ETSIT UPM

dit UPM

Desarrollo de Apps para iOS

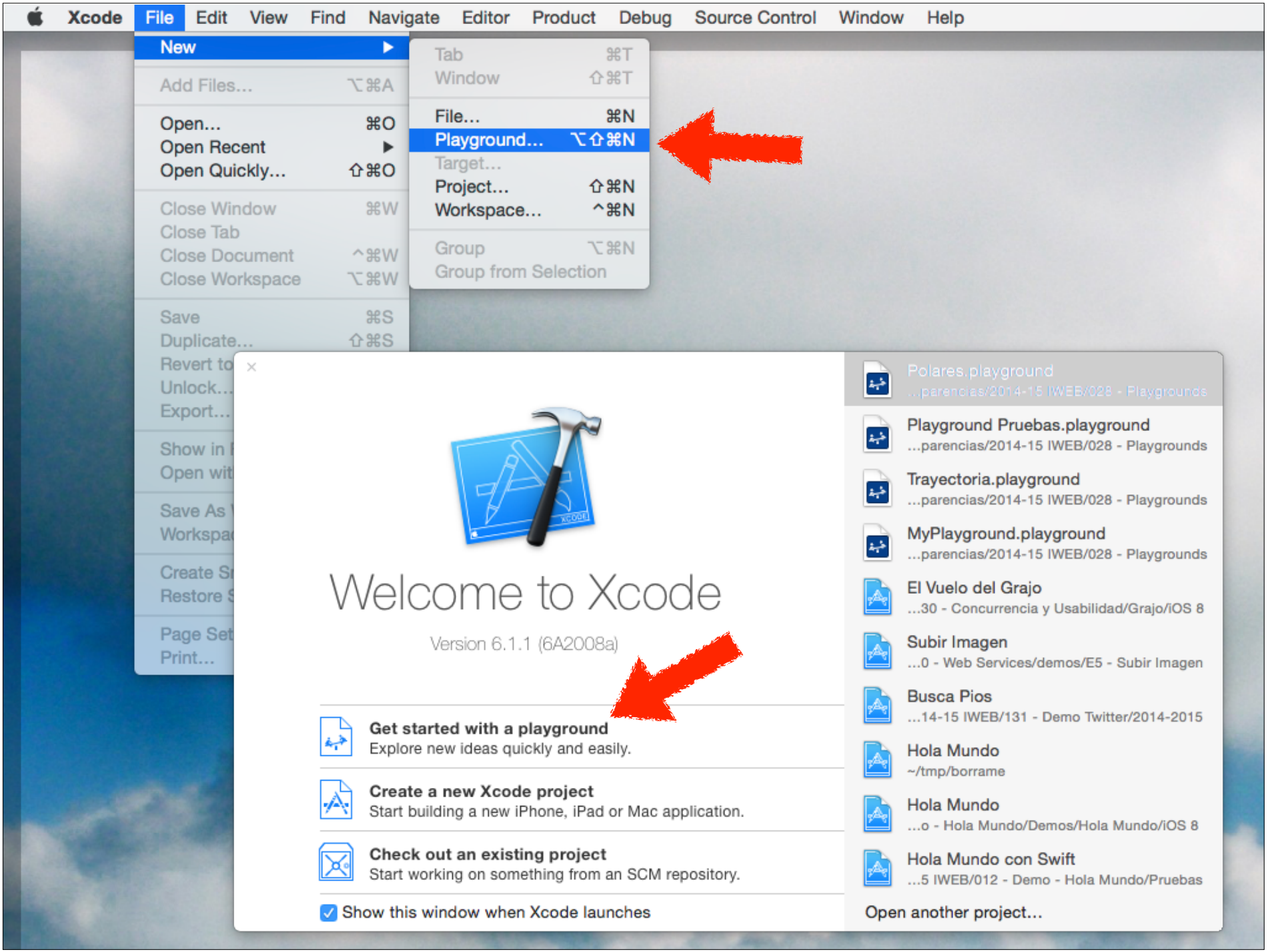
Playgrounds

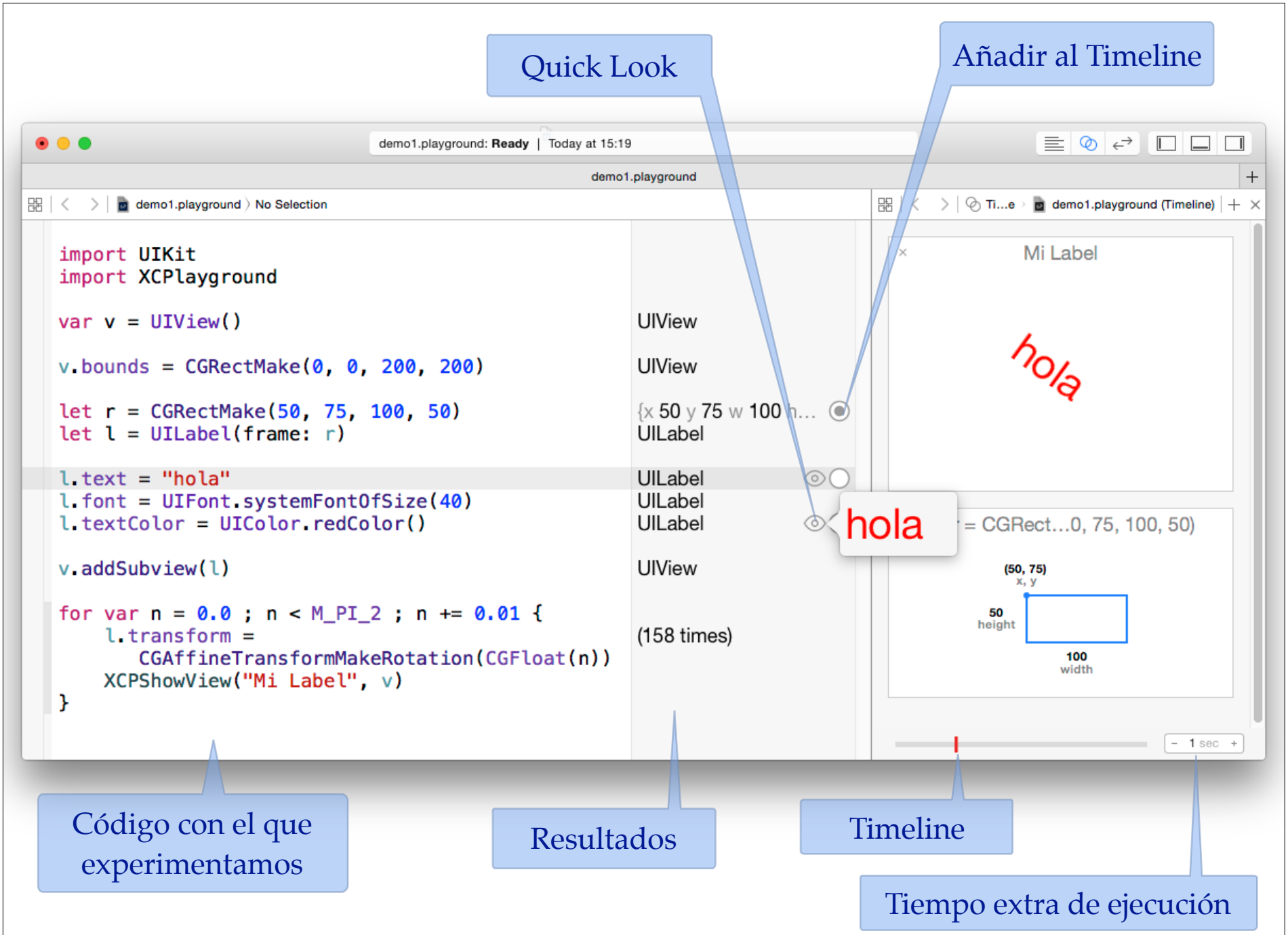
IWEB-LSWC 2014-2015
Santiago Pavón

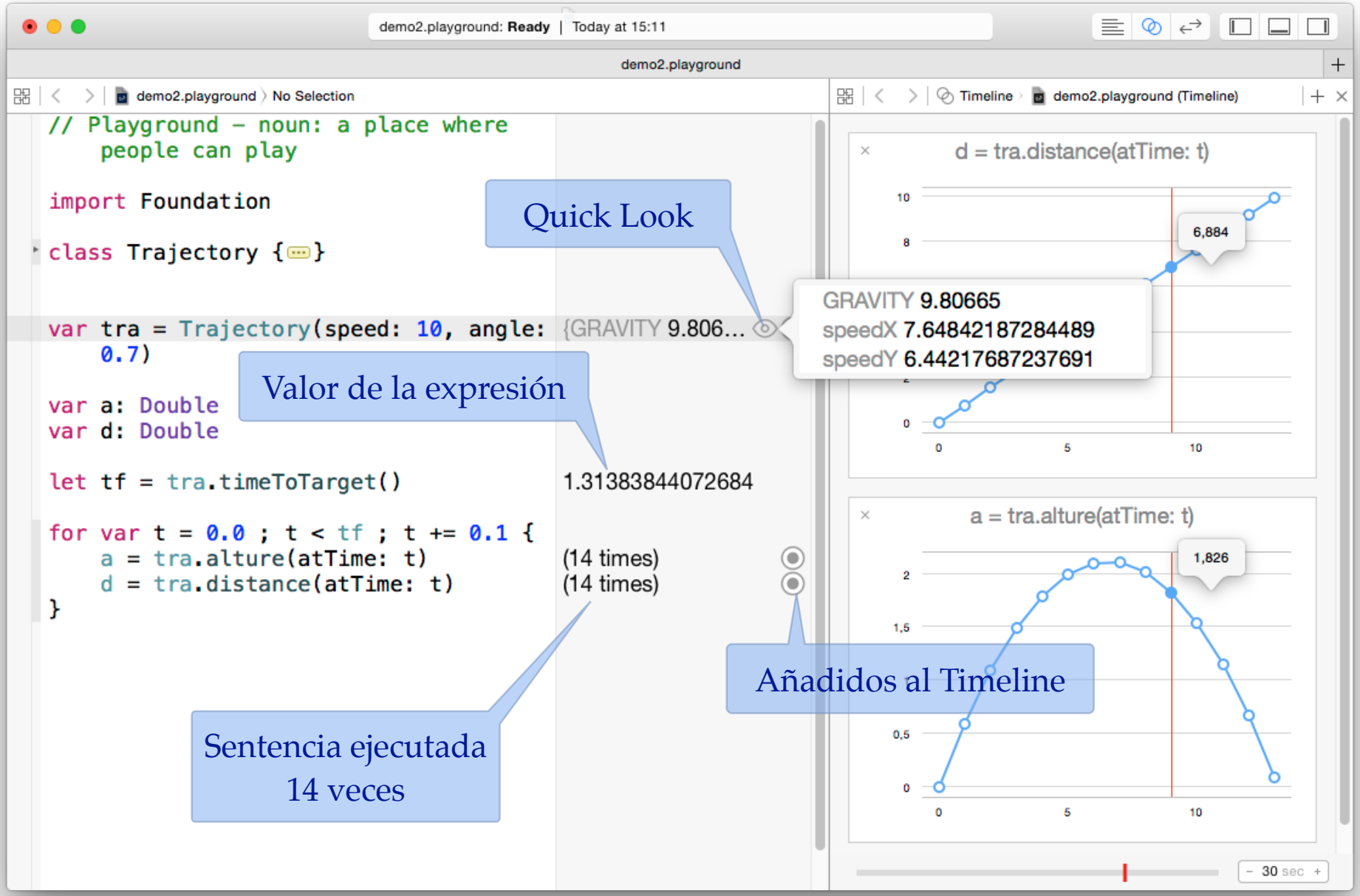
ver: 2015.02.18

¿Qué son los Playgrounds?

- Es un nuevo tipo de documento introducido en Xcode 6
  - Extensión de los ficheros: **`.playground`**
- Se usa para experimentar y probar fragmentos de código.
  - El código Swift se introduce y se ejecuta cada vez que se hace un cambio.
  - Muestra el resultado de ejecutar las expresiones existentes.
  - Se puede acceder a los recursos del proyecto o del sistema (por ejemplo: imágenes)
  - Tiene un timeline para visualizar el estado de la ejecución en cualquier instante de tiempo.


Xcode
File
Edit
View
Find
Navigate
Editor
Product
Debug
Source Control
Window
Help
New
Add Files...
Open...
Open Recent
Open Quickly...
Close Window
Close Tab
Close Document
Close Workspace
Save
Duplicate...
Revert to
Unlock...
Export...
Show in
Open wit
Save As
Workspa
Create S
Restore
Page Set
Print...
Tab
Window
File...
Playground...
Target...
Project...
Workspace...
Group
Group from Selection
Welcome to Xcode
Version 6.1.1 (6A2008a)
Get started with a playground
Explore new ideas quickly and easily.
Create a new Xcode project
Start building a new iPhone, iPad or Mac application.
Check out an existing project
Start working on something from an SCM repository.
Show this window when Xcode launches
Polares.playground
Playground Pruebas.playground
...parencias/2014-15 IWEB/028 - Playgrounds
Trayectoria.playground
...parencias/2014-15 IWEB/028 - Playgrounds
MyPlayground.playground
...parencias/2014-15 IWEB/028 - Playgrounds
El Vuelo del Grajo
...30 - Concurrencia y Usabilidad/Grajo/iOS 8
Subir Imagen
...0 - Web Services/demos/E5 - Subir Imagen
Busca Pios
...14-15 IWEB/131 - Demo Twitter/2014-2015
Hola Mundo
~/tmp/borrame
Hola Mundo
...o - Hola Mundo/Demos/Hola Mundo/iOS 8
Hola Mundo con Swift
...5 IWEB/012 - Demo - Hola Mundo/Pruebas
Open another project...

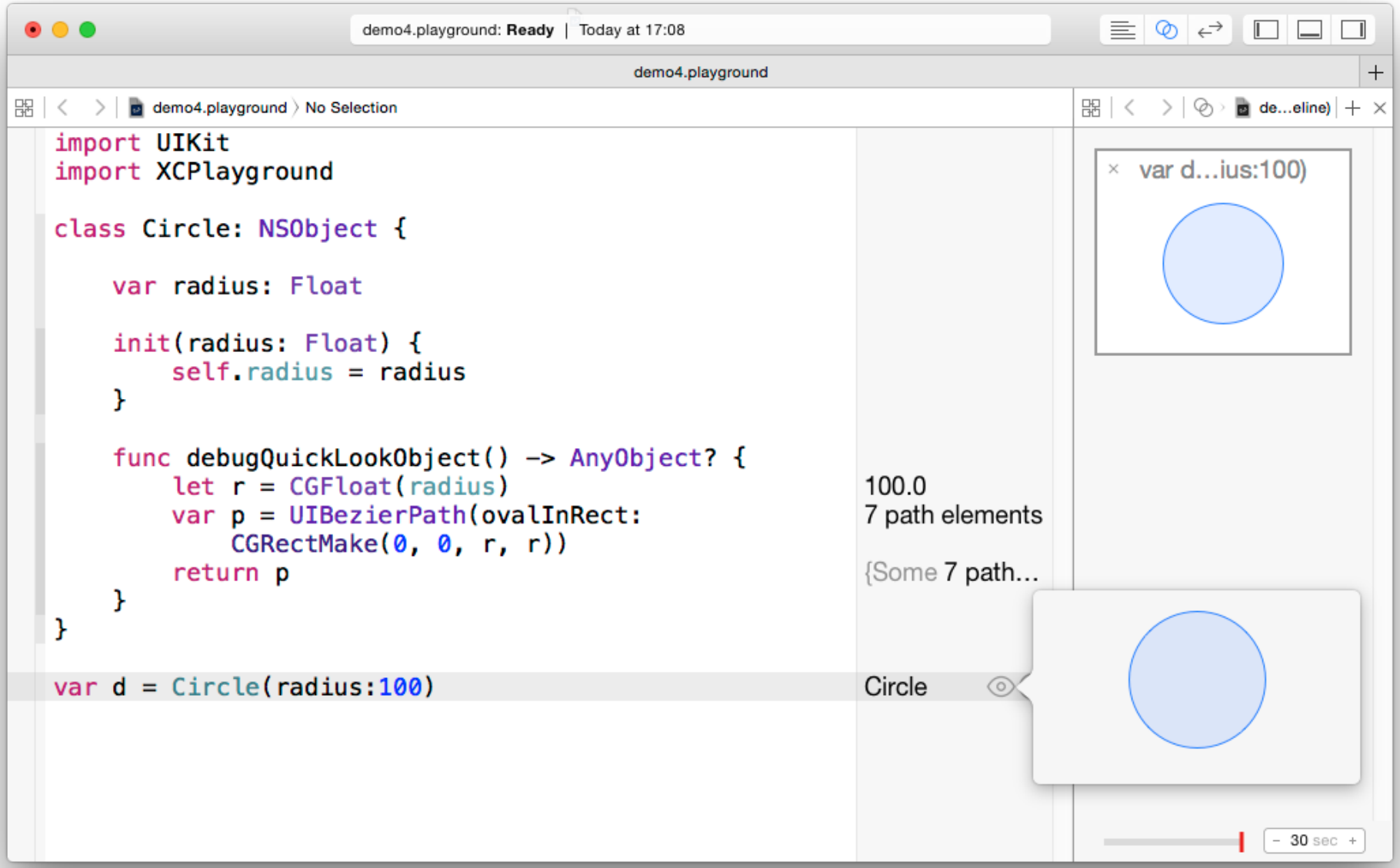




Quick Look

- El botón  muestra gráficamente el valor de una expresión.
- Tipos soportados:
  - Colores, Strings, Arrays, Diccionarios, Clases, Estructuras, Imágenes, Views, Bezier paths, Puntos, Dimensiones, Rectángulos, URLS, …
- Podemos crear Quick Looks personalizados para subclases de NSObject.
  - Implementando el método:

```
func debugQuickLookObject() -> AnyObject? {
    return <Color, String, Imagen o Bezier Path>
}
```

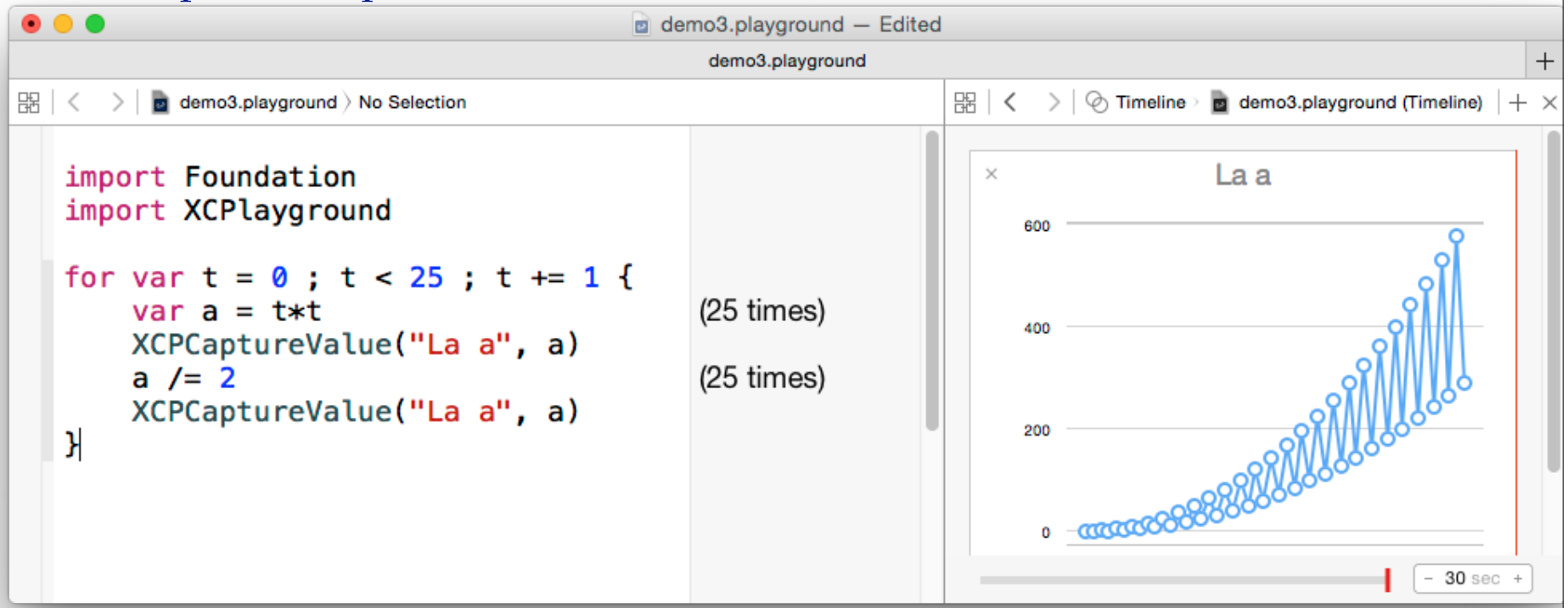


Framework XCPlayground

- Contiene utilidades para usar en los Playgrounds:

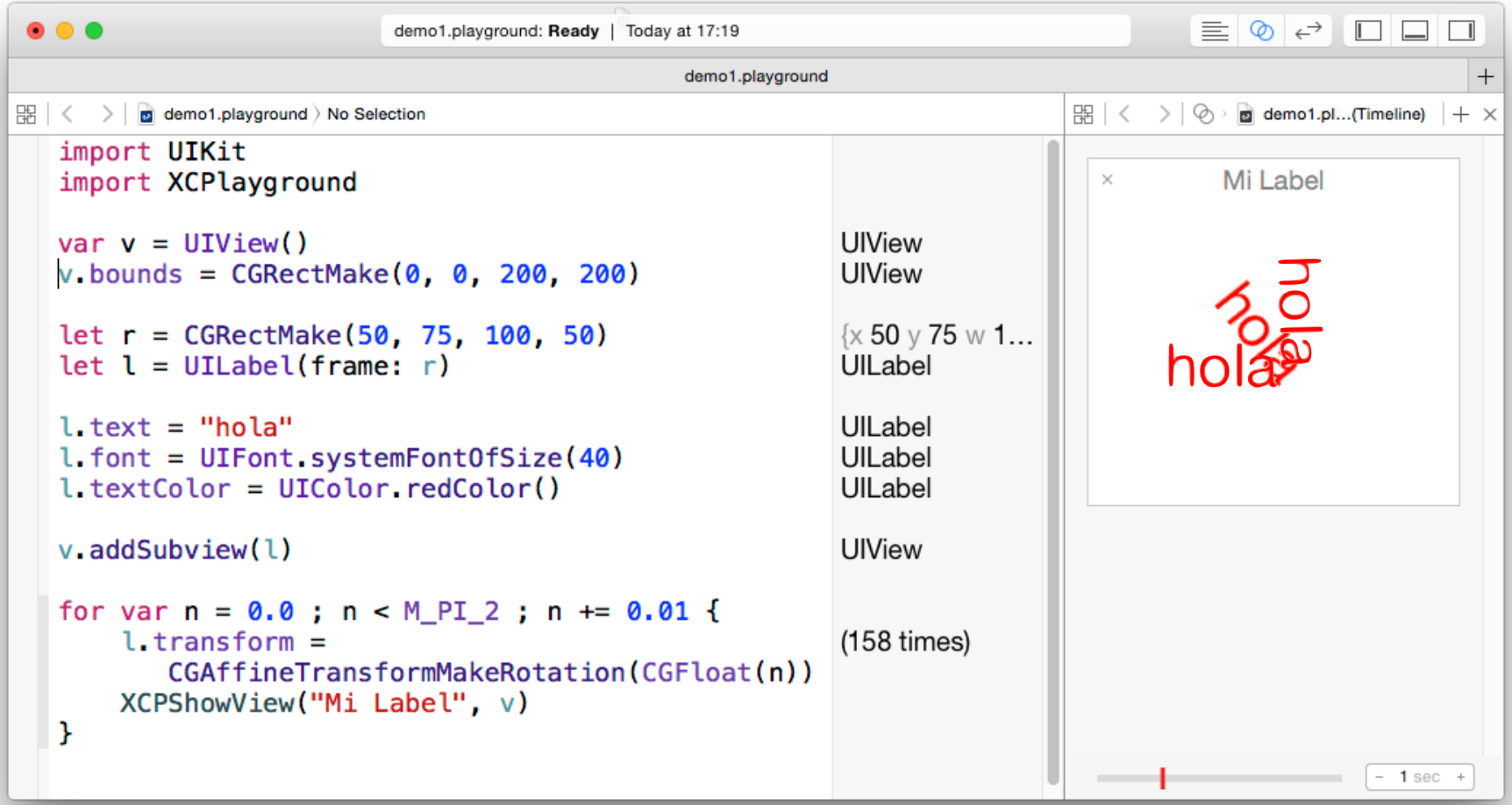
```
func XCPCaptureValue<T>(identifier: String, value: T)
```

  - Capturar valores en distintas partes del código, asociando los valores capturados con un identificador, y mostrar la evolución de los valores capturados para cada identificador.



```
func XCPShowView(identifier: String, view: UIView)
```

- Mostrar los cambios de una view, asociándolos con un identificador. Se crea una animación con la evolución de la view.
- La view a capturar no debe tener una superview.
  - La view se añade automáticamente a una superview para mostrarse en el timeline.



```
func XCPExecutionShouldContinueIndefinitely() -> Bool

func XCPSetExecutionShouldContinueIndefinitely(
                    continueIndefinitely: Bool = default)
```

- La ejecución del playground dura mientras se está ejecutando el código introducido.
  - Sin tener en cuenta si existen tareas en background, planificadas para ejecutarse en un futuro, o esperando respuestas de forma asíncrona.
  - Si se edita el código, se detiene la ejecución y se empieza a ejecutar todo de nuevo.
- Adicionalmente, la ejecución se prolonga por el tiempo indicado por el slider de timeout existente en el timeline. Por defecto son 30 segundos adicionales.
  - Estas funciones se usan para indicar (o consultar) si este tiempo adicional de ejecución debe concederse o no.

- Nota: la función **XCPShowView** llama implícitamente a la función **XCPSetExecutionShouldContinueIndefinitely** para habilitar este tiempo adicional.

Limitaciones

- No es válido para analizar las prestaciones del código.
- No puede usarse para tareas que requieran:
  - interacciones del usuario.
  - entitlements.
  - ejecución en un terminal real.
  - usar código existente en la app/framework.
    - (copiarlo al playground)