# Script 9

Strategy A is now our front runner so we’ll aim to leapfrog it by making Strategy B take account of the dealer’s face card.

We start, again, by making them same, following the same process as before, in reverse.

Delete strategy B

Copy strategy A and paste the copy below

Then rename the copy to strategy B, and all is good

Now insert a new if instruction above the previous one we added

We need to test whether the dealers face card is a 5 – and we have been passed the dealerFaceUp parameter so let’s try writing dealerFaceUp is 5

But we are almost immediately told that we have an error, because dealerFaceUp is of type Card and we are trying to compare it to an integer – 5

But if we delete back and put a dot after dealerFaceUp then we can see that it has a property named value, which probably contains an integer, and choosing that we can compare it to 5 successfully.

When that condition is true we want to set the drawLimit very conservatively, and we will choose 12 because that is the first value where drawing another card could cause you to go bust.

Now the only problem with this is that immediately afterwards we are testing for a softace and potentially setting the draw limit back up to 18

We only want to do that second check if dealerfaceup value is **not** 5. In programming terminology this is called an ‘else’ clause. So under here, insert an ‘else’.

Now we can cut and paste the whole next if instruction under the else, but still within the first if, but there is a neater solution.

Instead of just else, we can change it to read ‘else if’ and now fill in the fields copying from the other if.

We could use cut and paste, but in this instance it is just as quick to re-type.

And then delete the old one.

Note that we don’t do this with the next if instruction, because that needs to be executed no matter what has happened above it.

So, reading through the code..