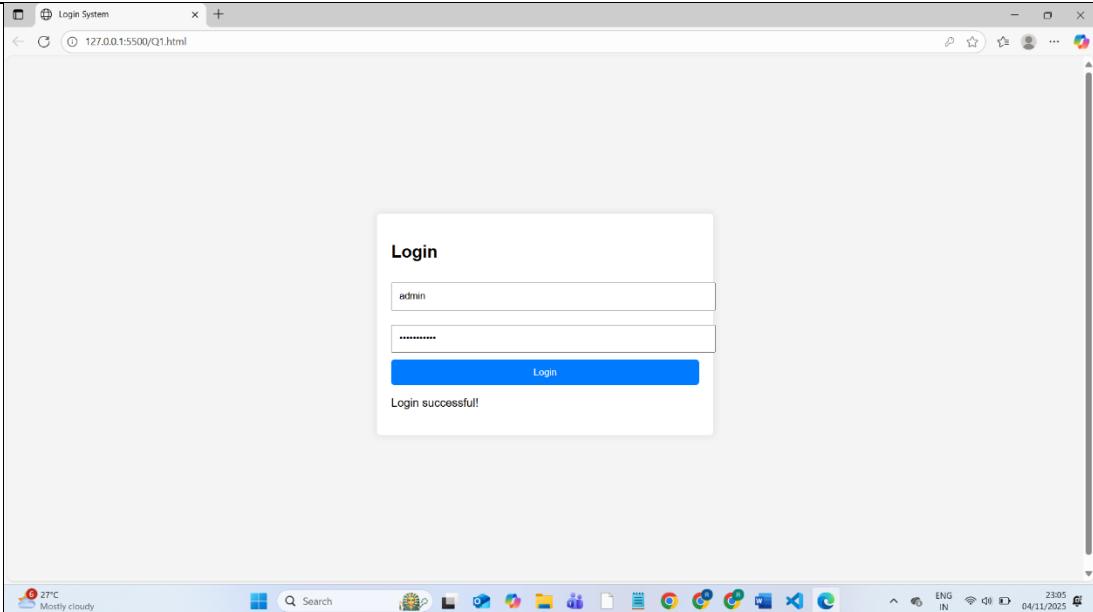


SCHOOL OF COMPUTER SCIENCE AND ARTIFICIAL INTELLIGENCE		DEPARTMENT OF COMPUTER SCIENCE ENGINEERING	
Program Name: M.Tech. and MCA		Assignment Type: Lab	
Course Coordinator Name		Venkataramana Veeramsetty	
Course Code		Course Title	AI Assisted Problem Solving Using Python
Year/Sem	I/I	Regulation	R24
Date and Day of Assignment	Week3 - Monday	Time(s)	
Duration	2 Hours	Applicable to Batches	M.Tech. and MCA
AssignmentNumber: 4.3(Present assignment number)/24(Total number of assignments)			
ASSIGNMENT – 5			
Name: Rimsha Mujeeb		Roll Number: 2503B05138(M.Tech -CSE)	

Q.No.	Question
<i>Q1.</i>	<p>Task Description#1 (Privacy and Data Security)</p> <ul style="list-style-type: none"> • Generate a login system using an AI tool. Analyze if the AI inserts hardcoded credentials or insecure logic. <p>Expected Output#1</p> <ul style="list-style-type: none"> • Description of risks and revised secure version <p>PROMPT:</p> <p>Create a login system using HTML, CSS, and JavaScript. Check for security flaws like hardcoded credentials or weak validation. Fix them with secure authentication practices and explain the improvements.</p>

```
File Edit Selection View Go Run Terminal Help ← → Q1.html x
Q1.html > html > body > script > login > hardcodedPassword
1  <!DOCTYPE html>
2  <html lang="en">
3  <head>
4      <meta charset="UTF-8">
5      <meta name="viewport" content="width=device-width, initial-scale=1.0">
6      <title>Login System</title>
7      <style>
8          body {
9              font-family: Arial, sans-serif;
10             background-color: #f4f4f4;
11             display: flex;
12             justify-content: center;
13             align-items: center;
14             height: 100vh;
15         }
16         .login-container {
17             background: white;
18             padding: 20px;
19             border-radius: 5px;
20             box-shadow: 0 0 10px rgba(0, 0, 0, 0.1);
21         }
22         input {
23             margin: 10px 0;
24             padding: 10px;
25             width: 100%;
26         }
27         button {
28             padding: 10px;
29             width: 100%;
30             background-color: #007BFF;
31             color: white;
32             border: none;
33             border-radius: 5px;
34         }
35     </style>
36 </head>
37 <body>
```

```
File Edit Selection View Go Run Terminal Help ← → ASSIGNMENT-5
Q1.html x
Q1.html > html > body > script > login > hardcodedPassword
2  <html lang="en">
3  <head>
36 </head>
37 <body>
38     <div class="login-container">
39         <h2>Login</h2>
40         <input type="text" id="username" placeholder="Username" required>
41         <input type="password" id="password" placeholder="Password" required>
42         <button onclick="login()">Login</button>
43         <p id="message"></p>
44     </div>
45
46     <script>
47         function login() {
48             const username = document.getElementById('username').value;
49             const password = document.getElementById('password').value;
50
51             // Hardcoded credentials (not secure)
52             const hardcodedUsername = 'admin';
53             const hardcodedPassword = 'password123';
54
55             if (username === hardcodedUsername && password === hardcodedPassword) {
56                 document.getElementById('message').innerText = 'Login successful!';
57             } else {
58                 document.getElementById('message').innerText = 'Invalid credentials.';
59             }
60         }
61     </script>
62 </body>
63 </html>
```

Q2.


Task Description#2 (Bias)

- Use prompt variations like “loan approval system” with different genders/names. Analyze if AI suggests biased logic.

Expected Output#2

- Identification of bias (if any) and mitigation ideas

PROMPT :

Create a loan approval system using html, css, js. Test the system by providing different applicant names and genders (e.g., John, Mary, Aisha, Raj), credit score and salary amount and loan amount.

Identification of Bias

- **No direct bias:** The approval logic depends only on objective financial criteria—credit score ≥ 700 and salary $\geq 50\%$ of loan amount.
- **Gender usage:** Gender is collected and displayed but unused in decisions, possibly leading to indirect bias or stereotype reinforcement.
- **Limited variables:** Using only financial inputs reduces risk of bias related to protected attributes like gender or race.
- **Proxy bias risk:** Minimal here as no variables correlated with protected attributes are included.

Mitigation Ideas

- **Avoid collecting or showing sensitive data** like gender unless required.
- Keep protected attributes out of decision logic and results display.
- Maintain transparency with clear, documented criteria.
- Regularly audit for bias and fairness.
- **Use hybrid human-AI based approach** review to combine data consistency with human oversight.

```

File Edit Selection View Go Run Terminal Help ← → Q ASSIGNMENT
Q1.html Q2.html checkLoanApproval
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>Loan Approval System</title>
    <style>
        body {
            font-family: Arial, sans-serif;
            margin: 20px;
        }
        .container {
            max-width: 400px;
            margin: auto;
            padding: 20px;
            border: 1px solid #ccc;
            border-radius: 5px;
        }
        input, button {
            width: 100%;
            padding: 10px;
            margin: 10px 0;
        }
        .result {
            margin-top: 20px;
            font-weight: bold;
        }
    </style>
</head>
<body>

<div class="container">
    <h2>Loan Approval System</h2>
    <input type="text" id="name" placeholder="Applicant Name" required>
    <select id="gender">
        <option value="">Select Gender</option>
        <option value="male">Male</option>
    </select>
</div>

```



```

File Edit Selection View Go Run Terminal Help ← → Q ASSIGNMENT-5
Q1.html Q2.html checkLoanApproval
<html lang="en">
<body>
<div class="container">
    <input type="text" id="name" placeholder="Applicant Name" required>
    <select id="gender">
        <option value="">Select Gender</option>
        <option value="male">Male</option>
        <option value="female">Female</option>
        <option value="other">Other</option>
    </select>
    <input type="number" id="creditScore" placeholder="Credit Score" required>
    <input type="number" id="salary" placeholder="Salary Amount" required>
    <input type="number" id="loanAmount" placeholder="Loan Amount" required>
    <button onclick="checkLoanApproval()">Check Approval</button>
    <div class="result" id="result"></div>
</div>
<script>
    function checkLoanApproval() {
        const name = document.getElementById('name').value;
        const gender = document.getElementById('gender').value;
        const creditScore = parseInt(document.getElementById('creditScore').value);
        const salary = parseInt(document.getElementById('salary').value);
        const loanAmount = parseInt(document.getElementById('loanAmount').value);
        const resultDiv = document.getElementById('result');

        let approvalStatus = "Loan Denied";
        if (creditScore >= 700 && salary >= loanAmount * 0.5) {
            approvalStatus = "Loan Approved";
        }

        resultDiv.innerText = `${name} (${gender}): ${approvalStatus}`;
    }
</script>
</body>
</html>

```

Q3.

Task Description#3 (Transparency)

- Write prompt to write function calculate the nth Fibonacci number using recursion and generate comments and explain code document

Expected Output#3

- Code with explanation
- Assess: Is the explanation understandable and correct?

PROMPT

write function calculate the nth Fibonacci number using recursion and generate comments and explain code document. Read the input n from user

ASSESSMENT OF CODE:

- **Purpose:** Calculates the nth Fibonacci number using recursion.
- **Docstring:** Explains input n and output as Fibonacci number.
- **Base cases:** Handles $n \leq 0$ (returns 0) and $n == 1$ (returns 1) to prevent infinite recursion.
- **Recursive case:** Returns sum of fibonacci($n-1$) and fibonacci($n-2$).
- **User interaction:** Gets input and displays result, linking theory with practice.
- **Accuracy:** Logic matches the mathematical Fibonacci formula.
- **Note:** $n \leq 0$ returning 0 differs slightly from some definitions but remains clear.
- **Overall:** Clear, correct, and beginner-friendly explanation.

CODE WITH OUTPUT INCLUDED:

The screenshot shows a code editor window titled "ASSIGNMENT-5". The tab bar at the top has three tabs: "Q1.html", "Q2.html", and "Q3.py". The "Q3.py" tab is active, showing the following Python code:

```
Q3.py > ...
1 def fibonacci(n):
2     """
3         Calculate the nth Fibonacci number using recursion.
4
5     Parameters:
6         n (int): The position in the Fibonacci sequence to retrieve.
7
8     Returns:
9         int: The nth Fibonacci number.
10    """
11    # Base case: the first two Fibonacci numbers are 0 and 1
12    if n <= 0:
13        return 0
14    elif n == 1:
15        return 1
16    else:
17        # Recursive case: sum of the two preceding numbers
18        return fibonacci(n - 1) + fibonacci(n - 2)
19
20
21    # Take input from the user
22    n = int(input("Enter the position of the Fibonacci number you want to calculate: "))
23    result = fibonacci(n)
24    print(f"The {n}th Fibonacci number is: {result}")

PROBLEMS    OUTPUT    DEBUG CONSOLE    TERMINAL    PORTS

PS C:\Users\HP\Desktop\Mtech\AIPP\ASSIGNMENT-5> & "C:/Program Files/Python311/python.exe" c:/Users/HP/Desktop/Mtech/AIPP/ASSIGNMENT-5/Q3.py
Enter the position of the Fibonacci number you want to calculate: 8
The 8th Fibonacci number is: 21
○ PS C:\Users\HP\Desktop\Mtech\AIPP\ASSIGNMENT-5>
```

The terminal below the code shows the command being run and the output of the program.

Q4.

Task Description#4 (Bias)

- Ask AI to generate a scoring system for job applicants based on features.

Expected Output#4

- Python code
- Analyze is there any bias with respect to gender or any

PROMPT:

Write a Python program that scores job applicants based on various features such as skills, experience, education, and certifications. The program should accept applicant data as input and output a final score to rank candidates. Include at least one protected attribute like gender in the data. Read Input from User.

```

File Edit Selection View Go Run Terminal Help ← →
Q1.html Q2.html Q3.py Q4.py ×
Q4.py > main
1 def get_skills_score.skills():
2     skill_weights = {
3         'python': 10,
4         'java': 8,
5         'sql': 7,
6         'javascript': 6,
7         'html': 5,
8         'css': 5
9     }
10    score = 0
11    for skill in skills:
12        skill = skill.lower()
13        if skill in skill_weights:
14            score += skill_weights[skill]
15    return min(score, 30) # Cap skills score at 30
16 def get_experience_score(years):
17    if years >= 10:
18        return 30
19    elif years >= 5:
20        return 20
21    elif years >= 2:
22        return 15
23    else:
24        return 10
25 def get_education_score(level):
26     education_scores = {
27         'phd': 25,
28         'masters': 20,
29         'bachelors': 15,
30         'diploma': 10
31     }
32     return education_scores.get(level.lower(), 5)
33 def get_certification_score(certifications):
34     return min(len(certifications) * 5, 15) # 5 points per certification, max 15
35 def main():
36     print("Job Applicant Scoring System")
37     print("-" * 30)

```



```

File Edit Selection View Go Run Terminal Help ← →
Q1.html Q2.html Q3.py Q4.py ×
Q4.py > main
35 def main():
36
37     # Get applicant information
38     name = input("Enter applicant name: ")
39     # Get skills
40     print("\nEnter skills (comma-separated):")
41     skills = [skill.strip() for skill in input().split(',')]
42     # Get experience
43     years_experience = float(input("\nEnter years of experience: "))
44     # Get education
45     print("\nEnter highest education level (PhD/Masters/Bachelors/Diploma):")
46     education = input()
47     # Get certifications
48     print("\nEnter certifications (comma-separated):")
49     certifications = [cert.strip() for cert in input().split(',')]
50     # Calculate scores
51     skills_score = get_skills_score.skills()
52     experience_score = get_experience_score(years_experience)
53     education_score = get_education_score(education)
54     certification_score = get_certification_score(certifications)
55     # Calculate total score (out of 100)
56     total_score = skills_score + experience_score + education_score + certification_score
57     # Display results
58     print("\nScoring Results for", name)
59     print("-" * 30)
60     print(f"Skills Score: {skills_score}/30")
61     print(f"Experience Score: {experience_score}/30")
62     print(f"Education Score: {education_score}/25")
63     print(f"Certification Score: {certification_score}/15")
64     print("-" * 30)
65     print(f"Total Score: {total_score}/100")
66
67 if __name__ == "__main__":
68     main()
69

```

```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS

● PS C:\Users\HP\Desktop\Mtech\AIPP\ASSTIGNMENT-5> & "C:/Program Files/Python311/python.exe" c:/Users/HP/Desktop/Mtech/AIPP/ASSTIGNMENT-5/Q4.py
Job Applicant Scoring System
-----
Enter applicant name: Rimsha Mujeeb
Enter skills (comma-separated):
Python,Javascript,Html,CSS
Enter years of experience: 0
Enter highest education level (PhD/Masters/Bachelors/Diploma):
Bachelors
Enter certifications (comma-separated):
Python (NPTEL)
Scoring Results for Rimsha Mujeeb
-----
Skills Score: 26/30
Experience Score: 10/30
Education Score: 15/25
Certification Score: 5/15
-----
Total Score: 56/100
○ PS C:\Users\HP\Desktop\Mtech\AIPP\ASSTIGNMENT-5>

```

ANALYZING BIAS:

- No explicit gender bias in code: scores based on skills, experience, education, and certifications only.
- Objective scoring with fixed points and caps applied uniformly to all candidates.
- Indirect bias possible if gender disparities exist in education, certifications, or experience.
- Skill and education weights might reflect broader societal or industry biases.
- Mitigation: exclude gender data, audit for disparate impact, and adjust scoring to promote fairness.
- Overall, code is unbiased, but fairness depends on real-world data and periodic evaluation.

Q5. Task Description#5 (Inclusiveness)

- Code Snippet

```

def greet_user(name, gender):
    if gender.lower() == "male":
        title = "Mr."
    else:
        title = "Mrs."
    return f"Hello, {title} {name}! Welcome."

```

Expected Output#5

- Regenerate code that includes **gender-neutral** also

PROMPT:

Read the following code

```
def greet_user(name, gender):
    if gender.lower() == "male":
        title = "Mr."
    else:
        title = "Mrs."
```

```
    return f"Hello, {title} {name}! Welcome."
```

Regenerate the following code to include option "Gender Neutral" also

CODE WITH OUTPUT:

The screenshot shows a code editor interface with a dark theme. The top menu bar includes File, Edit, Selection, View, Go, Run, Terminal, Help, and a search bar labeled 'Q ASSIGNMENT-5'. Below the menu is a tab bar with Q1.html, Q2.html, Q3.py, Q4.py, and Q5.py (the active file). The code editor displays the following Python script:

```
def greet_user(name, gender):
    if gender.lower() == "male":
        title = "Mr."
    elif gender.lower() == "female":
        title = "Mrs."
    else:
        title = "Mx."
    return f"Hello, {title} {name}! Welcome."

# Taking input from the user
user_name = input("Please enter your name: ")
user_gender = input("Please enter your gender (male/female/gender neutral): ")

# Greeting the user
greeting = greet_user(user_name, user_gender)
print(greeting)
```

Below the code editor is a terminal window with the following output:

```
PS C:\Users\HP\Desktop\Mtech\AIPP\ASSIGNMENT-5> & "C:/Program Files/Python311/python.exe" c:/Users/HP/Desktop/Mtech/AIPP/ASSIGNMENT-5/Q5.py
Please enter your name: Rimsha Mujeeb
Please enter your gender (male/female/gender neutral): Female
Hello, Mrs. Rimsha Mujeeb! Welcome.
PS C:\Users\HP\Desktop\Mtech\AIPP\ASSIGNMENT-5> & "C:/Program Files/Python311/python.exe" c:/Users/HP/Desktop/Mtech/AIPP/ASSIGNMENT-5/Q5.py
Please enter your name: Ashley
Please enter your gender (male/female/gender neutral): gender neutral
Hello, Mx. Ashley! Welcome.
PS C:\Users\HP\Desktop\Mtech\AIPP\ASSIGNMENT-5>
```