```python
import uasyncio as asyncio
from agbot import AgBot
from clock import Clock

import time

class Controller:
    @classmethod
    def get_default_controller(cls):
        agBot = AgBot.get_default_agbot()
        clock = Clock.get_default_clock()
        return Controller(agBot, clock)

    def __init__(self, agbot, clock):
        self.agbot = agbot
        self.clock = clock
        self.agbot.stop()

    async def take_moisture_reading(self):
        moisture = await self.agbot.read()
        print(f"Moisture level: {moisture:.2f}%")

    async def water_plant(self, water_amount: float):
        print(f"Watering with {water_amount} units...")
        await self.agbot.water(water_amount)

    async def routine(self, moisture_threshold: float, water_amount: float):
        moisture = await self.agbot.read()
        print(f"Moisture level: {moisture:.2f}%")
        if moisture > moisture_threshold:
            print("Moisture low — watering...")
            await self.agbot.water(water_amount)
        else:
            print("Moisture OK — no watering.")

if __name__ == "__main__":

    print("Welcome to the AgXRP Mini!\n")
    time.sleep(2)

    print("Choose an option:")
    print("1. Take a moisture reading.")
    print("2. Water a plant.")
    print("3. Both!\n")

    choice = input("Enter 1, 2, or 3: ")

    controller = Controller.get_default_controller()

    if choice == "1":
        asyncio.run(controller.take_moisture_reading())

    elif choice == "2":
        water_amount = float(input("Enter water amount: "))
        asyncio.run(controller.water_plant(water_amount))
        print("\nDone!")

    elif choice == "3":
        moisture_threshold = float(input("Enter moisture threshold (%): "))
        water_amount = float(input("Enter water amount: "))
        asyncio.run(controller.routine(moisture_threshold, water_amount))
        print("\nDone!")

    else:
```

```
65          print("Invalid choice. Please run the program again.")
66
67
```