Project: User Activity Tracker using deque
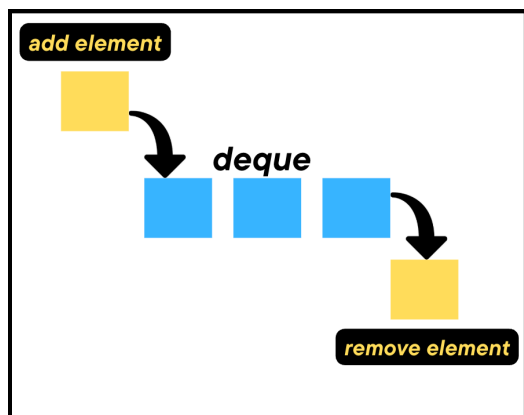
Project Overview:

This project demonstrates how to track the most recent user actions using Python's ==deque== from the ==collections== module.

```
In [ ]:  from collections import deque
```

==Deque (double-ended queue)== allows us to add or remove items from both ends efficiently and helps in maintaining a limited history of actions.

Application:

Applications of User Activity Tracker:

- **Track recent user actions in software and websites.**
- **Maintain limited chat history or activity logs.**
- **Monitor recent file edits in a system.**
- **Track recent searches in apps.**

Useful in real-time systems where only the latest actions are important.

Python Code:

from collections import deque

 # Create a deque with fixed size (maxlen)

```
recent_actions = deque(maxlen=5)

# Simulate user actions
def action_logs(action):
    recent_actions.append(action)
    print(f"Action Logged: {action}")
    print("Recent Actions:", list(recent_actions))
    print("-" * 40)

# Example usage
action_logs("User has logged in")
action_logs("User has searched Python Deque")
action_logs("User viewed Channel : LearnWithBhawana")
action_logs("User ran Python script")
action_logs("User opened Report.PDF")
action_logs("User locked Report.PDF")
action_logs("User logged out.")
```

```
In [17]: from collections import deque

         # Create a deque with fixed size (maxlen)
         recent_actions = deque(maxlen=5)

         # Simulate user actions
         def action_logs(action):
             recent_actions.append(action)
             print(f"Action Logged: {action}")
             print("Recent Actions:", list(recent_actions))
             print("-" * 40)

         # Example usage
         action_logs("User has logged in")
         action_logs("User has searched Python Deque")
         action_logs("User viewed Channel : LearnWithBhawana")
         action_logs("User ran Python script")
         action_logs("User opened Report.PDF")
         action_logs("User locked Report.PDF")
         action_logs("User logged out.")

         Action Logged: User has logged in
         Recent Actions: ['User has logged in']
         ----------------------------------------
         Action Logged: User has searched Python Deque
         Recent Actions: ['User has logged in', 'User has searched Python Deque']
         ----------------------------------------
         Action Logged: User viewed Channel : LearnWithBhawana
         Recent Actions: ['User has logged in', 'User has searched Python Deque', 'User viewed Channel : LearnWithBhawana']
         ----------------------------------------
         Action Logged: User ran Python script
         Recent Actions: ['User has logged in', 'User has searched Python Deque', 'User viewed Channel : LearnWithBhawana',
         'User ran Python script']
         ----------------------------------------
         Action Logged: User opened Report.PDF
         Recent Actions: ['User has logged in', 'User has searched Python Deque', 'User viewed Channel : LearnWithBhawana',
         'User ran Python script', 'User opened Report.PDF']
         ----------------------------------------
         Action Logged: User locked Report.PDF
         Recent Actions: ['User has searched Python Deque', 'User viewed Channel : LearnWithBhawana', 'User ran Python scrip
         t', 'User opened Report.PDF', 'User locked Report.PDF']
         ----------------------------------------
         Action Logged: User logged out.
         Recent Actions: ['User viewed Channel : LearnWithBhawana', 'User ran Python script', 'User opened Report.PDF', 'Use
         r locked Report.PDF', 'User logged out.']
         ----------------------------------------
```

Step-by-Step Explanation:

1. Import deque from the collections module.

2. Create a deque with maxlen=5 to store only the 5 most recent actions.

3. Define a function, action_logs( ) that logs each action to the deque and prints the current action list.

4. Call action_logs( ) multiple times to simulate user activity.

Only the most recent 5 actions are retained. Older actions are automatically removed.