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	Venkataramana Veeramsetty		
Instructor(s)Name	1. Dr. Mohammed Ali Shaik 2. Dr. T Sampath Kumar 3. Mr. S Naresh Kumar 4. Dr. V. Rajesh 5. Dr. Brij Kishore 6. Dr Pramoda Patro 7. Dr. Venkataramana 8. Dr. Ravi Chander 9. Dr. Jagjeeth Singh		
Course Code	24CS002PC215	**Course Title**	AI Assisted Coding
Year/Sem	II/I	**Regulation**	R24
Date and Day of Assignment	06-08-2025	**Time(s)**	
Duration	2 Hours	**Applicable to Batches**	
AssignmentNumber:6.5(Present assignment number)/**24**(Total number of assignments)			

Q.No.	Question	*ExpectedTime to complete*
1	**Lab 6: AI-Based Code Completion: Working with suggestions for classes, loops, conditionals** Lab Assignment 1: Intelligent Code Completion for Object-Oriented Programming **Objective:** To explore AI-powered code assistants for writing Python classes, constructors, and methods through intelligent suggestions. Suppose that you are hired as an intern at a tech company that develops inventory management systems. Your manager asks you to create a **Product** class and a **Warehouse** class with some basic methods. You have decided to use AI-powered code suggestions to help speed up development and reduce syntax errors. Tasks to be completed are as below **1. Setup AI Coding Tool:** • Install and configure GitHub Copilot or Kite with VS Code or JetBrains IDE. • Enable real-time code suggestions. **2. Class Design Using AI Assistance:** • Begin defining a Product class with attributes: name, price, quantity. • Use the AI suggestion feature to automatically complete the __init__() method. • Add a method calculate_value() to return price * quantity.	15.08.2025 EOD

Code:

```python
class Product:
    def __init__(self, name, price, quantity):
        self.name = name
        self.price = price
        self.quantity = quantity

    def calculate_value(self):
        return self.price * self.quantity

    def __str__(self):
        return f"Product(name={self.name}, price={self.price}, quantity={self.quantity})"

```

Output:

```
PROBLEMS  OUTPUT  DEBUG CONSOLE  TERMINAL  PORTS

PS C:\PROGRAMMES VSCODE\AI coding> & 'c:\Users\venkatesh\AppData\Local\Programs\Python\Python313\python.exe'
4\bundled\libs\debugpy\launcher' '49779' '--' 'c:\PROGRAMMES VSCODE\AI coding\ass6.5.py'
PS C:\PROGRAMMES VSCODE\AI coding>
```

3. Create Another Class:

- Define a Warehouse class with a list of Product objects.
- Use code completion to help implement**:**
  - A method to add a product.
  - A method to display the most valuable product.

Code:

```python
class Warehouse:
    def __init__(self):
        self.products = []

    def add_product(self, product):
        if isinstance(product, Product):
            self.products.append(product)

    def total_value(self):
        return sum(product.calculate_value() for product in self.products)

    def __str__(self):
        return f"Warehouse({[str(product) for product in self.products]})"
```

- Output:

```
ions\ms-python.debugpy-2025.10.0-win32-x64\bundled\libs\debugpy\launcher' '6049' '--' 'C:\PROGRAMMES VSCODE\AI coding\ass6.5.py'
Warehouse(['Product(name=Laptop, price=1000, quantity=5)', 'Product(name=Phone, price=500, quantity=10)'])
PS C:\PROGRAMMES VSCODE\AI coding>
```

4. Reflection:

- Identify how much of the code was completed by AI and what manual edits were needed.
- Comment on the relevance and accuracy of AI suggestions.

Comment:

- The AI suggestions were relevant and accurate in building a basic inventory management system. It correctly identified the need for Product and Warehouse classes and implemented methods for adding products and calculating total value. The code is well-structured and follows object-oriented principles.

Requirements:

- VS Code with Github Copilot or Cursor API and/or Google Colab with Gemini

	Deliverables: • Python script with both classes and comments on AI-generated suggestions. • Short report (1 page) summarizing your experience with AI code completion. • . Summarise; • The AI has built a simple inventory management system with Product and Warehouse classes. • It allows adding products to the warehouse and calculating their total value. • • The code demonstrates basic object-oriented programming principles in Python. • It showcases how to model real-world entities (products and a warehouse) using classes. • The implementation is straightforward, making it easy to understand and extend. • how much of the code was completed by AI and what manual edits were needed. •	