

The background of the image is a dark screen with Python code in a monospaced font. The code is color-coded: comments are green, keywords like 'if', 'else', and 'print' are purple, and strings and variables are blue. A white circle is centered over the text 'FAKE NEWS DETECTOR'.

# FAKE NEWS DETECTOR

CREATED BY  
GAVIN ANDERSON



# WHAT DOES IT DO?

- **Accepts User Input:** Takes either a URL or raw article text from the user to begin analysis.
- **Analyzes Content:** If a URL is provided, it fetches and extracts the article text using BeautifulSoup; then checks the content for suspicious keywords often found in fake news.
- **Estimates Fake News Likelihood:** Outputs a simple confidence score based on keyword matches, helping users identify if the content might be misleading or false.

```
return False

check_news():
user_input = input("Enter a URL or article text: ")

is_url_input = is_url(user_input)
trusted = False
matched_keywords = check_fake_keywords(user_input)

if is_url_input:
    print("You entered a URL.")
    trusted = check_trusted_domain(user_input)
    if trusted:
        print("✅ This URL is from a trusted news source.")
    else:
        print("⚠️ This URL is not from a recognized trusted")
else:
    print("You entered article text.")

if matched_keywords:
    print("⚠️ Warning: This might be fake news based on the")
    print("→ " + ", ".join(matched_keywords))
else:
    print("✅ No suspicious keywords were found.") 2
```

# WHATS HAPPENING IN THE SCREENSHOT?

- The user chooses to check a URL or article text.
- They enter scammer.com, which the system treats as text input.
- The script scans the text and detects the suspicious keyword: scam.  
It gives a warning that the content might be fake news, based on that keyword.
- The tool then returns to the main menu for further use.

```
==== Main Menu ====
1. Check a URL or article te
2. Exit
Choose an option (1-2): 1
Enter a URL or article text:
You entered article text.
▲Warning: This might be fak
→ scam

==== Main Menu ====
1. Check a URL or article te
2. Exit
Choose an option (1-2): █
```

```
def parse_arguments():
    """Read arguments from a command line."""
    parser = argparse.ArgumentParser(description='Arguments get parsed via --commands')
    parser.add_argument('-v', metavar='verbosity', type=int, default=2,
                        help='Verbosity of logging: 0 -critical, 1 -warning, 2 -info, 3 -debug')

    args = parser.parse_args()
    verbose = {0: logging.CRITICAL, 1: logging.WARNING, 2: logging.INFO, 3: logging.DEBUG}
    logging.basicConfig(format='%(message)s', level=verbose[args.v], stream=sys.stdout)

    return args

def main():
    pass

if __name__ == '__main__':
    args = parse_arguments()
    main()
```

# FAKE NEWS DETECTOR

## THANK YOU

This is my first Python project, but certainly not my last. Through this experience, I developed a stronger understanding of programming fundamentals, problem-solving, and how to approach a real-world challenge with structured logic. It has laid a solid foundation for future development work.