

A continuación se muestran los pasos a seguir para construir el ejercicio:

1. Abra Descartes y fije el espacio existente. Guarde la escena con el nombre *Algoritmos\_Evento\_Escena.html*.
2. Añada un gráfico tipo imagen y como imagen use el archivo *1.png* que se encuentra en *Ejercicios/Algoritmos\_Evento/images* dentro del archivo *DocumentacionDescartesJS.zip*. También puede encontrar este archivo con [este vínculo](#). Coloque la imagen en una carpeta *images* al mismo nivel que el interactivo html y haga referencia a la misma en el gráfico tipo imagen recién agregado.
3. Asocie la imagen a las coordenadas  $(x,y)$ .  $x$  y  $y$  son variables que se definirán adelante.
4. Agregue un evento y especifique que se ejecute una sola vez cuando el mouse se encuentra oprimido sobre el espacio *E1*. Haga que la acción del evento sea calcular y el parámetro a calcular sea asignar a la variable  $x$  la coordenada horizontal del mouse y a la variable  $y$  la coordenada vertical. Aplique los cambios y presione y arrastre el mouse sobre el espacio.
5. Cambie el modo de ejecución del evento a *alternar*. Aplique los cambios y oprima el mouse y arrástrelo varias veces.
6. Cambie el modo de ejecución del evento a *siempre*. Aplique los cambios y oprima el mouse y arrástrelo varias veces.

A continuación se presentan los resultados esperados de cada paso del ejercicio, así como algunas observaciones.

1. Es preciso marcar el checkbox *fijo* en el espacio *E1* del selector *Espacio*.  
Tras aplicar los cambios, ya no es posible arrastrar el espacio. La escena queda guardada con el nombre indicado.
2. La imagen se agrega en el selector *Gráficos*. El archivo de la imagen queda guardado en una carpeta *images* a la altura del interactivo, por lo que en el parámetro *archivo* del gráfico *imagen* debe introducirse *images/1.png*.  
Tras aplicar los cambios, la imagen de la manzana aparece en el interactivo.
3. El parámetro *expresión* de la imagen debe cambiarse a  $(x,y)$ .  
Tras aplicar los cambios, la imagen sigue apareciendo en el origen, dado que  $x$  y  $y$  siguen ambas valiendo cero.
4. El evento se agrega en el selector *Programa*. El parámetro *condición* del evento debe decir `E1.mouse_pressed==1`  
para indicar que se ejecutará el evento cuando el mouse se encuentra oprimido sobre el espacio *E1*. Las asignaciones en parámetro deben ser:  
`x=E1.mouse_x`  
`y=E1.mouse_y`  
El menú *ejecución* del evento debe estar colocado en la opción *una sola vez*.  
Tras aplicar los cambios, la imagen aparece aún en el origen. Cuando se presiona el mouse, la imagen aparece donde está el mouse en ese momento, pero al arrastrarlo mientras está oprimido, la imagen no se mueve con él. Si el mouse se suelta y se vuelve a oprimir, la imagen se mantiene en su posición. Ello responde a que el evento se ejecuta una sola vez (la primera vez que se cumple la condición, que es la primera vez que se oprime el mouse). Después de eso el evento ya no se ejecuta.
5. El menú *ejecución* del evento se debe colocar en *alternar*.  
Tras aplicar los cambios, la imagen aparece en el origen. Al oprimir el mouse, la imagen se recoloca a donde está el mouse al momento de presionarlo, pero si se arrastra mientras está oprimido, la imagen se queda a donde se movió originalmente. Si se deja de oprimir el mouse y se vuelve a oprimir, la imagen nuevamente se recoloca, pero nuevamente no sigue al mouse si éste se desplaza mientras está oprimido. Esto se debe a que el modo de ejecución del evento está en *alternar*, por lo que sólo se lanza el evento cada vez que su condición pasa de ser falsa

- (el botón del mouse no está apretado) a verdadera (el botón del mouse está apretado).
6. Tras cambiar el menú de *ejecución* a *siempre* y aplicar los cambios, la imagen aparece de inicio en el origen. Si se oprime el mouse, la imagen inmediatamente aparece en la posición del mouse y, si se arrastra, la imagen sigue al mouse mientras esté apretado, debido a que la acción se hace siempre, y no sólo cuando la presión del botón del mouse cambia de falso a verdadero.