

# Practicando Experiencia 2

ga.soto@duocuc.cl [Cambiar de cuenta](#)



Borrador guardado

**\*Obligatorio**

Correo \*

ga.soto@duocuc.cl

Nombre - Sección

Gabriel Soto - PGY005D

3. El método `coleccion.push(dato)` permite:

- ☐ Eliminar un objeto llamado dato de la colección o storage.
- ☐ Modificar un objeto llamado dato en la colección o storage.
- ☒ Agregar un objeto llamado dato a la colección o storage.

Borrar selección



8. ¿Cuáles son los tres parámetros que se pueden utilizar en un método que crea un alert:

- ☐ title, message, buttons
- ☐ title, content, button
- ☒ header, message, buttons
- ☐ header, message, button

Borrar selección

7. Para dar acceso a un determinado page una vez comprobado las credenciales de acceso de un usuario, el método que utilizamos es:

- ☐ navigate.Root
- ☒ NavigateRoot
- ☐ NavigateRoute
- ☐ Navigate.Route

Borrar selección

2. Para obtener un valor desde un storage, el comando es:

- ☐ getKey()
- ☐ getKeys(value)
- ☒ get('Key')
- ☐ getting(value)

Borrar selección



5. Existen dos tipos de formularios en angular para Ionic:

- ☐ Formulario basado en plantilla y formulario basado en clase.
- ☐ Formulario basado en plantilla y formulario basado en los datos.
- ☒ Formulario basado en plantilla y formulario reactivo.

Borrar selección

6. La biblioteca que se necesita importar para trabajar con formularios basados en el componente es:

- ☐ `import{FormsModule} from '@angular/forms'`
- ☐ `import{ReactiveForms} from '@angular/forms'`
- ☐ `import {ReactiveFormsModule} from '@angular/forms'`
- ☒ `import {FormsModule} from '@angular/forms'`

Borrar selección

1. Para instalar Ionic Storage el comando es:

- ☐ `node install @ionic/storage`
- ☐ `node install --save @ionic/storage`
- ☒ `npm install --save @ionic/storage`

Borrar selección



4. Un servicio se crea a través del comando:

- ☐ ionic install service 'ruta'
- ☒ ionic generate service 'ruta'
- ☐ npm generate service 'ruta'

Borrar selección

9. Se necesita programar en un servicio que devuelva todos los objetos desde el storage. La definición que permite realizar esto es:

- ☐ async getDatos(Datos[]): Promise<Datos[]>
- ☐ async getDatos(Datos[]): Promise<>
- ☒ async getDatos(): Promise<Datos[]>
- ☐ async getDatos(): Promise<Datos>

Borrar selección

10. FormGroup, FormControl, Validators, FormBuilder se importan desde:

- ☐ @ionic/forms
- ☒ @angular/forms
- ☐ Las dos alternativas anteriores son válidas

Borrar selección

Enviar

Borrar formulario

Nunca envíes contraseñas a través de Formularios de Google.



Este formulario se creó en DuocUC. [Notificar uso inadecuado](#)

# Google Formularios

