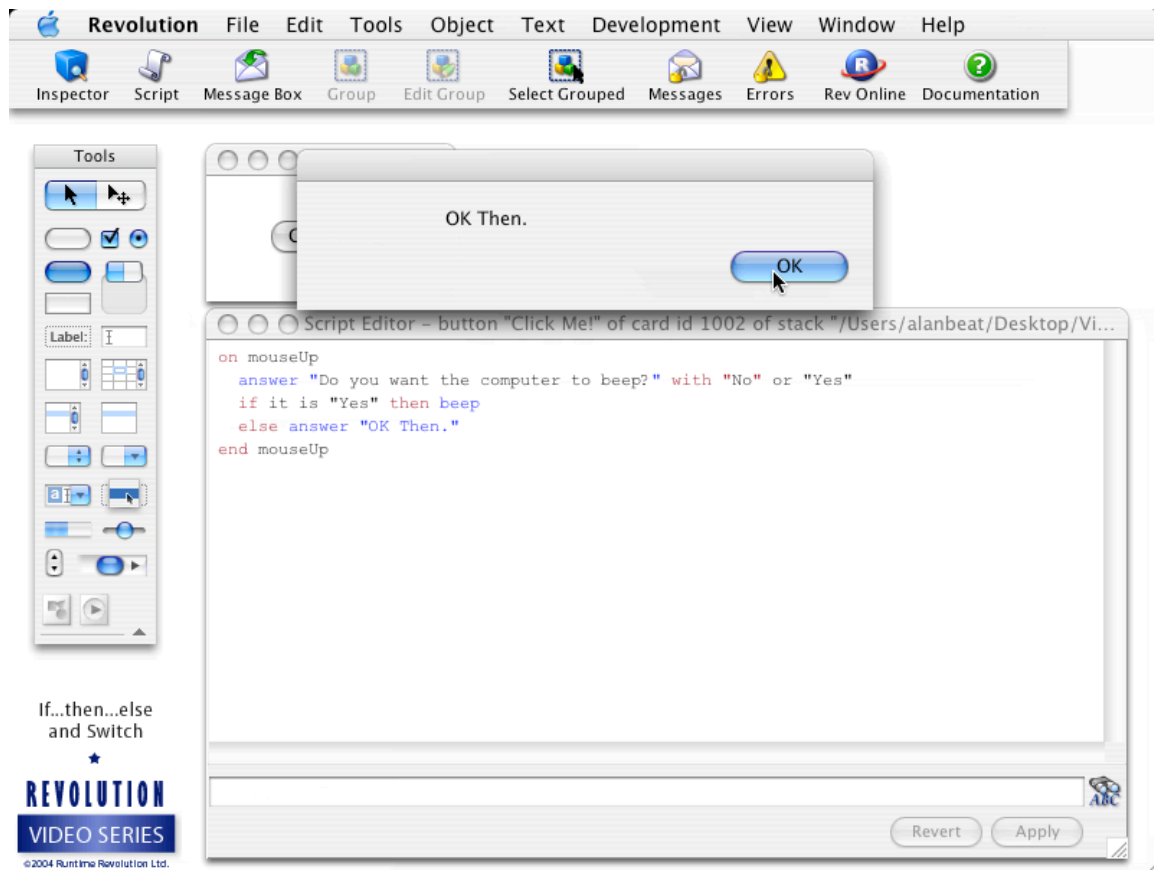
Let's provide an alternative action for if the user has clicked 'No'. We'll add

```
else answer "OK Then."
```

The screenshot displays the Revolution IDE interface. At the top is a menu bar with 'Revolution', 'File', 'Edit', 'Tools', 'Object', 'Text', 'Development', 'View', 'Window', and 'Help'. Below the menu bar is a toolbar with icons for Inspector, Script, Message Box, Group, Edit Group, Select Grouped, Messages, Errors, Rev Online, and Documentation. On the left is a 'Tools' palette with various icons for creating and editing objects. The main workspace shows a 'Script Editor' window titled 'Script Editor - button "Click Me!" of card id 1002 of stack "/Users/alanbeat/Desktop/Vi...'. The script editor contains the following code:

```
on mouseUp
  answer "Do you want the computer to beep?" with "No" or "Yes"
  if it is "Yes" then beep
  else answer "OK Then."
end mouseUp
```

Overlaid on the script editor is a dialog box titled 'Do you want the computer to beep?' with two buttons: 'No' and 'Yes'. The 'No' button is highlighted with a mouse cursor. At the bottom of the script editor window are 'Revert' and 'Apply' buttons. In the bottom left corner, there is a logo for 'REVOLUTION VIDEO SERIES' and the text '© 2004 Runtime Revolution Ltd.'.



And there we have the alternative action. The script uses the 'if then' and 'else' statements to make a decision about what to do next.

Now let's suppose that we have a decision involving three possible choices. How can we choose between them? Let's edit the answer dialog to provide another answer 'Perhaps':

```
answer "Do you want the computer to beep?" with "No" or "Yes" \
  or "Perhaps"
```

(Note that here this script is too long to fit on one line without wrapping so we've used the Transcript line continuation character '\' to separate it into two lines. Revolution will still treat it as a single line of script when it executes.)

Instead of using 'else' we'll use 'else if'.

```
else if it is "No" then answer "OK Then."
```

Now we can add another else statement below for the third possible value.

```
else if it is "Perhaps"
```

How do we want the computer to respond if the user answers with 'Perhaps'? Well, to provide a random outcome you use the random function.

```
else if it is "Perhaps" and the random of 2 is 1 then beep
```

So what you are saying is that if the user has clicked the 'Perhaps' button in the dialog, and a random choice between one and two turns out to be one, the computer should beep.

What happens if you want to run several lines of script as a result of each possible option? You need to change the format of the script very slightly to allow us to write multiple lines for each option. In this format, you also need to tell Revolution that the possible choices have ended. You do this by adding `end if`.

```
else if it is "Perhaps" and the random of 2 is 1 then  
    beep  
    --another command  
end if
```

Any script placed after the 'end if' will be run regardless of the choice made by the 'if' structure above.

If you have a list of options, you can use the 'switch' control structure instead of using 'if...then...else'. Let's have a look at how we could use switch to make these choices instead of the script we have. We'll start by writing

```
switch it
```

to tell Revolution that we want to compare the it variable against a list of possible choices. Then we'll add

```
case "Yes"
```

then enter the lines of script here. We could put additional lines of script here. When we've finished with the script for the "Yes" option, we'll put in an

```
exit switch
```

statement. Then we'll add the other options:

```
case "No"
```

```
    answer "OK Then."
```

```
    exit switch
```

```
case "Perhaps"
```

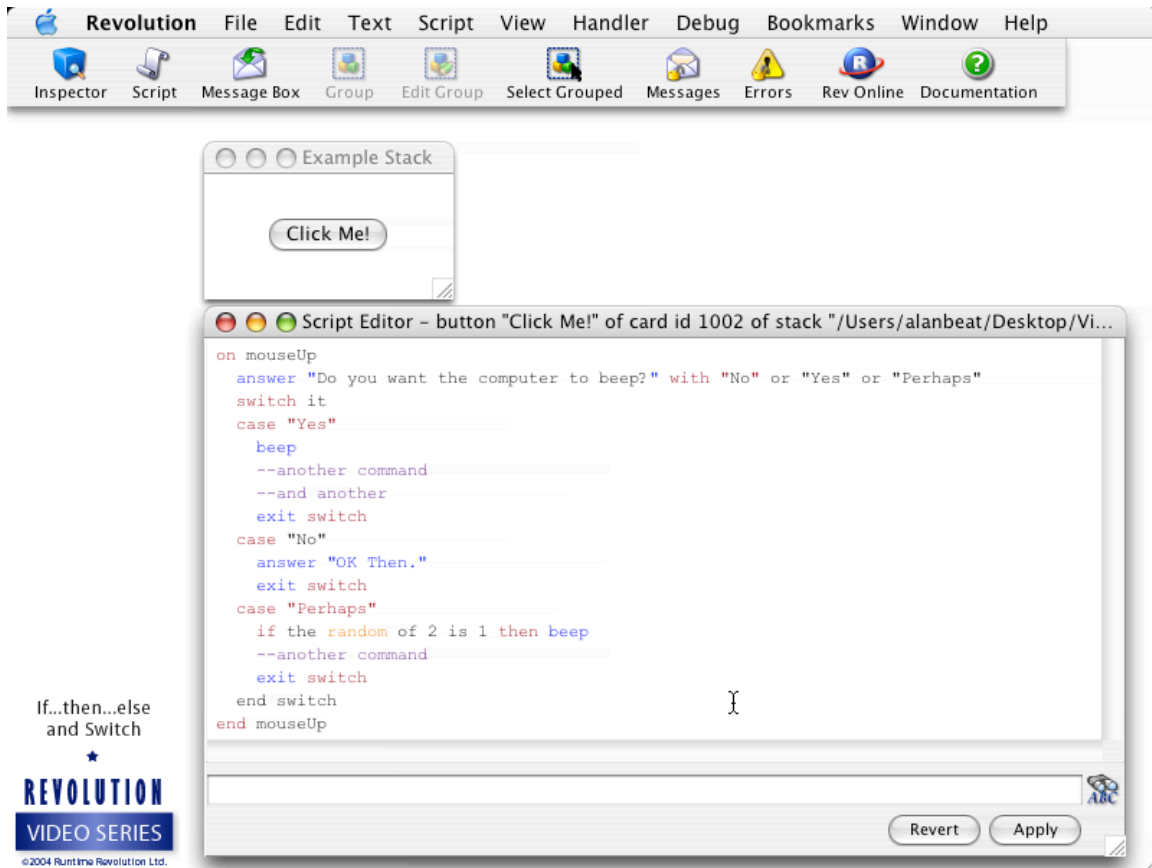
```
    if the random of 2 is 1 then beep
```

```
    exit switch
```

and we end the switch with

```
end switch
```

which is similar to 'end if' in that it means that the script after this statement will be run regardless of what choice was picked above.



If you look at the sample scripts in the appendix, you will see that we have used text after a pair of dashes to place a comment about what the script does. You can place a comment or note at the end of any line of script or on any blank line by placing two dashes in front of the comment. Commented text is ignored by Transcript. However, it is very useful in helping you to understand what is going on. We recommend you get into the habit of regularly writing comments in your scripts. By writing comments you make it easier to come back and edit the application later, even when you haven't worked on it for several months and can't remember how all of it works.

Appendix: Scripts used in this tutorial

```
on mouseUp
    answer "Do you want the computer to beep?" with "No" or "Yes"
    if it is "Yes" then beep
end mouseUp
```

```
on mouseUp
    answer "Do you want the computer to beep?" with "No" or "Yes"
    if it is "Yes" then beep
    else answer "OK Then."
end mouseUp
```

```
on mouseUp
    answer "Do you want the computer to beep?" with "No" or "Yes" \
    or "Perhaps"
    if it is "Yes" then beep
    else if it is "No" then answer "OK Then."
    else if it is "Perhaps" and the random of 2 is 1 then beep
end mouseUp
```



```
on mouseUp
    answer "Do you want the computer to beep?" with "No" or "Yes" \
        or "Perhaps"
    if it is "Yes" then
        beep
        --you can insert additional script lines here for "Yes"
    else if it is "No" then
        answer "OK Then."
        --you can insert additional script lines here for "No"
    else if it is "Perhaps" and the random of 2 is 1 then
        beep
        --you can insert additional script lines here for "Perhaps"
    end if
end mouseUp
```

```
on mouseUp
    answer "Do you want the computer to beep?" with "No" or "Yes" \
        or "Perhaps"
    switch it
        case "Yes"
            beep
            --you can insert additional script lines here for "Yes"
        exit switch
        case "No"
            answer "OK Then."
            --you can insert additional script lines here for "No"
        exit switch
        case "Perhaps"
            if the random of 2 is 1 then beep
            --you can insert additional script lines here for "Perhaps"
        exit switch
    end switch
end mouseUp
```