

**Aplicaciones telemáticas**  
**Examen de teoría. 20 de junio de 2019**  
Grado en ingeniería telemática. Universidad Rey Juan Carlos

---

### Instrucciones:

- Entra en el puesto del laboratorio con el nombre de usuario *examen* y la contraseña *examen*.
- Ejecuta el script *prepara* y contesta sus preguntas.
- Esto dejará en el ordenador el fichero `~/at/TULOGIN/teoria.txt`, donde debes escribir tus respuestas. TULOGIN será tu nombre de usuario en el laboratorio.

### Ejercicio 1 (3.3 puntos)

Explica qué hace el siguiente código jQuery

```
$("#j23").on('change keyup paste mouseup', function() {  
    blablabla();  
});
```

### Respuesta

- El signo de dólar es un alias de la función jQuery. Todo el código en jQuery comienza por una llamada a esta única función, que a su vez recibe otras funciones como argumentos.
- ("`#j23`") es el selector. En la sintaxis de jQuery, indica sobre qué elementos se realizará la acción. La almohadilla significa *identificador*, en este caso el identificador *j23*. Por tanto estamos seleccionando el elemento HTML con identificador *j23*.
- `on('change keyup paste mouseup', function() { ...})` es un método que asigna un manejador a una serie de eventos. El método *on* recibe:
  - Un primer argumento que es una cadena que contiene una lista de nombres de eventos, separados por espacios.
  - Un segundo argumento con la función a ejecutar cuando el elemento seleccionado reciba esos eventos.

Los eventos son: cambio en el elemento, pulsar una tecla, pegar desde el portapapeles y soltar el botón del ratón tras haberlo pulsado.

Por tanto, estas líneas hacen que se dispare la función cuyo código es *blablabla()* cuando el elemento html con indentificador *j23* cambia, se pulsa sobre él, se pega algo o se hace clic sobre él.

## Ejercicio 2 (3.3 puntos)

Indica si los siguientes selectores CSS son correctos o no. Si son correctos, indica su significado. Si no son correctos, explica brevemente por qué.

- 1) `.oferta.precio`
- 2) `oferta precio`
- 3) `#oferta#precio`
- 4) `oferta #precio`

### Respuesta

- 1) Correcto. Selecciona los elementos pertenecientes simultáneamente a las clases *oferta* y *precio*
- 2) Incorrecto. Sería un elemento HTML con etiqueta *precio* descendiente de un elemento HTML de etiqueta *oferta*. Pero en HTML no existen estas etiquetas.
- 3) Incorrecto. Sería un elemento HTML con el identificador *oferta* y simultáneamente con el indicador *precio*. Pero en HTML un elemento no puede tener más de un identificador.
- 4) Incorrecto. Seleccionaría el elemento de identificador *precio*, descendiente de un elemento HTML con etiqueta *oferta*. Pero esta etiqueta no existe en HTML.

## Ejercicio 3 (3.4 puntos)

¿Por qué JavaScript 5 no se pueden definir clases? ¿Por qué JavaScript 6 sí se puede?

### Respuesta

Desde su origen, JavaScript usa el paradigma de programación orientada a objetos basada en prototipos, donde no se definen clases sino que los objetos se construyen a partir de otros objetos.

Esto es así hasta la versión 5, pero la versión actual de JavaScript, la 6, (del año 2015) introduce muchos cambios. Uno de ellos es que, manteniendo la programación orientada a prototipos, añade una sintaxis que permite una programación orientada a objetos tradicional, basada en clases.