Create A Clock Recipe

Ingredients:

- Library (UIKit)

- Function (startClock)

- Loops (Timer, RunLoop)

- Variable (currentDate, dateFormatter)

Recipe:

Step 1: Import UIKit

- Import the UIKit library to access the necessary iOS framework functionalities.

Step 2: Define the `startClock` function

- Create a function named `startClock` responsible for initiating and managing the clock.

Step 3: Create a Timer in the `startClock` function

- Inside the `startClock` function, create a Timer that fires every 1 second (1-second interval) and repeats (`repeats: true`). The closure inside the timer is where we'll update the clock.

Step 4: Add the timer to the current RunLoop

- Add the created timer to the current RunLoop using the common mode, allowing it to run in the main thread.

Step 5: Define the `updateClock` function

- Create a function named `updateClock` to handle updating the clock's display with the current time.

Step 6: Get the current date and time

- Within the `updateClock` function, obtain the current date and time using the `Date` class.

Step 7: Create a DateFormatter to format the time

- Instantiate a `DateFormatter` object to format the time in the "HH:mm:ss" format.

Step 8: Print the current time

- Inside the `updateClock` function, print the current time in the specified "HH:mm:ss" format using the `dateFormatter`.

Step 9: Start the clock

- Call the `startClock` function to begin the clock's operation.

Step 10: Keep the playground running

- For the clock continues updating, adding the `import PlaygroundSupport` statement to the code. This keeps the playground running indefinitely.