Interrupts : work in the context of microcontrollers like the ESP32. The attachInterrupt() function in the Arduino IDE is used to set up an interrupt, and it takes the GPIO pin, the name of the function to be executed, and the interrupt mode as arguments. When an interrupt occurs, the processor stops executing the main program and jumps to the interrupt handler function, or ISR, which handles the event. This allows the microcontroller to respond promptly to external events without having to continuously poll for them in the main program loop.

attachInterrupt() function takes in three arguments:

1. **Function to be triggered**
2. **GPIO Interrupt**
3. **Mode**

**Timer: implement a motion detection system with a PIR sensor and an LED, we can use the millis() method as a timer to turn the LED on for a specific amount of time after motion is detected. This approach is preferred over using the delay() method, which would pause the program execution for a specified time instead of allowing it to continue running while the timer counts down. By using millis(), the program can continue to execute and perform other tasks while the LED is on, which can be useful in more complex applications.**

timers and interrupts are important features of the ESP32 microcontroller. Timers can be used to generate periodic events or measure time intervals with high accuracy, while interrupts allow for quick responses to external events without having to continuously poll for them in the main program loop