

SCHOOL OF COMPUTER SCIENCE and ARTIFICIAL INTELLIGENCE

**Program :**B.tech(CSE)

**Specialization :**AIML

**Course Title :**AI Assisted Coding

**Course Code :**24CS002PC215

**Semester :**3rd semester

**Academic Session :**2025-2026

**Name of Student :** Kaveti Manohar

**Enrollment No. :**2403A52079

**Batch No. :**02

**Date :0**8/10/2025

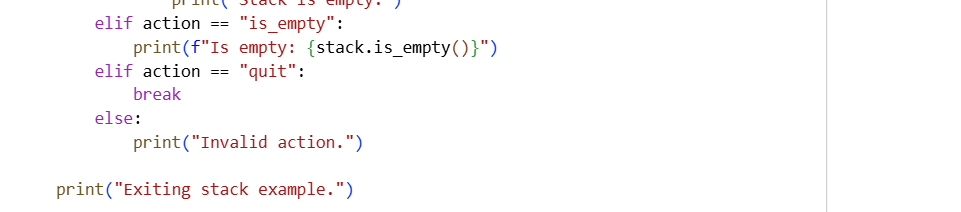
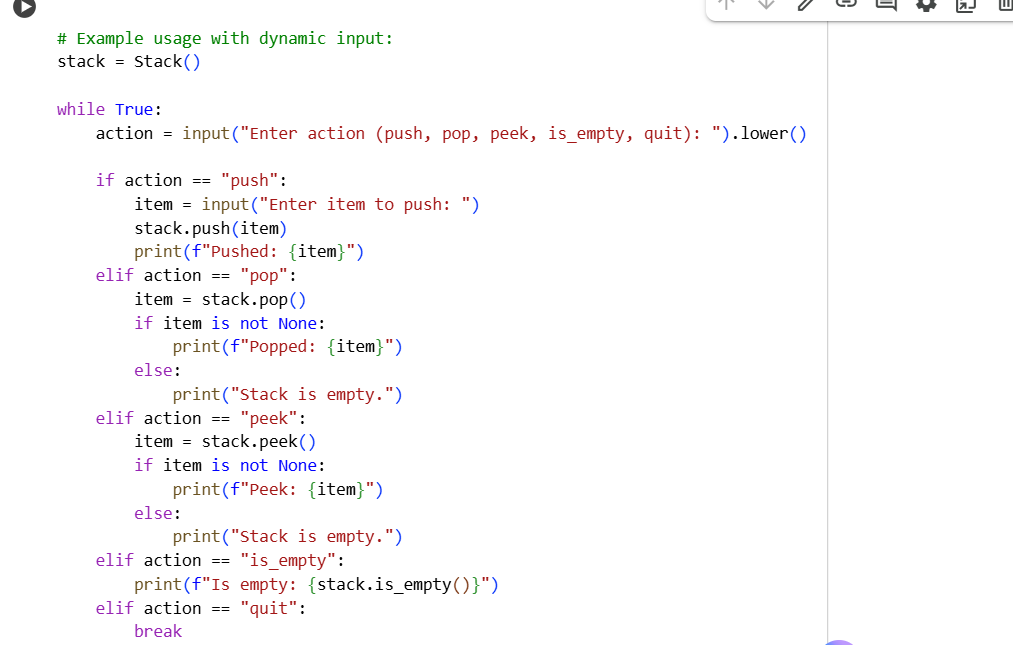
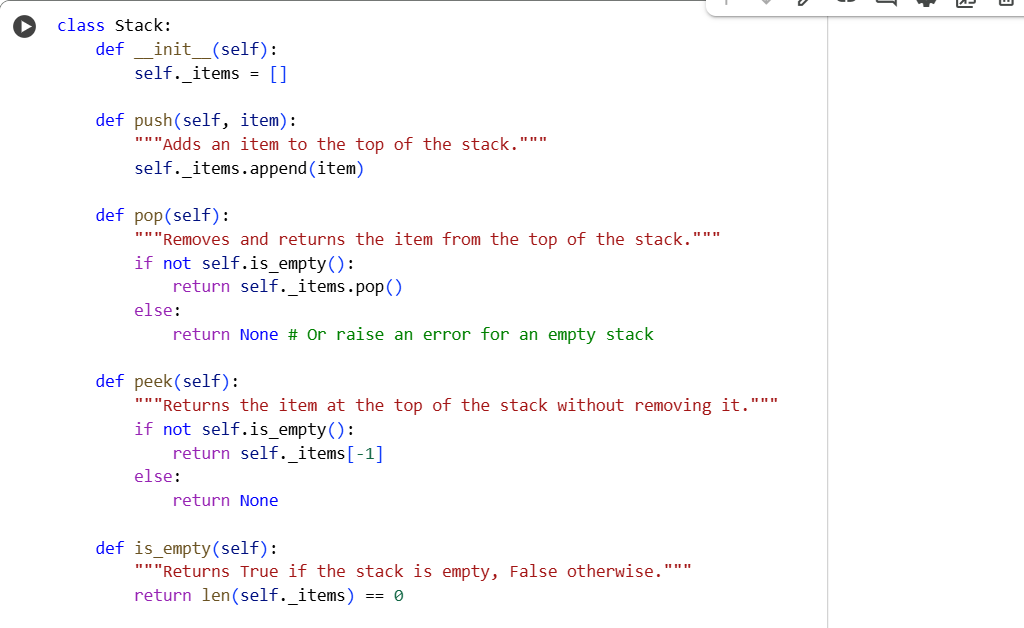
#LAB ASSIGNMENT

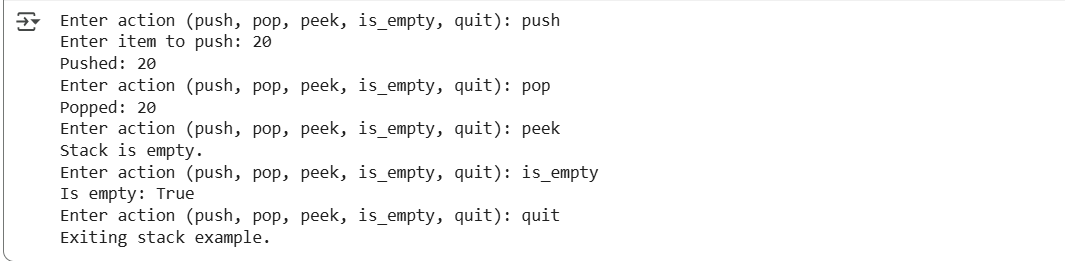
* TASK DESCRIPTION-1:

Stack class implementation  
Task: Ask AI to implement a stack class with push(), pop(), peek() and is\_empty() methods.

* PROMPT:

Create a Python class Stack that implements a stack with methods push(item), pop(), peek(), and is\_empty() with dynamic input.

* CODE: 
* OUTPUT:



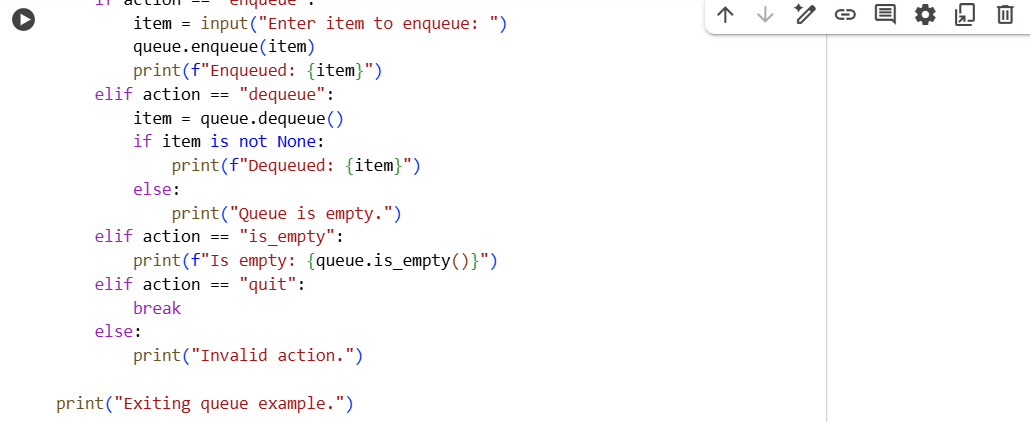
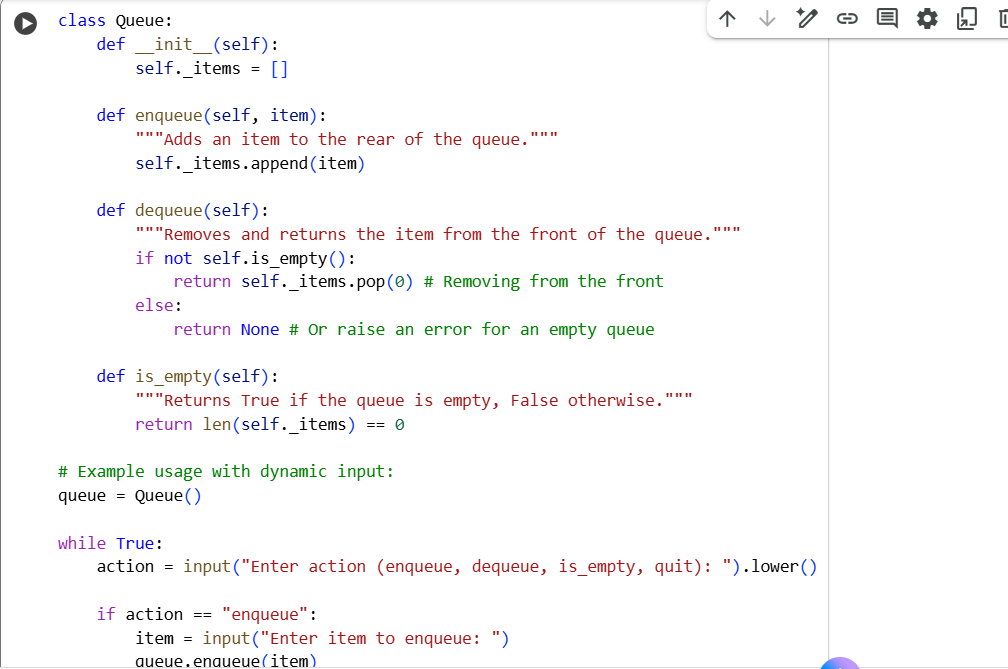
* TASK DESCRIPTION-2:

Queue Implementation  
Task: Use AI to generate a Queue class with enqueue(), dequeue(), and is\_empty().

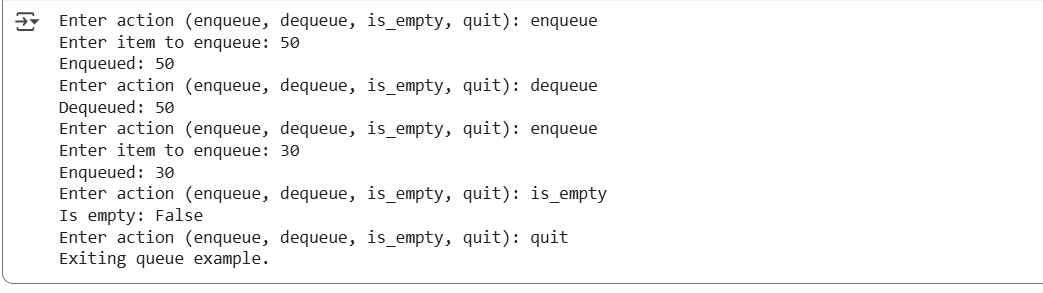
* PROMPT:

Create a Python class Queue that implements a queue with methods enqueue(item), dequeue(), and is\_empty() with user input.

* CODE:



* OUTPUT:

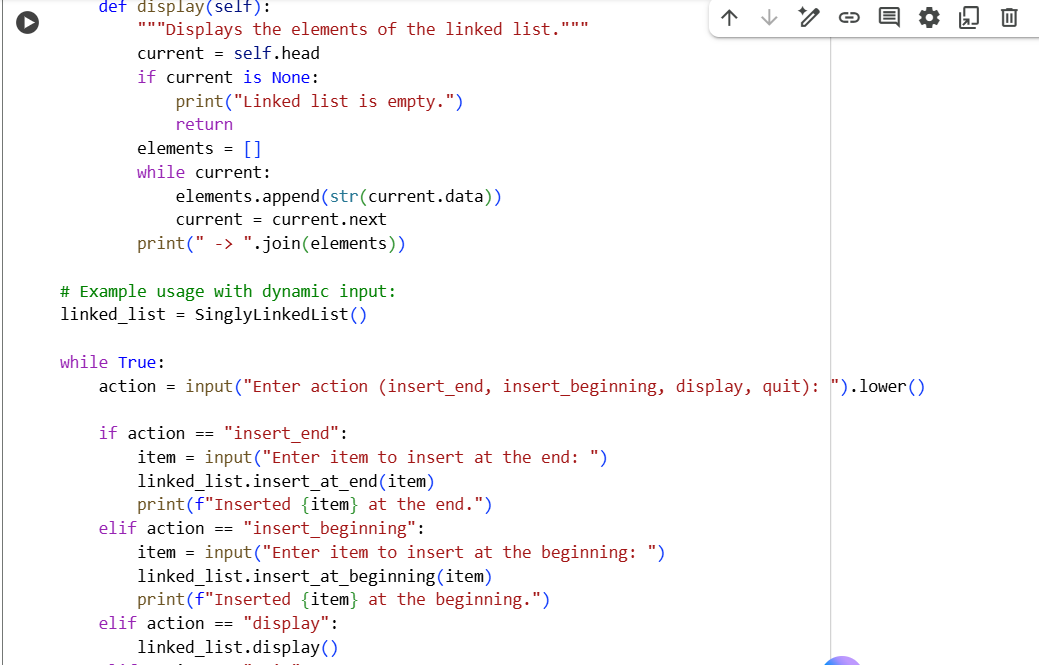
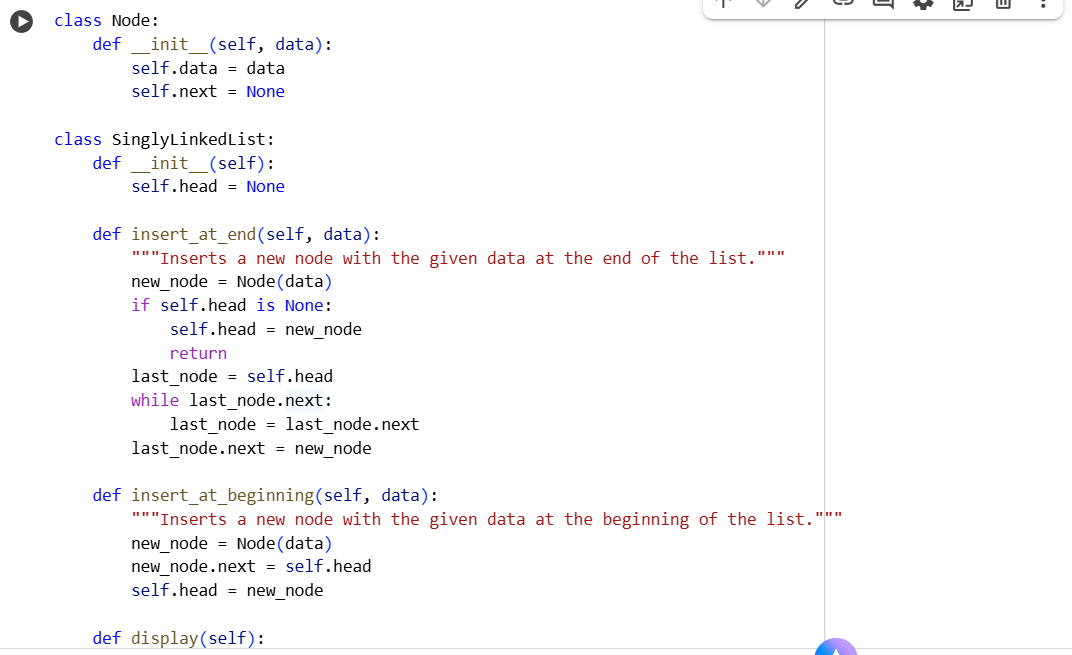


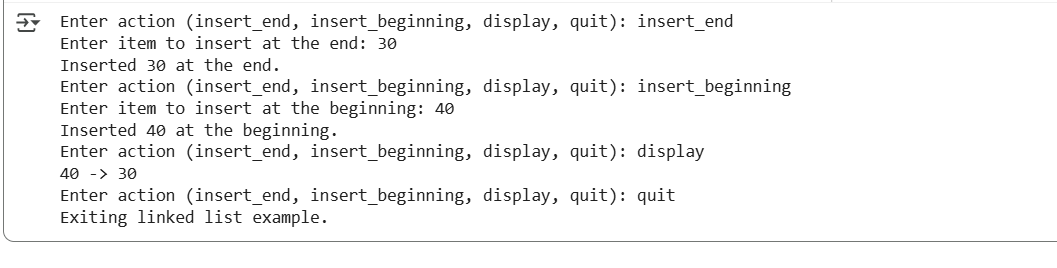
* TASK DESCRIPTION-3:

Linked List Implementation  
Task: Ask AI to create a singly linked list with insert\_at\_end(),  
insert\_at\_beginning(), and display().

* PROMPT:

Create a Python class for a singly linked list with methods insert\_at\_end(), insert\_at\_beginning(), and display() with dynamic input.

* CODE: 
* OUTPUT:



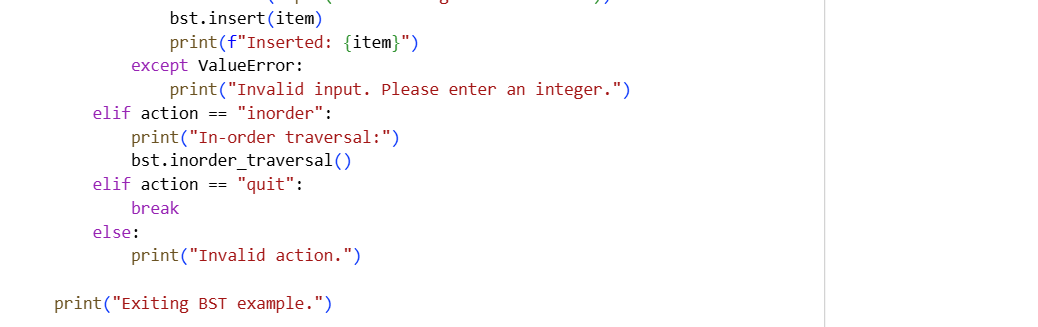
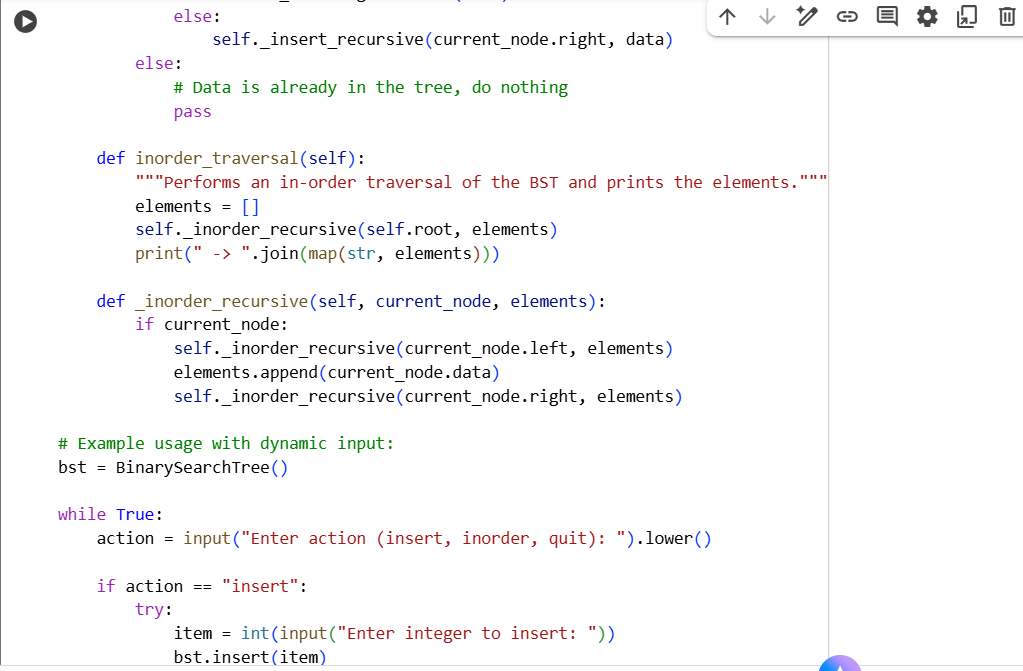
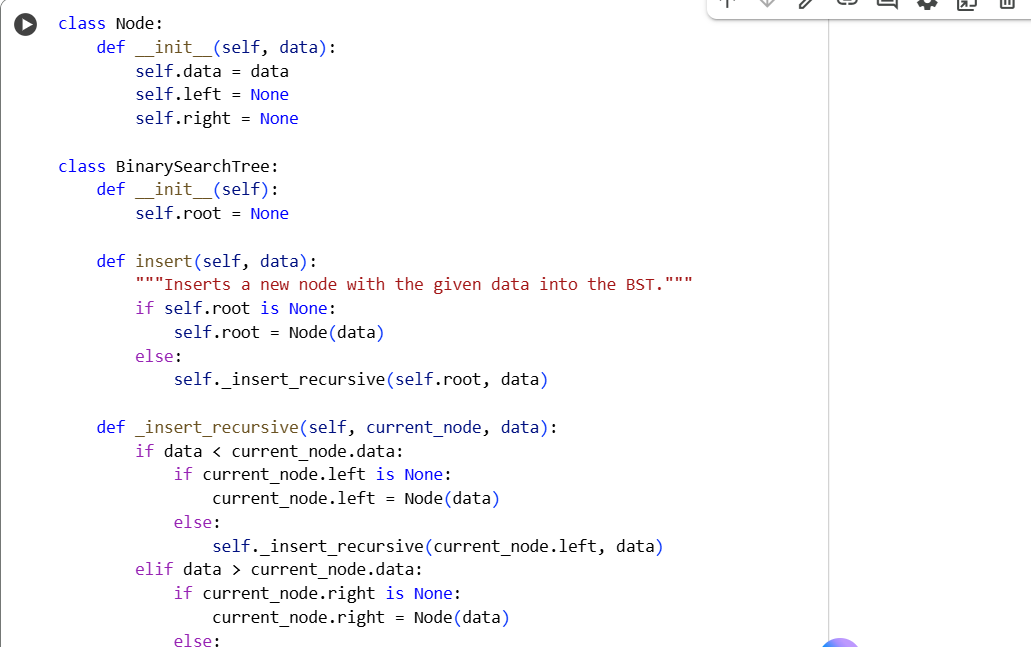
* TASK DISCRIPTION-4:

Binary Search Tree (BST)  
Task: Ask AI to generate a simple BST with insert() and  
inorder\_traversal().

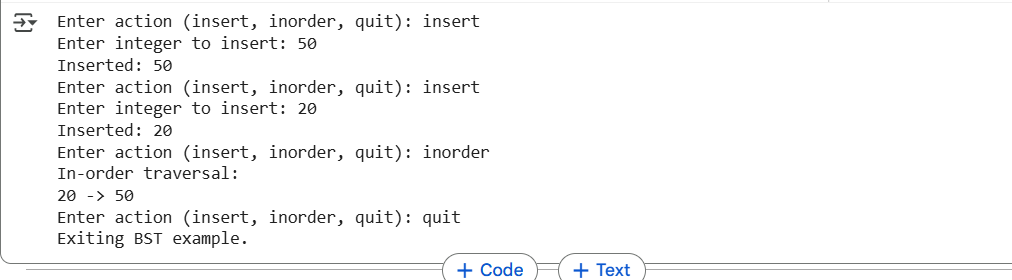
* PROMPT:

Create a Python class for a Binary Search Tree (BST) with methods insert(data) and inorder\_traversal() with dynamic input.

* CODE:



* OUTPUT:



---------------Thank You------------