Question 3: Adaptive RL Agent for Atari Breakout with Dynamic Difficulty

When I was given this question on implementing an adaptive Deep Q-Network (DQN) for a dynamically changing version of Atari Breakout, I realized I was entering a territory I had not explored before. While I have worked on machine learning models and image processing tasks, reinforcement learning (RL) was a domain I had only heard of but never implemented.

I studied how RL works, what a DQN is, how experience replay functions, and how curriculum learning can be used to gradually expose agents to increasing complexity.

I read multiple papers, watched tutorials, and tried to sketch out the components of the DQN pipeline, including the neural network structure, replay buffer, epsilon-greedy strategy, and target networks. I now have a clearer conceptual understanding of how an agent learns by interacting with an environment, especially when the environment itself is changing in unpredictable ways.

Due to time constraints and being new to RL implementation, I was not able to complete the code fully.

This experience has motivated me to explore RL further, and I intend to follow through by implementing this project independently—even if not under the deadline.