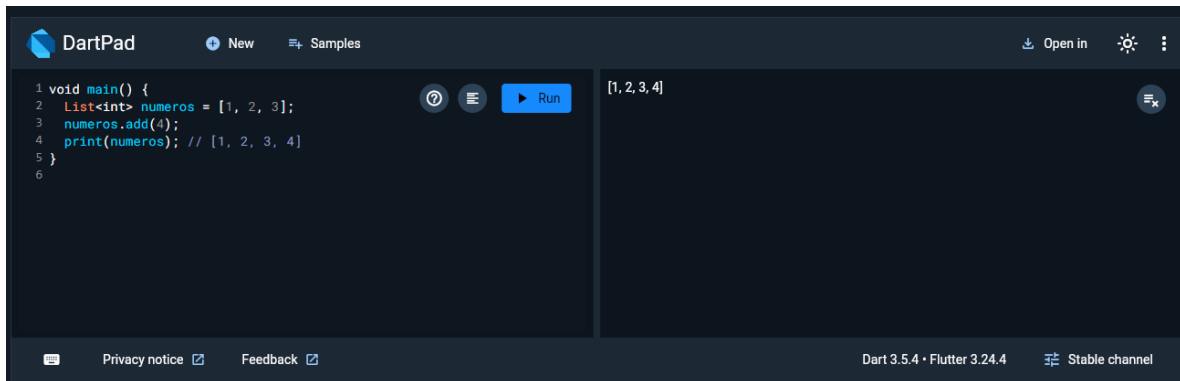


**Alumno:** Hugo Bustamante Ramírez.

## Lists

Una lista en Dart es una colección ordenada de elementos. Puede ser fija o dinámica.

Puedes acceder a los elementos por su índice, modificar valores y realizar operaciones como agregar, eliminar y ordenar.



The screenshot shows the DartPad interface. On the left, the code editor contains the following Dart code:

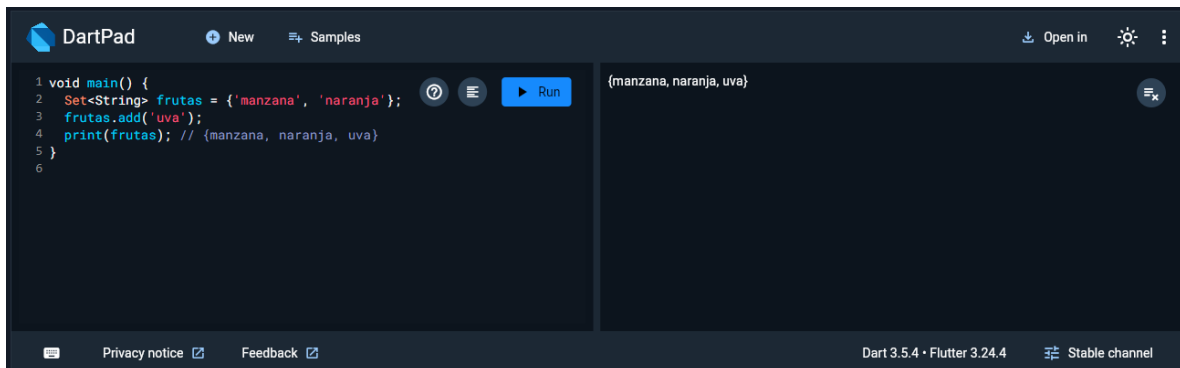
```
1 void main() {  
2   List<int> numeros = [1, 2, 3];  
3   numeros.add(4);  
4   print(numeros); // [1, 2, 3, 4]  
5 }  
6
```

On the right, the output console displays the result of the execution: `[1, 2, 3, 4]`. The interface includes a 'Run' button and a 'Privacy notice' link at the bottom.

## Sets

Un Set es una colección no ordenada que no permite duplicados.

Ideal para verificar si un elemento existe o para operaciones de conjuntos como unión o intersección.



The screenshot shows the DartPad interface. On the left, the code editor contains the following Dart code:

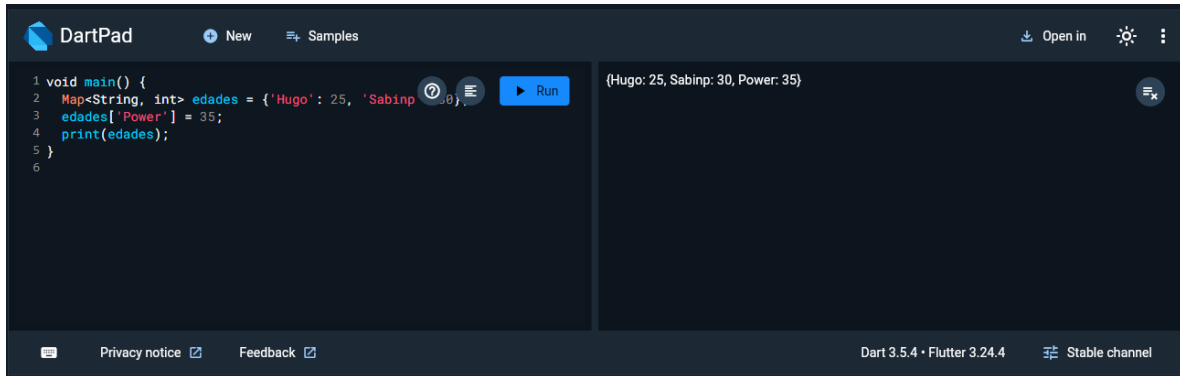
```
1 void main() {  
2   Set<String> frutas = {'manzana', 'naranja'};  
3   frutas.add('uva');  
4   print(frutas); // {manzana, naranja, uva}  
5 }  
6
```

On the right, the output console displays the result of the execution: `{manzana, naranja, uva}`. The interface includes a 'Run' button and a 'Privacy notice' link at the bottom.

## Maps

Los Maps son pares clave-valor, donde las claves deben ser únicas.

Se usan para acceder a valores a través de sus claves.



The screenshot shows the DartPad interface. On the left, the code editor contains the following Dart code:

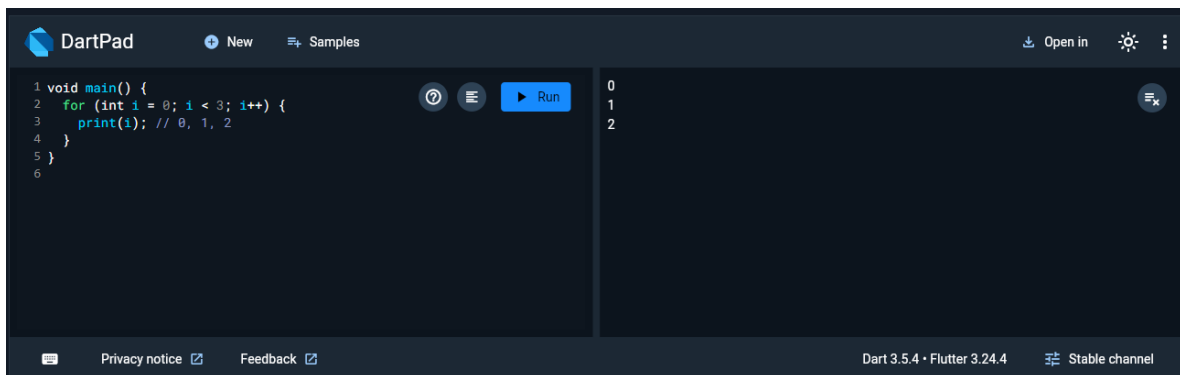
```
1 void main() {  
2   Map<String, int> edades = {'Hugo': 25, 'Sabinp  
3   edades['Power'] = 35;  
4   print(edades);  
5 }  
6
```

On the right, the output console displays the result of the code execution: `{Hugo: 25, Sabinp: 30, Power: 35}`. The bottom status bar indicates the environment is Dart 3.5.4 and Flutter 3.24.4 on the Stable channel.

**for:** Itera un número fijo de veces.

**while:** Itera mientras una condición sea verdadera.

**do-while:** Itera al menos una vez antes de verificar la condición.



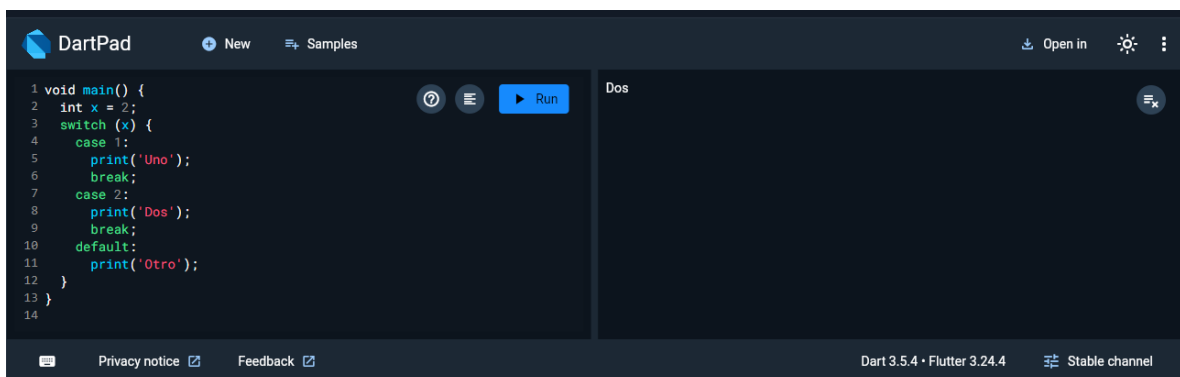
The screenshot shows the DartPad interface. On the left, the code editor contains the following Dart code:

```
1 void main() {  
2   for (int i = 0; i < 3; i++) {  
3     print(i); // 0, 1, 2  
4   }  
5 }  
6
```

On the right, the output console displays the result of the code execution: `0`, `1`, and `2` on separate lines. The bottom status bar indicates the environment is Dart 3.5.4 and Flutter 3.24.4 on the Stable channel.

**if:** Verifica una condición y ejecuta código si es verdadera.

**switch:** Selecciona entre múltiples casos basados en un valor.



The screenshot shows the DartPad interface. On the left, the code editor contains the following Dart code:

```
1 void main() {  
2   int x = 2;  
3   switch (x) {  
4     case 1:  
5       print('Uno');  
6       break;  
7     case 2:  
8       print('Dos');  
9       break;  
10    default:  
11      print('Otro');  
12    }  
13 }  
14
```

On the right, the output console displays the result of the code execution: `Dos`. The bottom status bar indicates the environment is Dart 3.5.4 and Flutter 3.24.4 on the Stable channel.