

Name:I. Abhinay Powar

H.No:2303A51811

Batch:26

SCHOOL OF COMPUTER SCIENCE AND ARTIFICIAL INTELLIGENCE		DEPARTMENT OF COMPUTER SCIENCE ENGINEERING																		
Program Name: B. Tech		Assignment Type: Lab	Academic Year: 2025-2026																	
Course Coordinator Name		Dr. Rishabh Mittal																		
Instructor(s) Name		<table border="1"> <tr><td>Mr. S Naresh Kumar</td></tr> <tr><td>Ms. B. Swathi</td></tr> <tr><td>Dr. Sasanko Shekhar Gantayat</td></tr> <tr><td>Mr. Md Sallauddin</td></tr> <tr><td>Dr. Mathivanan</td></tr> <tr><td>Mr. Y Srikanth</td></tr> <tr><td>Ms. N Shilpa</td></tr> <tr><td>Dr. Rishabh Mittal (Coordinator)</td></tr> <tr><td>Dr. R. Prashant Kumar</td></tr> <tr><td>Mr. Ankushavali MD</td></tr> <tr><td>Mr. B Viswanath</td></tr> <tr><td>Ms. Sujitha Reddy</td></tr> <tr><td>Ms. A. Anitha</td></tr> <tr><td>Ms. M.Madhuri</td></tr> <tr><td>Ms. Katherashala Swetha</td></tr> <tr><td>Ms. Velpula sumalatha</td></tr> <tr><td>Mr. Bingi Raju</td></tr> </table>		Mr. S Naresh Kumar	Ms. B. Swathi	Dr. Sasanko Shekhar Gantayat	Mr. Md Sallauddin	Dr. Mathivanan	Mr. Y Srikanth	Ms. N Shilpa	Dr. Rishabh Mittal (Coordinator)	Dr. R. Prashant Kumar	Mr. Ankushavali MD	Mr. B Viswanath	Ms. Sujitha Reddy	Ms. A. Anitha	Ms. M.Madhuri	Ms. Katherashala Swetha	Ms. Velpula sumalatha	Mr. Bingi Raju
Mr. S Naresh Kumar																				
Ms. B. Swathi																				
Dr. Sasanko Shekhar Gantayat																				
Mr. Md Sallauddin																				
Dr. Mathivanan																				
Mr. Y Srikanth																				
Ms. N Shilpa																				
Dr. Rishabh Mittal (Coordinator)																				
Dr. R. Prashant Kumar																				
Mr. Ankushavali MD																				
Mr. B Viswanath																				
Ms. Sujitha Reddy																				
Ms. A. Anitha																				
Ms. M.Madhuri																				
Ms. Katherashala Swetha																				
Ms. Velpula sumalatha																				
Mr. Bingi Raju																				
Course Code	23CS002PC304	Course Title	AI Assisted Coding																	
Year/Sem	III/II	Regulation	R23																	
Date and Day of Assignment	Week3 – Wednesday	Time(s)	23CSBTB01 To 23CSBTB52																	
Duration	2 Hours	Applicable to Batches	All batches																	
AssignmentNumber: 6.3(Present assignment number)/24(Total number of assignments)																				
Q.No.	Question	Expected Time to complete																		
1	Lab 6: AI-Based Code Completion – Classes, Loops, and Conditionals Lab Objectives <ul style="list-style-type: none"> • To explore AI-powered auto-completion features for core Python constructs such as classes, loops, and conditional statements. • To analyze how AI tools suggest logic for object-oriented programming and control structures. • To evaluate the correctness, readability, and completeness of AI-generated Python code. 	Week3 - Wednesday																		

	Lab Outcomes (LOs) After completing this lab, students will be able to: <ul style="list-style-type: none">• Use AI tools to generate and complete Python class definitions and methods.• Understand and assess AI-suggested loop constructs for iterative tasks.• Generate and evaluate conditional statements using AI-driven prompts.• Critically analyze AI-assisted code for correctness, clarity, and efficiency.	
	Task Description #1: Classes (Student Class) Scenario You are developing a simple student information management module. Task <ul style="list-style-type: none">• Use an AI tool (GitHub Copilot / Cursor AI / Gemini) to complete a Student class.• The class should include attributes such as name, roll number, and branch.• Add a method <code>display_details()</code> to print student information.• Execute the code and verify the output.• Analyze the code generated by the AI tool for correctness and clarity. Expected Output #1 <ul style="list-style-type: none">• A Python class with a constructor (<code>__init__</code>) and a <code>display_details()</code> method.• Sample object creation and output displayed on the console.• Brief analysis of AI-generated code.	

	<div><div><div><div>JS resume devopsjs</div><div>Tomato.html</div><div>lab 3 devops.html</div><div>DOM.html</div></div><div><div>C: > Users > shash > Downloads > counter-app > AAC A 6.3.py > ...</div><div><div>1class Student:</div><div>2def __init__(self, name, roll_number, branch):</div><div>3self.name = name</div><div>4self.roll_number = roll_number</div><div>5self.branch = branch</div><div>6</div><div>7def display_details(self):</div><div>8print(f"Name: {self.name}")</div><div>9print(f"Roll Number: {self.roll_number}")</div><div>10print(f"Branch: {self.branch}")</div><div>11</div><div>12if __name__ == "__main__":</div><div>13s = Student("Alice Smith", "CS2026", "Computer Science")</div><div>14s.display_details()</div></div></div><div><div>PROBLEMS</div><div>OUTPUT</div><div>DEBUG CONSOLE</div><div>TERMINAL</div><div>PORTS</div></div><div><div>conda : The term 'conda' is not recognized as the name of a cmdlet, func</div><div>+ conda activate Shashidhar</div><div>+ ~~~~~</div><div>+ CategoryInfo : ObjectNotFound: (conda:String) [], Command</div><div>+ FullyQualifiedErrorId : CommandNotFoundException</div><div>PS C:\Users\shash\Downloads\counter-app> & 'c:\Users\shash\anaconda3\em</div><div>\.vscode\extensions\ms-python.debugpy-2025.18.0-win32-x64\bundled\libs\d</div><div>hash\Downloads\counter-app\AAC A 6.3.py'</div><div>● Name: Alice Smith</div><div>Roll Number: CS2026</div><div>Branch: Computer Science</div><div>○ PS C:\Users\shash\Downloads\counter-app> </div></div></div></div>	
	<div><div><div>Task Description #2: Loops (Multiples of a Number)</div><div>Scenario</div><div>You are writing a utility function to display multiples of a given number.</div><div>Task</div><div><ul style="list-style-type: none">• Prompt the AI tool to generate a function that prints the first 10 multiples of a given number using a loop.• Analyze the generated loop logic.• Ask the AI to generate the same functionality using another controlled looping structure (e.g., while instead of for).</div><div>Expected Output #2</div><div><ul style="list-style-type: none">• Correct loop-based Python implementation.• Output showing the first 10 multiples of a number.• Comparison and analysis of different looping approaches.</div></div></div>	

	<div><div>JS resume devops.js<> Tomato .html<> lab 3 devops.t</div><div>C: > Users > shash > Downloads > counter-app > AAC A 6.3.py ></div><div><pre>1 def print_multiples_for(n): 2 for i in range(1, 11): 3 print(n * i) 4 5 def print_multiples_while(n): 6 i = 1 7 while i <= 10: 8 print(n * i) 9 i += 1 10 11 if __name__ == "__main__": 12 number = 7 13 print("For loop multiples:") 14 print_multiples_for(number) 15 print("While loop multiples:") 16 print_multiples_while(number)</pre></div><div>PROBLEMSOUTPUTDEBUG CONSOLETERMINALPORTS</div><div>PS C:\Users\shash\Downloads\counter-app> c:; cd 'c:da3\envs\Shashidhar\python.exe' 'c:\Users\shash\.vscode\libs\debugpy\launcher' '56132' '--' 'c:\Users\shash\ For loop multiples: 28 35 42 49 56 63 70 PS C:\Users\shash\Downloads\counter-app> </div></div>	
	<p>Task Description #3: Conditional Statements (Age Classification)</p> <p>Scenario You are building a basic classification system based on age.</p> <p>Task</p> <ul style="list-style-type: none">• Ask the AI tool to generate nested if-elif-else conditional statements to classify age groups (e.g., child, teenager, adult, senior).• Analyze the generated conditions and logic.• Ask the AI to generate the same classification using alternative conditional structures (e.g., simplified conditions or dictionary-based logic). <p>Expected Output #3</p> <ul style="list-style-type: none">• A Python function that classifies age into appropriate groups.• Clear and correct conditional logic.• Explanation of how the conditions work.	

```
C:\Users\shash\Downloads\counter-app> Python3.py ...
1  def classify_age_nested(age):
2      if age < 0:
3          return "invalid"
4      if age <= 12:
5          return "child"
6      elif age <= 17:
7          return "teenager"
8      elif age <= 64:
9          return "adult"
10     else:
11         return "senior"
12
13  def classify_age_simplified(age):
14      if age < 0:
15          return "invalid"
16      if 0 <= age <= 12:
17          return "child"
18      if 13 <= age <= 17:
19          return "teenager"
20      if 18 <= age <= 64:
21          return "adult"
22      return "senior"
23
24  def classify_age_dict(age):
25      if age < 0:
26          return "invalid"
27      thresholds = [(12, "child"), (17, "teenager"), (64, "adult"), (float('inf'), "senior")]
28      for limit, label in thresholds:
29          if age <= limit:
30              return label
31
32  if __name__ == "__main__":
33
PROBLEMS  OUTPUT  DEBUG CONSOLE  TERMINAL  PORTS
PS C:\Users\shash\Downloads\counter-app> c:: cd 'c:\Users\shash\Downloads\counter-app'; & 'c:\Users\shash\Downloads\counter-app\Python3.py'
shidhar\python.exe' 'c:\Users\shash\.vscode\extensions\ms-python.debugpy-2025.18.0-win32-x64\bundle\libs\python\python.exe' '-.' 'c:\Users\shash\Downloads\counter-app\AAC A 6.3.py'
Nested if-elif-else:
-1 invalid
Dictionary-threshold approach:
3 child
15 teenager
30 adult
70 senior
-1 invalid
PS C:\Users\shash\Downloads\counter-app> 
```

	<div><div>C: > Users > shash > Downloads > counter-app > AAC A 6.3.py > ...</div><div><pre>13 def classify_age_simplified(age): 20 if 18 <= age <= 64: 21 return "adult" 22 return "senior" 23 24 def classify_age_dict(age): 25 if age < 0: 26 return "invalid" 27 thresholds = [(12, "child"), (17, "teenager"), (64, "adult"), (float('inf'), "senior")] 28 for limit, label in thresholds: 29 if age <= limit: 30 return label 31 32 if __name__ == "__main__": 33 sample_ages = [3, 15, 30, 70, -1] 34 print("Nested if-elif-else:") 35 for a in sample_ages: 36 print(a, classify_age_nested(a)) 37 print("Simplified chained conditions:") 38 for a in sample_ages: 39 print(a, classify_age_simplified(a)) 40 print("Dictionary-threshold approach:") 41 for a in sample_ages: 42 print(a, classify_age_dict(a))</pre></div><div>PROBLEMS OUTPUT DEBUG CONSOLE <u>TERMINAL</u> PORTS</div><div><pre>PS C:\Users\shash\Downloads\counter-app> c::; cd 'c:\Users\shash\Downloads\counter-app'; & 'c:\Us shidhar\python.exe' 'c:\Users\shash\.vscode\extensions\ms-python.debugpy-2025.18.0-win32-x64\bund 49215' '--' 'c:\Users\shash\Downloads\counter-app\AAC A 6.3.py' Nested if-elif-else: -1 invalid Dictionary-threshold approach: 3 child 15 teenager 30 adult 70 senior -1 invalid PS C:\Users\shash\Downloads\counter-app> █</pre></div></div>	
	<div><div>Task Description #4: For and While Loops (Sum of First n Numbers)</div><div>Scenario You need to calculate the sum of the first n natural numbers.</div><div>Task<ul style="list-style-type: none">• Use AI assistance to generate a sum_to_n() function using a for loop.• Analyze the generated code.• Ask the AI to suggest an alternative implementation using a while loop or a mathematical formula.</div><div>Expected Output #4<ul style="list-style-type: none">• Python function to compute the sum of first n numbers.• Correct output for sample inputs.• Explanation and comparison of different approaches.</div></div>	

	<pre>C: > Users > shash > Downloads > counter-app > AAC A 6.3.py > ... 1 def sum_to_n_for(n): 2 total = 0 3 for i in range(1, n + 1): 4 total += i 5 return total 6 7 def sum_to_n_while(n): 8 total = 0 9 i = 1 10 while i <= n: 11 total += i 12 i += 1 13 return total 14 15 def sum_to_n_formula(n): 16 if n < 0: 17 return None 18 return n * (n + 1) // 2 19 20 if __name__ == "__main__": 21 samples = [0, 1, 10, 100] 22 for n in samples: 23 print(n, sum_to_n_for(n), sum_to_n_while(n), sum_to_n_formula(n))</pre> <p>PROBLEMS OUTPUT DEBUG CONSOLE <u>TERMINAL</u> PORTS</p> <pre>PS C:\Users\shash\Downloads\counter-app> c::; cd 'c:\Users\shash\Downloads\counter-app' da3\envs\Shashidhar\python.exe' 'c:\Users\shash\.vscode\extensions\ms-python.debugpy\launcher' '65013' '--' 'c:\Users\shash\Downloads\counter-app\AAC A 0 0 0 0 PS C:\Users\shash\Downloads\counter-app> c::; cd 'c:\Users\shash\Downloads\counter-app' da3\envs\Shashidhar\python.exe' 'c:\Users\shash\.vscode\extensions\ms-python.debugpy\launcher' '61058' '--' 'c:\Users\shash\Downloads\counter-app\AAC A 0 0 0 0 1 1 1 1 10 55 55 55 100 5050 5050 5050 PS C:\Users\shash\Downloads\counter-app> █</pre>	
	<p>Task Description #5: Classes (Bank Account Class)</p> <p>Scenario You are designing a basic banking application.</p> <p>Task</p> <ul style="list-style-type: none">• Use AI tools to generate a Bank Account class with methods such as deposit(), withdraw(), and check_balance().• Analyze the AI-generated class structure and logic.• Add meaningful comments and explain the working of the code. <p>Expected Output #5</p> <ul style="list-style-type: none">• Complete Python Bank Account class.• Demonstration of deposit and withdrawal operations with updated balance.• Well-commented code with a clear explanation.	

```
ab 3 devops.html  DUM.html  CP A 4.7.java  AAC A 6.3.py  DUM.css
C: > Users > shash > Downloads > counter-app > AAC A 6.3.py > BankAccount > withdraw

1 class BankAccount:
2     def __init__(self, owner, balance=0.0):
3         self.owner = owner
4         self.balance = float(balance)
5     def deposit(self, amount):
6         if amount <= 0:
7             raise ValueError("Deposit amount must be positive")
8         self.balance += amount
9         return self.balance
10    def withdraw(self, amount):
11        if amount <= 0:
12            raise ValueError("Withdrawal amount must be positive")
13        if amount > self.balance:
14            return False
15        self.balance -= amount
16        return True
17    def check_balance(self):
18        return self.balance
19
20    def __repr__(self):
21        return f"BankAccount(owner={self.owner!r}, balance={self.balance:.2f})"
22
23
24 if __name__ == "__main__":
25     owner = input("Enter account owner name: ").strip()
26     bal = input("Enter starting balance (leave empty for 0): ").strip()
27     try:
28         start_balance = float(bal) if bal else 0.0
29     except ValueError:
30         start_balance = 0.0
31     acct = BankAccount(owner or "Unknown", start_balance)
32     print("Account created:".acct)

PROBLEMS  OUTPUT  DEBUG CONSOLE  TERMINAL  PORTS

PS C:\Users\shash\Downloads\counter-app> c::; cd 'c:\Users\shash\Downloads\counter-app'
da3\envs\Shashidhar\python.exe' 'c:\Users\shash\.vscode\extensions\ms-python.debugpy-20
libs\debugpy\launcher' '63018' '--' 'c:\Users\shash\Downloads\counter-app\AAC A 6.3.py'
Enter account owner name: Shashidhar Ashadapu
Enter starting balance (leave empty for 0): 1000
Account created: BankAccount(owner='Shashidhar Ashadapu', balance=1000.00)

Options: [d]eposit, [w]ithdraw, [c]heck balance, [q]uit
Choose option: w
Amount to withdraw: 500
Success: True Balance: 500.0
```

Note: Report should be submitted as a word document for all tasks in a single document with prompts, comments & code explanation, and output and if required, screenshots.