Code Explanation: Servo Turning Based on Light Detection (CPX + Crickit)

```
import time
from adafruit_crickit import crickit
```

• Imports the required libraries: time for adding delays, and crickit to control sensors and servo via the Crickit board.

```
ss = crickit.seesaw
```

• Creates a shortcut variable `ss` to interact with Crickit sensors and actuators.

Sensor Pin Setup

```
photo_left = crickit.SIGNAL2
```

• Left photoresistor (light sensor) connected to SIGNAL2.

```
ss.pin_mode(photo_left, ss.INPUT)
```

• Sets the left photoresistor pin as INPUT to read light intensity.

```
photo_right = crickit.SIGNAL6
```

• Right photoresistor (light sensor) connected to SIGNAL6.

```
ss.pin_mode(photo_right, ss.INPUT)
```

• Sets the right photoresistor pin as INPUT.

```
moisture = crickit.SIGNAL2
```

• Moisture sensor connected to SIGNAL2 (Analog input). (Note: Ensure this pin is different from photo_left in real wiring to avoid conflict.)

```
ss.pin_mode(moisture, ss.INPUT)
```

• Sets the moisture sensor pin as INPUT.

Main Loop

```
while True:
```

• Starts an infinite loop to continuously read sensor values and control the servo.

```
val_left = ss.analog_read(photo_left)
val_right = ss.analog_read(photo_right)
moisture_val = ss.analog_read(moisture)
```

• Reads analog values from the left light sensor, right light sensor, and moisture sensor.

```
print("Left:", val_left, " | Right:", val_right)
```

• Displays current light levels from both sides in the serial monitor.

Servo Control Logic

```
if val_left < 400 and val_right > 400:
    crickit.servo_1.angle = 0
    print('Turning to LEFT has light → Angle 0')
```

• If the left side has light and the right side does not, turn servo to 0° (face left).

```
elif val_right < 400 and val_left > 400:
    crickit.servo_1.angle = 180
    print('Turning to RIGHT has light → Angle 180')
```

• If the right side has light and the left side does not, turn servo to 180° (face right).

```
elif val_left < 400 and val_right < 400:
    crickit.servo_1.angle = 90
    print('Both sides light → CENTER at 90')
```

• If both sides have light, position servo at 90° (center).

```
else:
    print('No movement')
```

• If neither side has enough light, do nothing.

```
time.sleep(0.5)
```

• Wait 0.5 seconds before repeating the loop to avoid rapid movement.