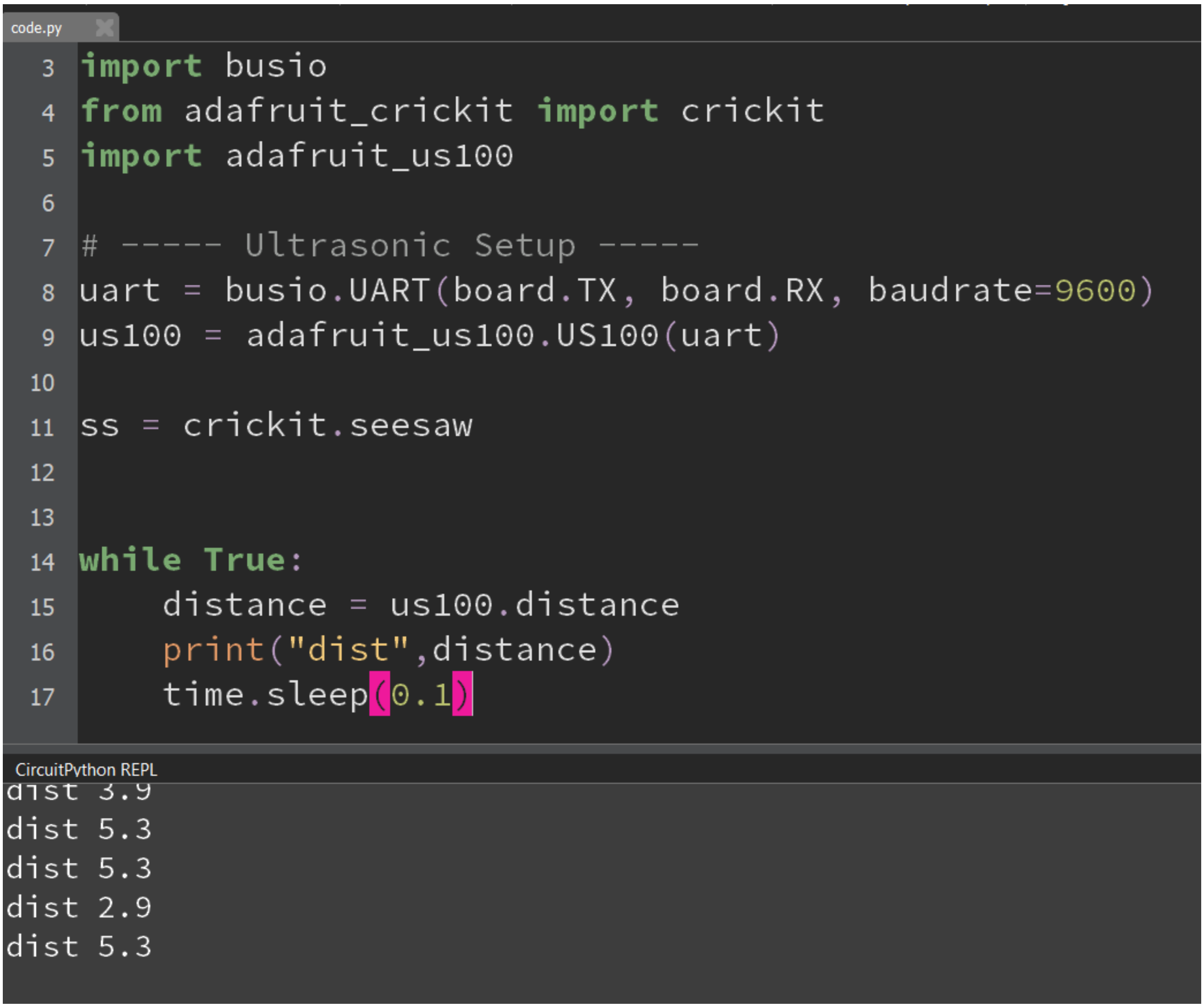
Solution-1



```python
import busio
from adafruit_crickit import crickit
import adafruit_us100

# ----- Ultrasonic Setup -----
uart = busio.UART(board.TX, board.RX, baudrate=9600)
us100 = adafruit_us100.US100(uart)

ss = crickit.seesaw


while True:
    distance = us100.distance
    print("dist",distance)
    time.sleep(0.1)
```

```
dist 3.9
dist 5.3
dist 5.3
dist 2.9
dist 5.3
```

Code Explanation:

```python
import time
import board
import busio
from adafruit_crickit import crickit
import adafruit_us100
```

- time → allows delays and timing operations (used for sleep()).
- board → provides pin names (like board.TX, board.RX) for the connected hardware.
- busio → used to create communication buses like UART, I2C, or SPI.

adafruit_crickit → library for controlling the Crickit robotics board.

- adafruit_us100 → library for controlling the US100 ultrasonic sensor.

```
# ----- Ultrasonic Setup -----
uart = busio.UART(board.TX, board.RX, baudrate=9600)
us100 = adafruit_us100.US100(uart)
```

- *busio.UART(board.TX, board.RX, baudrate=9600)*
  Creates a **UART serial connection** between the board and the US100 sensor.
- TX = transmit pin
- RX = receive pin
- baudrate=9600 = communication speed (must match the sensor's default rate)
- *us100 = adafruit_us100.US100(uart)*
  Creates an instance of the US100 sensor, allowing you to read distance and temperature data.

```
ss = crickit.seesaw
```

- Initializes the **Seesaw** chip on the Crickit board.
  (Even though not directly used here, it's often needed when integrating with Crickit.)

```
while True:
    distance = us100.distance
    print("dist", distance)
    time.sleep(0.1)
```

while True: → creates an infinite loop that continuously reads sensor data.

- us100.distance → reads the **distance in centimeters** from the ultrasonic sensor.
- print("dist", distance) → displays the distance value in the serial console.
- time.sleep(0.1) → waits for 0.1 seconds before the next reading (to avoid flooding the output).