Bloque config

* Se ejecuta 1º
* Sólo para configurar providers (es la forma de configurar los servicios antes de usarlos en run o en otro lugar de la aplicación. Usamos el provider para que un servicio sea configurable y usamos el bloque config para configurar el provider y este al servicio)
* NO puede acceder a $rootScope

Bloque run

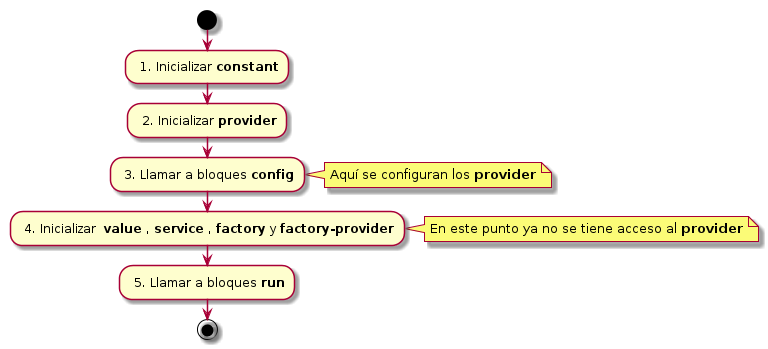
* Se ejecuta 2º
* Parecido a un Main , se ejecuta cualquier cosa
* Puede acceder a $rootScope

Tipos

* constant
* value
* service
* factory
* provider

La principal diferencia es donde se puede inyectar.

|  |  |  |  |
| --- | --- | --- | --- |
| **Se puede inyectar en un** | | | |
| **Tipo** | **Provider** | **Bloque config** | **Servicio**[**1)**](http://www.cursoangularjs.es/doku.php?id=unidades:03_servicios:04_tiposservicios#fn__1) | **Bloque run** |
| constant | Si | Si | Si | Si |
| provider | Si [2)](http://www.cursoangularjs.es/doku.php?id=unidades:03_servicios:04_tiposservicios#fn__2) | Si | No | No |
| value | No | No | Si | Si |
| service | No | No | Si | Si |
| factory | No | No | Si | Si |
| factory-provider [3)](http://www.cursoangularjs.es/doku.php?id=unidades:03_servicios:04_tiposservicios#fn__3) | No | No | Si | Si |



**Actividad 1**: Se puede ver cómo lo primero que se hace es inicializar las constant, por lo tanto éstas se pueden inyectar en cualquier sitio.

**Actividad 2:** Seguidamente se crean los provider

**Actividad 3:** Se llaman a los bloques config. Por lo tanto en un bloque config se puede inyectar tanto una constant como un provider. En un bloque config lo que se hace es configurar los provider.

**Actividad 4**: Ahora es cuando se llama a todos los servicios normales que son los value , service , factory y factory-provider. Llegados a este punto los provider no están accesibles ya que al estar ya configurados solo se tiene acceso a los factory-provider correspondientes a sus provider.

**Actividad 5:** Se llama al bloque run para inicializar la aplicación pudiendo inyectar cualquier tipo de servicio excepto los provider

Diferencias

Entre constant y value es que value no puede ser usado en provider ni config

Un service es una función que actua como constructor de una clase. Angular permite que lo podamos inyectar servicios como por ejemplo inyectarle un value

La diferencia entre service y value, es que service se usa para poder inyectar dependencias dentro cuando el valor es un objeto de una clase.

Value, service y Factory son internamente para angular providers, pero al ser estos un poco mas complejos de programar, se han creado estos.

Una ventaja de un service sobre un value , es que si nunca se va a usar, no se instancia, mientras no se use. Por lo que es mejor usar service cuando es una instancia de una clase ya que

* Nos ahorramos la creación de la instancia
* Permite inyectar dependencias