PROJECT 8: WI-FI NETWORK SCANNER

Objective:

To create a Python-based tool that scans nearby Wi-Fi networks and displays essential information such as SSID (network name), signal strength, and security type. This helps users identify the best available Wi-Fi connection.

Tools & Technologies Used:

- **Python 3.13**
- **Libraries**:
    - pywifi – for interacting with wireless adapters
    - comtypes – required dependency for Windows support
    - time – to pause during scanning
- **Platform**: Windows (tested)

Setup Instructions:

1. Install required libraries:

   pip install pywifi comtypes

python code:

```
import pywifi

from pywifi import const

import time


def scan_networks():
```

```
    wifi = pywifi.PyWiFi()
    iface = wifi.interfaces()[0]
    iface.scan()
    time.sleep(3)
    results = iface.scan_results()

    print(f"{'SSID':<30} {'Signal Strength (dBm)':<20} {'Security'}")
    print("-" * 70)

    seen = set()
    for network in results:
        ssid = network.ssid
        signal = network.signal
        auth = network.akm
        if ssid not in seen and ssid != "":
            seen.add(ssid)
            auth_type = 'Open' if not auth else 'Secured'
            print(f"{ssid:<30} {signal:<20} {auth_type}")

if __name__ == "__main__":
    scan_networks()
```

3. Run the script in command prompt:

   py wificode.py

How It Works:

- Uses pywifi to access the Wi-Fi interface
- Scans for nearby wireless networks
- Displays:
    - SSID (Wi-Fi name)
    - Signal strength in dBm (higher = better)
    - Security type: Open or Secured

```
C:\Users\Nikhilnick\OneDrive\Desktop>pip install pywifi
Collecting pywifi
  Downloading pywifi-1.1.12-py3-none-any.whl.metadata (2.7 kB)
Downloading pywifi-1.1.12-py3-none-any.whl (15 kB)
Installing collected packages: pywifi
Successfully installed pywifi-1.1.12

C:\Users\Nikhilnick\OneDrive\Desktop>
```

```
C:\Users\Nikhilnick\OneDrive\Desktop>pip install comtypes
Collecting comtypes
  Downloading comtypes-1.4.11-py3-none-any.whl.metadata (7.2 kB)
Downloading comtypes-1.4.11-py3-none-any.whl (246 kB)
Installing collected packages: comtypes
Successfully installed comtypes-1.4.11
```

```
C:\Users\Nikhilnick\OneDrive\Desktop>pip install comtypes
Collecting comtypes
  Downloading comtypes-1.4.11-py3-none-any.whl.metadata (7.2 kB)
Downloading comtypes-1.4.11-py3-none-any.whl (246 kB)
Installing collected packages: comtypes
Successfully installed comtypes-1.4.11

C:\Users\Nikhilnick\OneDrive\Desktop>py wificode.py
SSID                          Signal Strength (dBm) Security
------------------------------------------------------------------------
Airtel_ONEPIECE               -35                   Secured
ACT101556530399_5g            -97                   Secured
SS0824                        -94                   Secured
 Rammi                        -90                   Secured
Airtel_Vijayakumar806         -88                   Secured
Airtel_UdayKiran              -87                   Secured
DIRECT-Go-SPPL 4K GSmartTV    -84                   Secured
archerc20                     -83                   Secured

C:\Users\Nikhilnick\OneDrive\Desktop>
```

Use Cases:

- Choosing the best Wi-Fi network based on signal strength
- Detecting open or insecure networks
- Educational demonstration of network scanning with Python

Key Learnings:

- How to use Python to interact with system hardware (Wi-Fi adapter)
- Importance of handling external dependencies (like comtypes)
- Real-world application of scanning and filtering data

Conclusion: The Wi-Fi Network Scanner project successfully demonstrates how Python can be used to interact with system-level components to gather useful network information. By utilizing the pywifi library, the script efficiently scans and displays nearby Wi-Fi networks along with their signal strength and security type. This tool can assist users in selecting the most reliable and secure network available.

Through this project, a deeper understanding was gained about wireless networking, signal strength interpretation, and secure vs open networks. It also highlights the importance of using the right libraries and managing platform-specific dependencies (like comtypes on Windows). This scanner serves as a foundation for building more advanced network management or monitoring tools in the future.