



**Universidad Tecnológica
del Norte de Guanajuato**
Organismo Público Descentralizado del Gobierno del Estado
"Educación y progreso para la vida"



Nombre: Cristian Israel Buclon Pedroza.

Num.Control: 1220100646

Grupo: GDS0351

Materia: Aplicaciones Web para I4.0.

Profesor: José De Jesús Eduardo

Barrientos Avalos

Actividad: Ejercicio 4. Stepper

1. Explica las siguientes líneas de tu app

`import 'package:flutter/material.dart';`

Esta importa los paquetes necesarios para poder trabajar con distintos elementos que flutter puede aportar, es decir que está sola línea puede importar todos y cada uno de materiales para que la aplicación pueda trabajar ya sea para el objetivo para el cual está destinada, por ejemplo el contenido para poder nombrar alguna aplicación en específico o el ir formándola de apoco para que esta quede lista para que el programador pueda ir introduciendo elementos más complejos.

2. ¿Qué es una función abreviada y una función anónima? Escribe un ejemplo de cada una

La función abreviada es aquella que al dar una palabra en especifica se completa la una función entera para que se complemente de forma automática gracias a las funciones de flutter y la función anónima es aquella función que se realiza de forma manual en la que se flutter no este consciente de la variable que se introduzca.

3. ¿En esta aplicación utilizas alguna función abreviada o anónima?, ejemplifica con código

En este ejercicio se implementó una función anónima la cual maneja el comportamiento del cancel y todas aquellas propiedades que puedan presentarse para que la aplicación pueda cumplir de manera satisfactoria con el ciclo antes establecido.

```
//funcion anonima para manejar el comportamiento de cancel
setState(() {
  if (currentStep > 0) {
    currentStep = currentStep - 1;
  } else {
    currentStep = 0;
  }
});
},
onStepTapped: (step){
  setState(() {
    currentStep=step;
  });
},
),); // Stepper // Container
}
```

4. Explica: ¿Que hace la siguiente línea de código? `void main() => runApp(MyApp());`

Esta sola línea puede hacer que el código especificado por el programador pueda ejecutarse de manera completa, solo si flutter detecta que el código esta correcto ya que esta línea no funciona si algún elemento del código no funciona para el propósito al cual está dirigido.

5. En qué consiste la clase StatelessWidget y ¿Dónde se encuentra?

La función de stateless tiene el objetivo de empezar a formar la aplicación con elementos esencialmente básica para poder formar cuestiones de diseño para poder implementar un diseño básica para la aplicación.

```
5 class MyApp extends StatelessWidget {
6   @override
7   Widget build(BuildContext context) {
8
9     return MaterialApp(
10       debugShowCheckedModeBanner:false,
11       title: 'Demo Stepper',
12       home: Scaffold(
13         appBar: AppBar(
14           title: Text('Demostracion del Stepper'), //se puede ver en el appbar
15           backgroundColor: Colors.blue, //color de fondo
16         ), // AppBar
17         body: Center(
18           child: new MyStepper() //Creando objeto de la clase que define widget del Stepper (aparecia)
19         ), // Center
20       ), // Scaffold
21     ); // MaterialApp
22
23   }
24 }
```

6. ¿Cuáles son las propiedades para manejar el comportamiento de los Steps de continuar y cancelar?

El comportamiento CONTINUE para seguir con el ciclo

```
48     steps: mySteps,
49     currentStep: _currentStep,
50     onStepContinue: () {
51       setState(() {
52         //lo que queremos que haga
53         if ( _currentStep < mySteps.length -1){
54           //condicion para limitar con la coleccion
55           currentStep = _currentStep + 1; //Incrementa contador
56         } else {
57           currentStep = 0; //El contadpr se reinicia
58         }
59       });
60     },
```

El comportamiento para cancelar el ciclo y regresar el ciclo.

```

61         onStepCancel: () {
62             //funcion anonima para manejar el comportamiento de cancel
63             setState(() {
64                 if (_currentStep > 0) {
65                     currentStep = currentStep - 1;
66                 } else {
67                     currentStep = 0;
68                 }
69             });
70         },
71         onStepTapped: (step) {
72             setState(() {
73                 currentStep = step;
74             });
75         },

```

7. Explica que se hace en:

MyStepper({Key? key}) : super(key: key);

Es el constructor el cual empieza con la definición para poder implementar el método que definirá la lista a pasos al menos en esta ocasión, es decir es el constructor de la clase MyStepper.

8. ¿Menciona el nombre de la clase y el método en dónde se establecen los Widgets contenidos y se establece el comportamiento?

Todos y cada uno de los widget se encuentra en la clase de MyStepperState en la cual establece el método widget el cual realiza un método para poder realizar un ciclo para poder empezar que la lista fluya al darle al botón aceptar y si no al darle al botón cancelar, esto hará que el contenedor se reinicie a 0 para poder regresar al paso anterior de la lista.

9. En qué consiste la clase StatefulWidget y ¿Dónde se encuentra?

Esta establece la colección de la lista con todos y cada uno de sus datos para que se muestren en la aplicación.

```

class _MyStepperState extends State<MyStepper> {
    //en esta clase incorporamos el contenido y comportamiento
    int _currentStep = 0 ;
    //crear una coleccion (List) de pasos (Step)
    List<Step> mySteps = [new Step(title: Text ('Paso 1 '), content: Text ('Identificar el objetivo')),
                           new Step(title: Text ('Paso 2 '), content: Text ('Abordar objetivo')),
                           new Step(title: Text ('Paso 3 '), content: Text ('Conocer objetivo')),
                           new Step(title: Text ('Paso 4 '), content: Text ('Desarrollador objetivo'))];
}

```

10. ¿Para qué sirve la función setState en esta aplicación?

Se puede establecer el método para mantener el ciclo if para que la lista de pasos se pueda corregir el estatus para poder mantener el estado de la colección para poder limitar la colección y que no sea limitado