

EXERCISE 6: ASYNC COUNTDOWN TIMER

```
import flet as ft
import asyncio

# Define a new class Countdown where we define the timer
class Countdown(ft.Column):
    def __init__(self, seconds):
        super().__init__()
        self.initial_seconds = seconds
        self.seconds = seconds
        self.running = False

        # Display elements for the countdown timer
        self.time_display = ft.TextField(f"Initial value:
            {self.format_time(self.initial_seconds)}",
            read_only=True, width=200)

        # Control buttons
        self.start_btn = ft.ElevatedButton("Start",
            on_click=self.start)
        self.pause_btn = ft.ElevatedButton("Pause",
            on_click=self.pause)
        self.reset_btn = ft.ElevatedButton("Reset",
            on_click=self.reset)

        # Define the progress bar
        self.progress = ft.ProgressBar(value=0.0)
        self.controls = [
            ft.Row([self.time_display,
                alignment=ft.MainAxisAlignment.CENTER),
            ft.Row([self.start_btn, self.pause_btn, self.reset_btn],
                alignment=ft.MainAxisAlignment.CENTER),
            self.progress
        ]

    def start(self, e=None):
        self.running = True
        self.page.run_task(self.update_timer)

    def pause(self, e=None):
        self.running = False

    def reset(self, e=None):
        self.seconds = self.initial_seconds
        self.running = False
        self.update_display()
```

```
def update_display(self):
    self.time_display.value = self.format_time(self.seconds)
    self.progress.value = (self.initial_seconds - self.seconds)
        / self.initial_seconds
    self.update()

def format_time(self,seconds):
    mins, secs = divmod(seconds, 60)
    return "{:02d}:{:02d}".format(mins, secs)

async def update_timer(self):
    while self.seconds > 0 and self.running:
        self.update_display()
        await asyncio.sleep(1)
        self.seconds -= 1
        if not self.running:
            break
        self.update_display()

# Define the main function
def main(page: ft.Page):
    page.title = "Async Countdown"
    page.update()

    # Create three tabs, with a timer in each
    tabs = ft.Tabs(
        selected_index=0,
        animation_duration=300,
        tabs = [
            ft.Tab(text="Timer 1", content = Countdown(120)),
            ft.Tab(text="Timer 2", content = Countdown(140)),
            ft.Tab(text="Timer 3", content = Countdown(160))
        ]
    )
    page.add(tabs)
ft.app(main)
```

