Queues in dart

Exercise #5

Flutter Developer Bootcamp

# **Purpose**

This Exercise demonstrates how to create, add elements to, access elements in, and remove elements from a queue in Dart using the Queue class. It also demonstrates basic queue operations such as retrieving the first and last elements and removing elements until the queue becomes empty.

**Problem**

In the provided Exercise You need to create as follows:

Add integers (10, 20, 30, 40, and 50) to it, prints the initial queue, accesses the first and last elements, removes elements from the queue until it's empty, and finally prints the queue after removal.

**How to Solve**

1. Checkout the Exercise from Git Repo:

git clone -b <user-branch> <repo-URL>

2. Open the root folder inside VS Code

3. Open the root folder in terminal

4. Run the command dart run filename.dart

5. Queue Creation:

* The code initializes a queue named queue using the Queue class from the dart:collection library.

6. Add Elements:

* The integers 10, 20, 30, 40, and 50 are added to the queue using the add method.

7. Print Initial Queue:

* The code prints the initial state of the queue using print("Initial Queue: $queue").

8. Accessing Elements:

* The first element of the queue is accessed using the first property and stored in the variable firstElement.
* Similarly, the last element of the queue is accessed using the last property and stored in the variable lastElement.
* Both the first and last elements are printed.

9. Removing Elements:

* The code enters a while loop that continues as long as the queue is not empty (queue.isNotEmpty).
* Inside the loop, the first element of the queue is removed using the removeFirst method and stored in the variable removedElement.
* The removed element is printed.
* This process continues until the queue becomes empty.

10. Printing Queue After Removal:

* After all elements have been removed from the queue, the code prints the state of the queue again using print("Queue after removal: $queue")..

11. Go To File: <specific-file-with-async and await-method> à <method-name>, implement your logic.

**You will Achieve**

When you complete this Exercise you will learn the following:

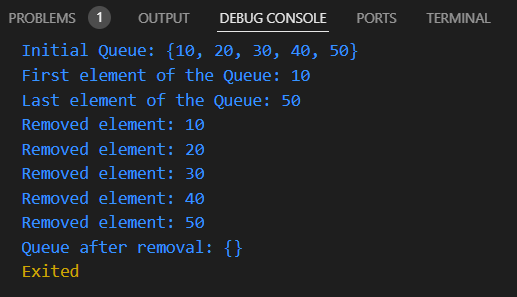
* **Initialization:** Initialize a queue using Dart's Queue class from the dart:collection library.
* **Adding Elements:** Add elements (10, 20, 30, 40, 50) to the queue using the add method.
* **Printing Initial Queue:** Print the initial state of the queue using print.
* **Accessing Elements:** Access the first and last elements of the queue using the first and last properties, respectively.
* **Removing Elements:** Remove elements from the queue one by one until it becomes empty using a while loop and the removeFirst method.
* **Printing Queue After Removal:** Print the state of the queue after all elements have been removed.

**Functions and methods involved:**

* **Queue:** Initializes a new queue.
* **add:** Adds an element to the queue.
* **first:** Retrieves the first element of the queue.
* **last:** Retrieves the last element of the queue.
* **removeFirst:** Removes and returns the first element of the queue.
* **isNotEmpty:** Checks if the queue is not empty.
* **print:** Prints the queue elements to the console.

# **Screenshots**

## **Expected output (using Queues)**



## 

# **How to submit your Exercise**

Push your project back to the same git branch using command:

<command name>

# **Happy Coding!**