Lab 5: Ethical Foundations – Responsible AI Coding Practices

M.PARAMESH

HNO : 2303A53021

BATCH-46

Aim

To study ethical issues in AI-generated code and practice responsible AI coding by addressing security, privacy, transparency, and accountability.

Theory

AI tools can generate code quickly, but such code may include ethical risks like insecure data handling, lack of transparency, and biased logic. Developers must carefully analyze, modify, and validate AI-generated code to ensure it follows ethical and secure coding standards.

Task 1: Privacy in API Usage

Objective: Fetch weather data securely without exposing API keys.

Secure Code

```
import os
import requests

API_KEY = os.getenv("WEATHER_API_KEY") CITY
= "Delhi"

url = f"https://api.openweathermap.org/data/2.5/weather?q={CITY}&appid={API_KEY}"
```

```
response = requests.get(url) data = response.json()

print("Temperature:", data["main"]["temp"])
```

Sample Output

```
Temperature: 301.15
```

Task 2: Privacy & Security in File Handling

Objective: Store user data securely by hashing passwords.

Secure Code

```
import hashlib

name = input("Enter name: ") email =
input("Enter email: ") password = input("Enter password: ")

hashed_password = hashlib.sha256(password.encode()).hexdigest()

with open("users.txt", "a") as file:

file.write(f"{name},{email},{hashed_password}\n")

print("User data stored securely")
```

Sample Output

```
Enter name: Ravi
Enter email: ravi@gmail.com
Enter password: ****
User data stored securely
```

Task 3: Transparency in Algorithm Design (Armstrong Number)

Objective: Implement Armstrong number checking with transparent logic.

Code

```
def is_armstrong(num):
digits = str(num)     power
= len(digits)     total = 0

    for d in digits:

total += int(d) ** power
```

```
return total == num

print(is_armstrong(153))
```

Sample Output

```
True
```

Task 4: Transparency in Algorithm Comparison

Bubble Sort Code

```
def bubble_sort(arr):    n = len(arr)    for i in
range(n):        for j in range(0, n-i-1):
if arr[j] > arr[j+1]:            arr[j], arr[j+1]
= arr[j+1], arr[j]

arr = [5, 3, 8, 4]
bubble_sort(arr) print(arr)
```

Quick Sort Code

```
def quick_sort(arr):    if len(arr) <= 1:        return
arr    pivot = arr[0]    left = [x for x in arr[1:] if x
<= pivot]    right = [x for x in arr[1:] if x > pivot]

return quick_sort(left) + [pivot] + quick_sort(right)

print(quick_sort([5, 3, 8, 4]))
```

Sample Output

```
[3, 4, 5, 8]
```

Task 5: Transparency in AI Recommendations

Objective: Recommend products with clear explanations.

Code

```
def recommend_products():    products = {
"Laptop": "Recommended for programming and work",
    "Headphones": "Recommended for online classes and music",
    "Notebook": "Recommended for study notes"
}
```

```
    for product, reason in products.items():
print(product, "-", reason) recommend_products()
```

Sample Output

```
Laptop - Recommended for programming and work

Headphones - Recommended for online classes and music

Notebook - Recommended for study notes
```

Conclusion

This experiment demonstrates the importance of ethical responsibility in AI-assisted programming. Developers must ensure security, privacy, and transparency while using AI tools to build reliable and trustworthy software systems.