

# Aproximación de la Ecuación de Movimiento de un Péndulo Simple utilizando Redes Neuronales

Juan David Osorio

Mayo 2024

## Abstract

Este informe presenta el proceso y los resultados del Proyecto II del curso de Sistemas Inteligentes II, cuyo objetivo fue entrenar y comparar diferentes modelos DQN (Deep Q-Network) para resolver los problemas de control clásicos en el entorno OpenAI Gym CartPole y MountainCar. Se experimentó con diversas configuraciones de hiperparámetros para evaluar su impacto en el rendimiento del modelo.

## 1 Introducción

El control de sistemas dinámicos es un área crucial en la inteligencia artificial y la robótica. Este proyecto se centra en el uso del algoritmo DQN para resolver problemas de control en entornos simulados proporcionados por OpenAI Gym. El objetivo es entrenar y comparar diferentes configuraciones de modelos DQN para evaluar su eficacia en los entornos CartPole y MountainCar.

## 2 Metodología

### 2.1 Configuración del entorno OpenAI Gym

Se utilizaron los entornos CartPole y MountainCar de OpenAI Gym para entrenar los modelos DQN. Estos entornos proporcionan desafíos clásicos de control que son ampliamente utilizados para evaluar algoritmos de aprendizaje por refuerzo.

## **2.2 Arquitectura del modelo DQN**

El mejor modelo DQN para MountainCar con 2 capas con 64 neuronas y una función de activación ReLU. La capa de salida es una función de activación lineal.

## **2.3 Hiperparámetros utilizados**

Los hiperparámetros clave que se experimentaron incluyen la tasa de aprendizaje, el tamaño del batch, el factor de descuento (gamma), la tasa de exploración-explotación (epsilon), el número de épocas, el número de capas y el número de neuronas en cada capa. En el caso específico de MountainCar se añadió un hiperparámetro para controlar el peso que tiene la velocidad en la recompensa.

## **2.4 Estrategia de entrenamiento**

Los modelos se entrenaron utilizando la técnica de aprendizaje por refuerzo con experiencia de repetición, donde se almacenan transiciones en un buffer de experiencia y se realizan actualizaciones periódicas de la red objetivo. Además se utilizó la política de epsilon-greedy para controlar la exploración del modelo.

## **3 Resultados**

Los modelos finales muestran una buena capacidad de acción y precisión en los entornos de control de OpenAI Gym. Se evaluaron las recompensas promedio y la tasa de éxito en cada entorno. Los resultados se presentan en forma de gráficos y tablas comparativas.

**Mountain**

**CartPole**

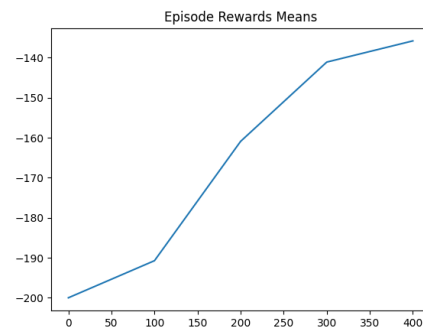
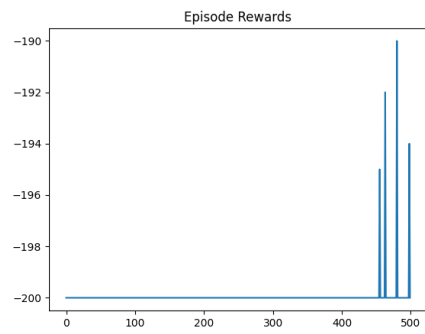
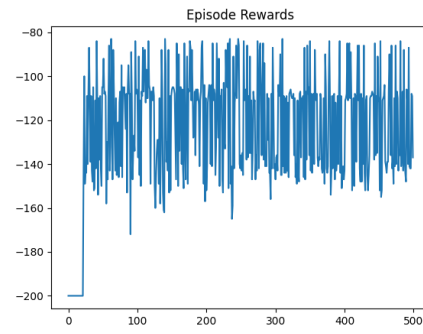
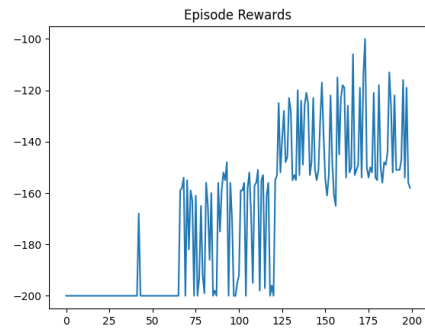
## **4 Discusión**

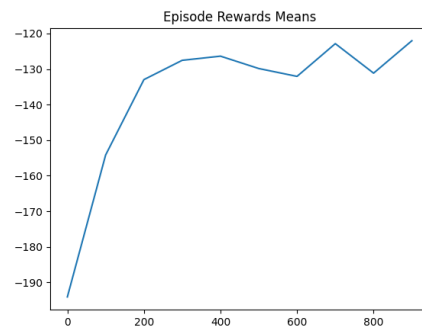
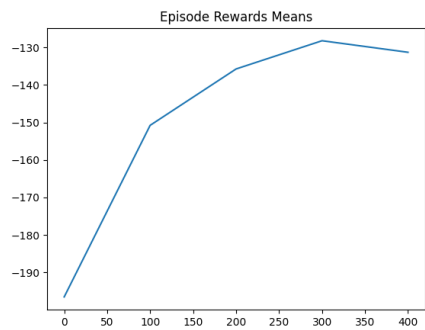
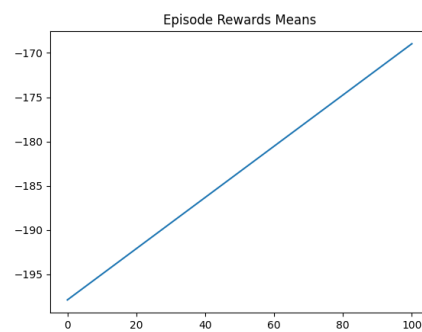
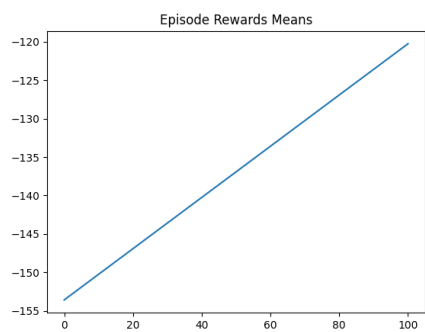
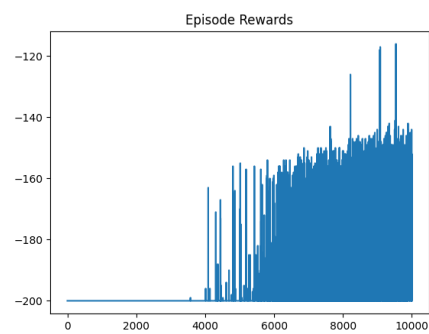
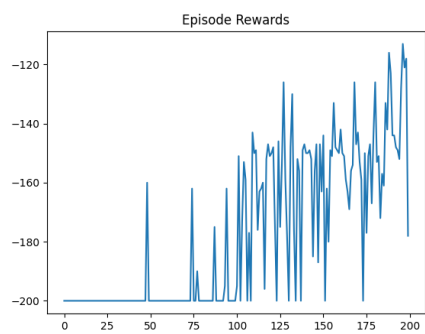
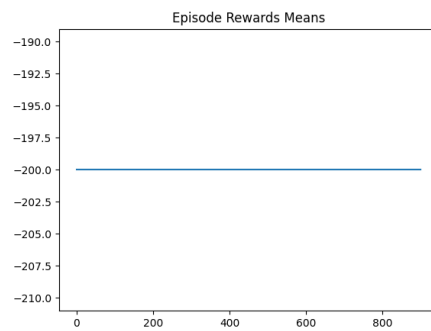
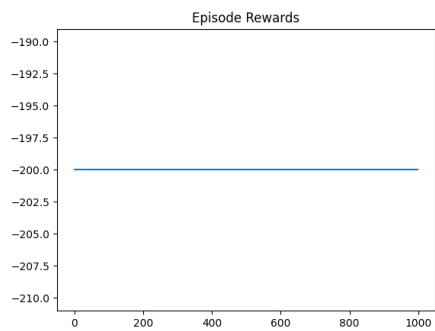
Se entrenaron múltiples modelos variando los hiperparámetros como la cantidad de epochs, la cantidad de capas y la cantidad de neuronas en cada capa. La decisión del mejor modelo se tomó a partir del valor mas bajo de error en validación.

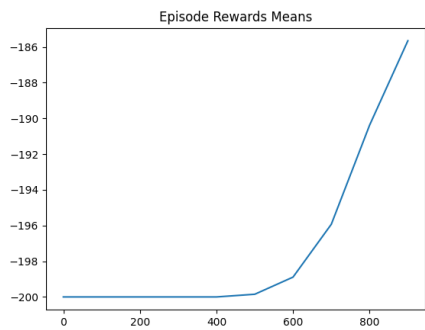
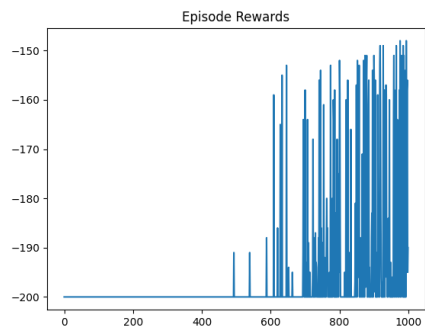
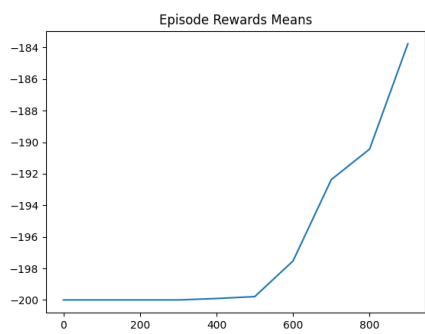
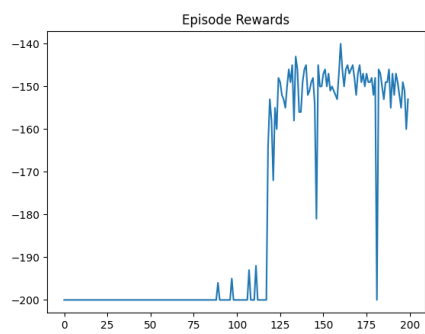
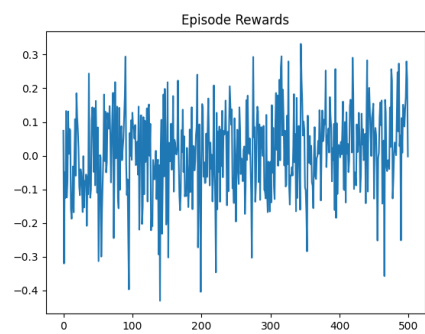
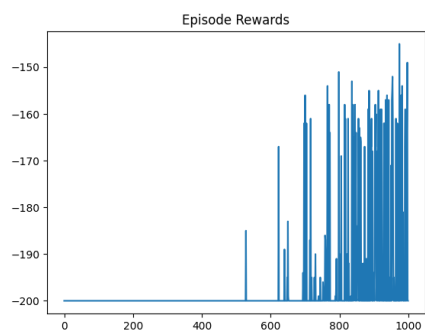
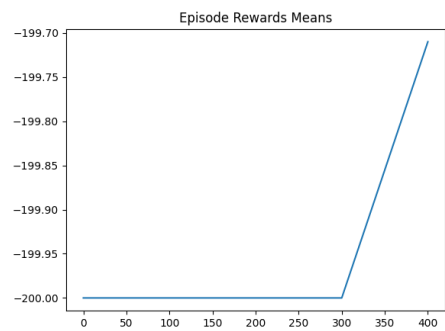
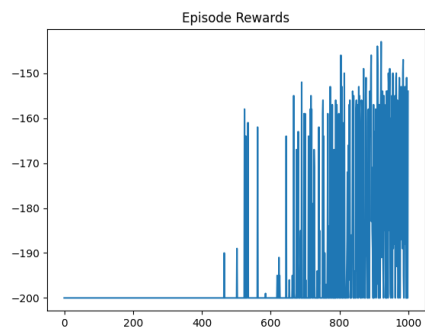
model	evaluate_max_reward	evaluate_rewards_mean	evaluate_rewards_std
2024-06-15_12-31-16_dqn_mountaincar	-87.0	-109.1	20.344
2024-06-15_13-02-52_dqn_mountaincar	-86.0	-112.9	22.106
2024-06-15_13-39-22_dqn_mountaincar	-146.0	-150.3	2.9681
2024-06-15_13-45-28_dqn_mountaincar	-116.0	-183.5	33.006
2024-06-15_12-43-21_dqn_mountaincar	-84.0	-121.7	23.656
2024-06-15_13-00-05_dqn_mountaincar	-85.0	-115.1	26.200
2024-06-15_08-13-32_dqn_mountaincar	-200.0	-200.0	0.0
2024-06-15_13-20-27_dqn_mountaincar	-140.0	-147.8	8.6810
2024-06-15_10-54-10_dqn_mountaincar	-108.0	-141.2	11.745
2024-06-15_13-41-53_dqn_mountaincar	-92.0	-130.9	23.394
2024-06-15_12-23-27_dqn_mountaincar	-87.0	-126.7	23.056
2024-06-15_11-21-25_dqn_mountaincar	-110.0	-137.7	19.365
2024-06-15_13-45-47_dqn_mountaincar	-96.0	-134.2	24.632
2024-06-15_11-01-53_dqn_mountaincar	-85.0	-109.1	20.539
2024-06-15_12-35-51_dqn_mountaincar	-89.0	-127.1	22.232
2024-06-15_16-46-16_dqn_mountaincar	-111.0	-128.5	14.235
2024-06-15_12-16-56_dqn_mountaincar	-108.0	-126.6	17.482
2024-06-15_13-48-51_dqn_mountaincar	-93.0	-113.8	14.218
2024-06-15_12-43-26_dqn_mountaincar	-144.0	-154.9	17.397
2024-06-15_13-36-42_dqn_mountaincar	-84.0	-104.4	18.456
2024-06-15_11-57-24_dqn_mountaincar	-118.0	-136.7	16.186
2024-06-15_13-56-21_dqn_mountaincar	-85.0	-114.8	26.324

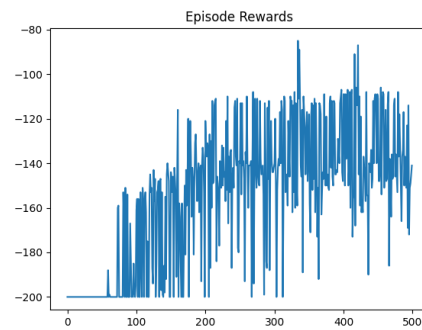
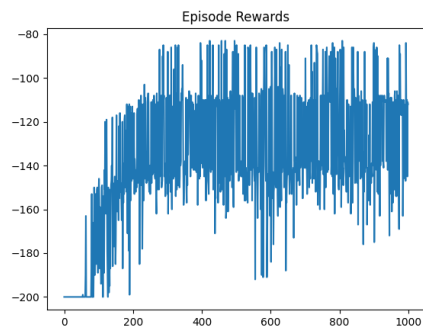
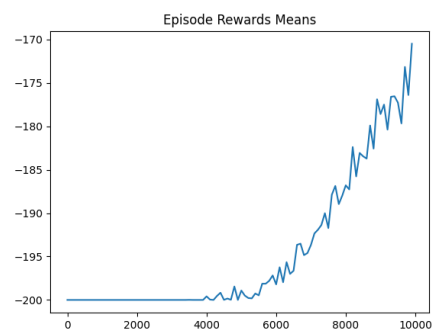
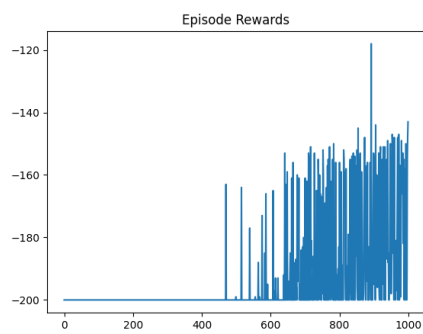
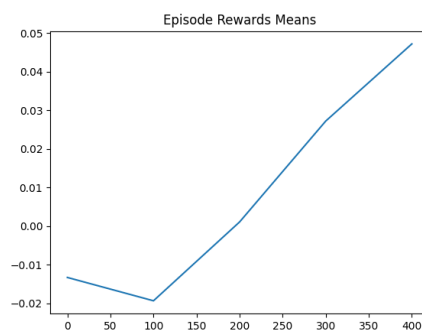
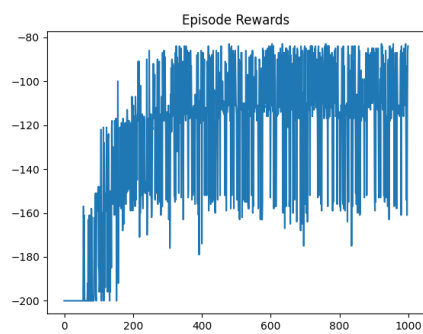
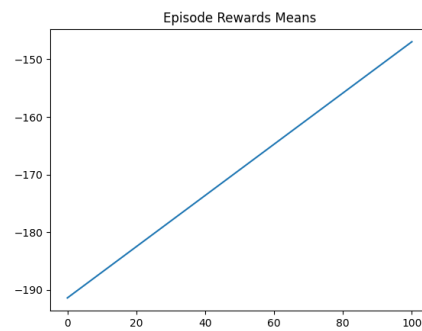
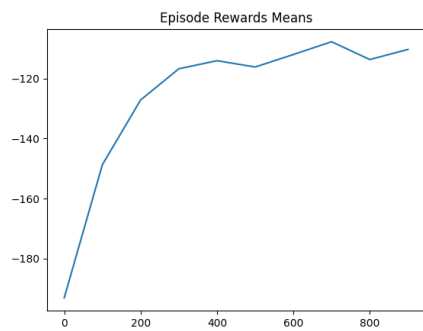
model	evaluate_max_reward	evaluate_rewards_mean	evaluate_re
2024-06-13_17-34-57_dqn_cartpole	10.0	10.0	0.0
2024-06-14_00-20-45_dqn_cartpole	86.0	86.0	0.0
2024-06-13_21-08-24_dqn_cartpole	92.0	92.0	0.0
2024-06-14_09-34-50_dqn_cartpole	201.0	201.0	0.0
2024-06-13_18-59-17_dqn_cartpole	86.0	86.0	0.0
2024-06-14_17-31-42_dqn_cartpole	214.0	214.0	0.0
2024-06-14_18-54-27_dqn_cartpole	9.0	9.0	0.0
2024-06-14_18-17-10_dqn_cartpole	95.0	95.0	0.0
2024-06-13_18-42-33_dqn_cartpole	9.0	9.0	0.0
2024-06-13_23-08-40_dqn_cartpole	89.0	89.0	0.0
2024-06-14_16-42-26_dqn_cartpole	119.0	119.0	0.0
2024-06-15_00-29-02_dqn_cartpole	10.0	10.0	0.0
2024-06-13_18-50-05_dqn_cartpole	336.0	336.0	0.0
2024-06-14_18-50-42_dqn_cartpole	500.0	500.0	0.0
2024-06-13_19-11-27_dqn_cartpole	88.0	88.0	0.0
2024-06-14_13-15-56_dqn_cartpole	9.0	9.0	0.0
2024-06-14_15-38-10_dqn_cartpole	214.0	214.0	0.0
2024-06-13_18-56-41_dqn_cartpole	10.0	10.0	0.0
2024-06-14_11-09-14_dqn_cartpole	255.0	255.0	0.0
2024-06-13_19-37-45_dqn_cartpole	9.0	9.0	0.0
2024-06-13_21-45-52_dqn_cartpole	230.0	230.0	0.0
2024-06-13_22-34-12_dqn_cartpole	96.0	96.0	0.0
2024-06-13_20-39-12_dqn_cartpole	14.0	14.0	0.0
2024-06-14_19-20-40_dqn_cartpole	500.0	500.0	0.0
2024-06-14_18-05-51_dqn_cartpole	97.0	97.0	0.0
2024-06-14_13-24-09_dqn_cartpole	44.0	44.0	0.0
2024-06-13_23-49-30_dqn_cartpole	92.0	92.0	0.0
2024-06-14_17-21-25_dqn_cartpole	86.0	86.0	0.0
2024-06-13_22-09-48_dqn_cartpole	90.0	90.0	0.0
2024-06-13_19-16-11_dqn_cartpole	49.0	49.0	0.0

# Mountain

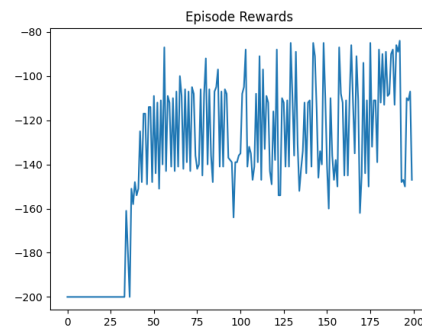
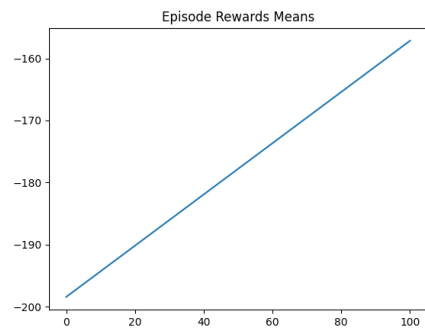
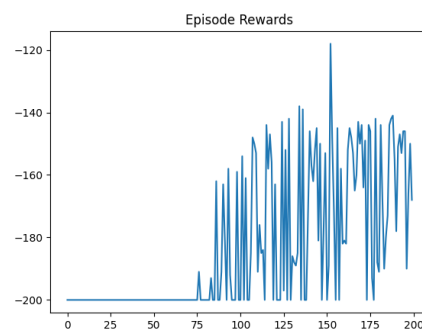
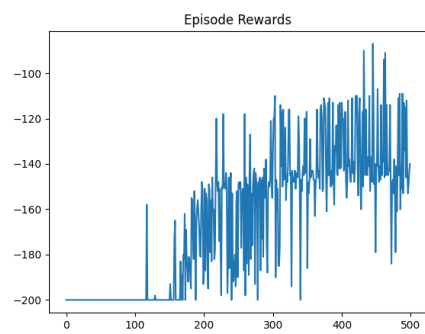
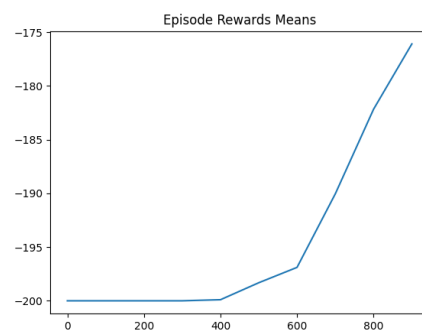
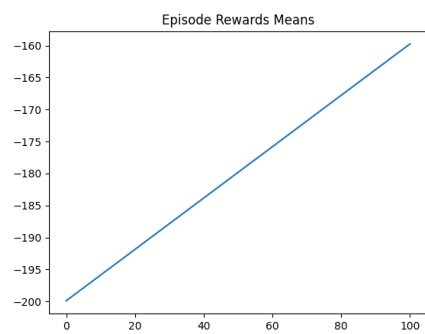
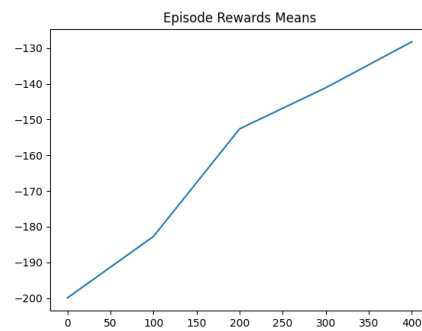
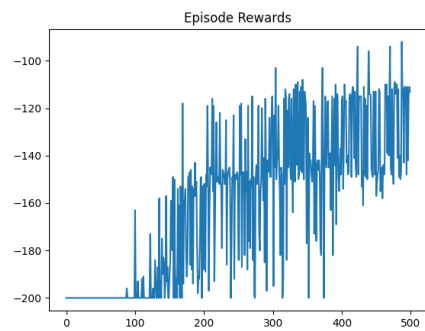


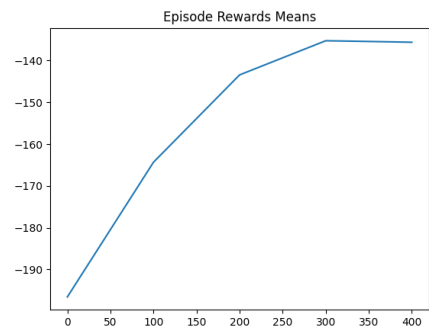
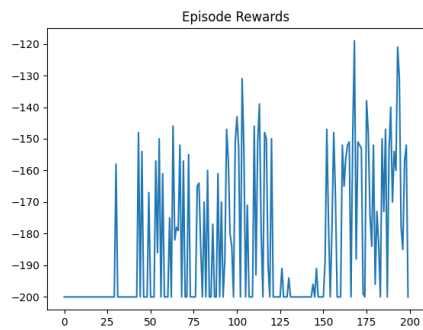
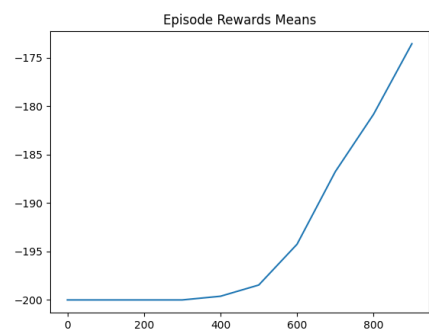
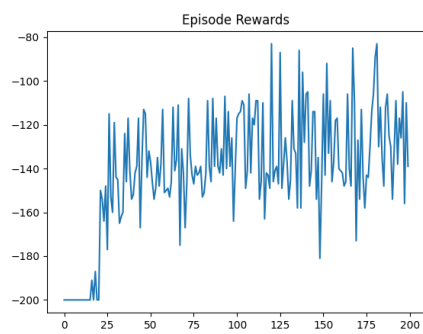
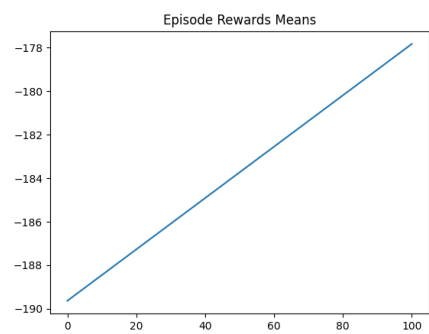
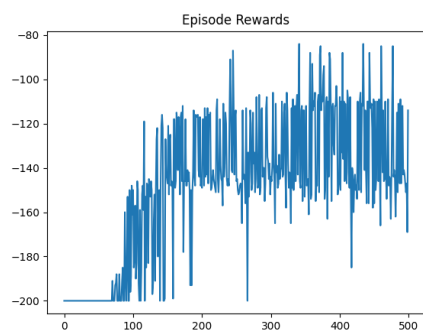
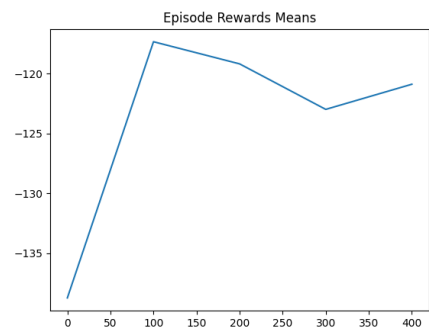
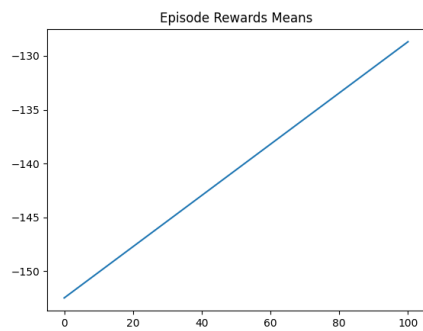




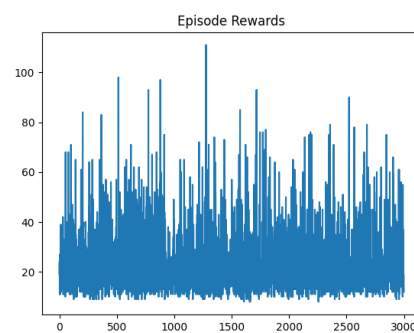
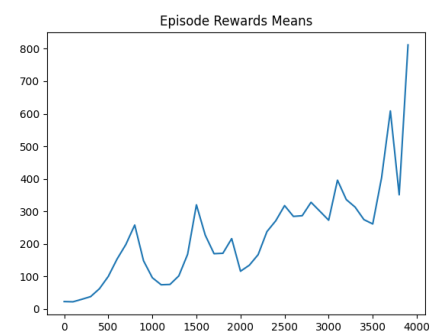
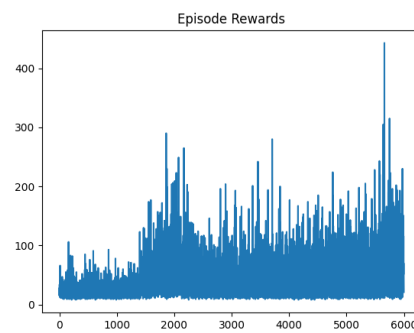
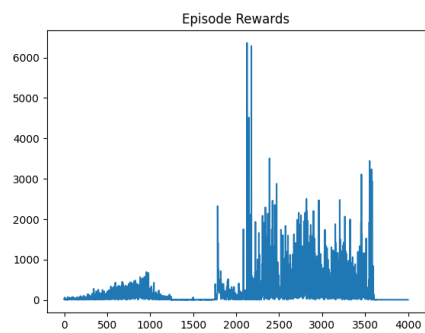


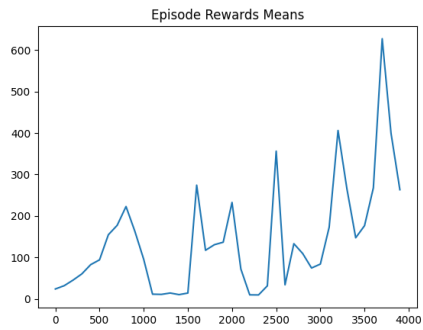
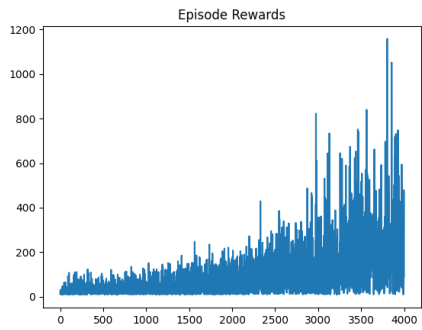
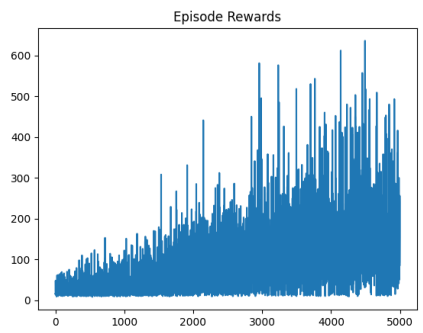
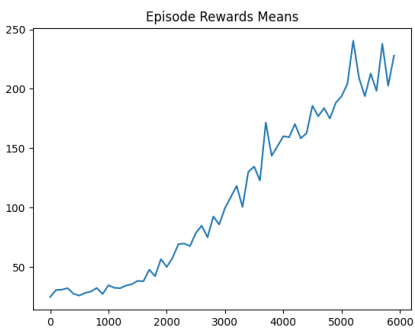
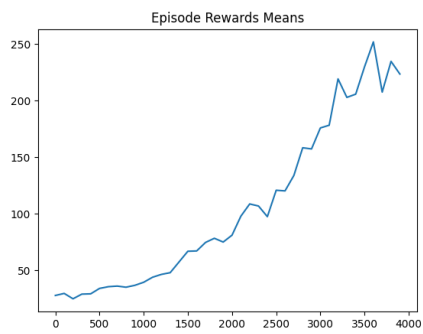
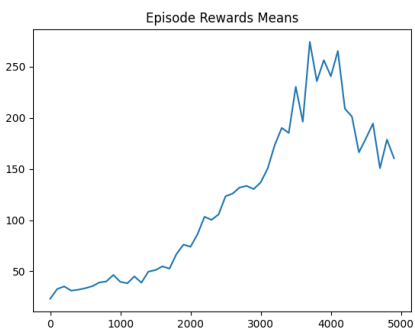
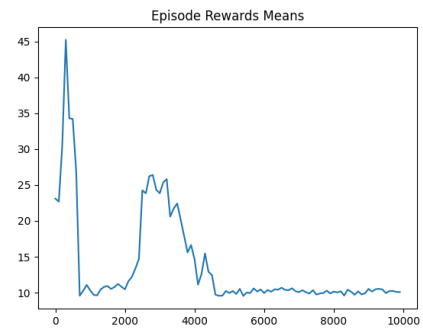
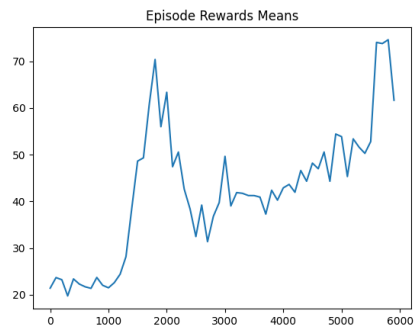


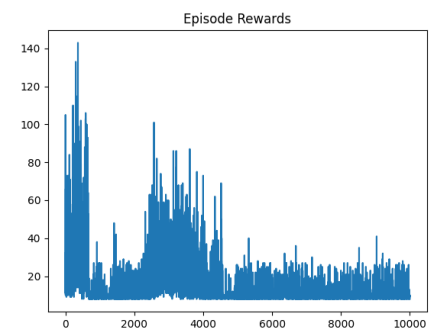
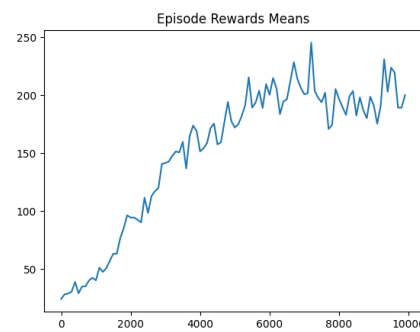
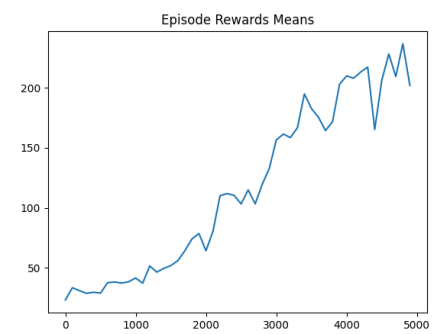
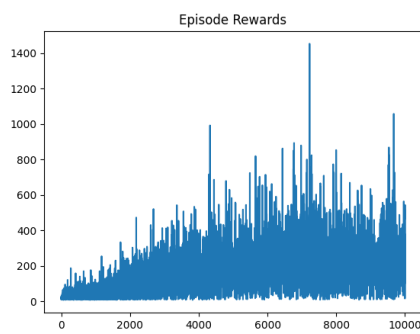
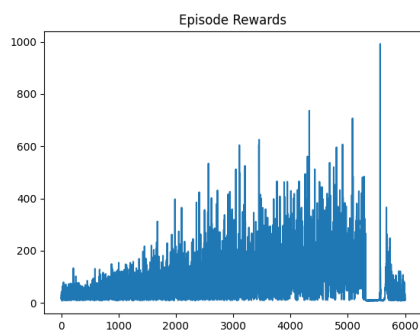
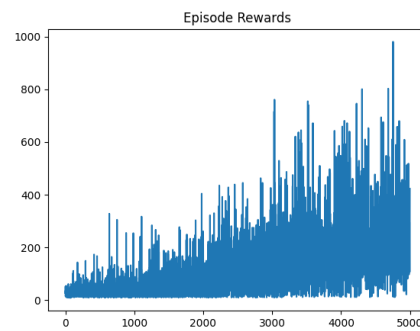
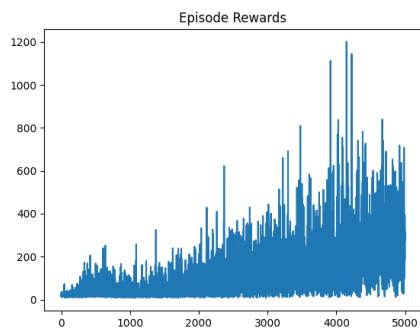


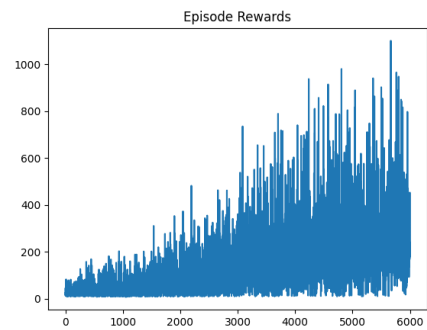
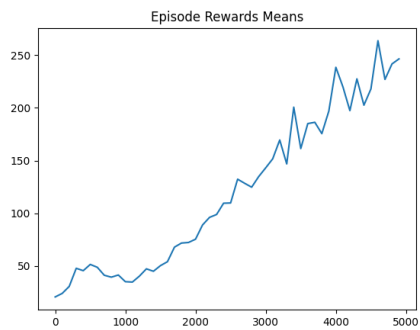
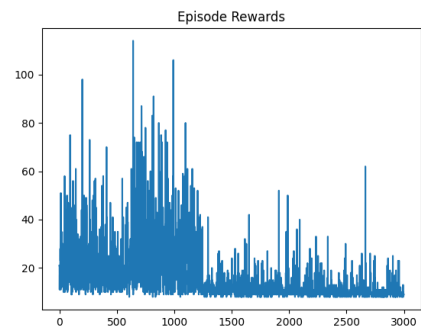
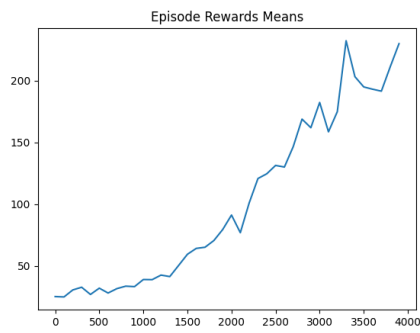
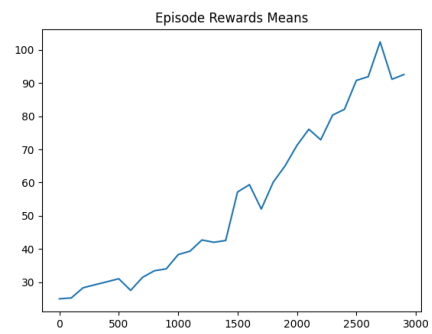
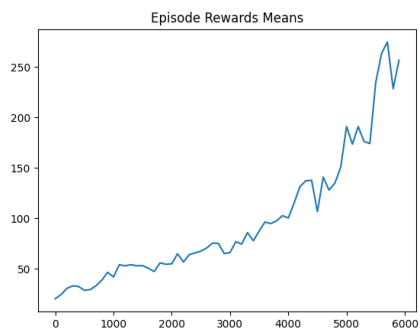
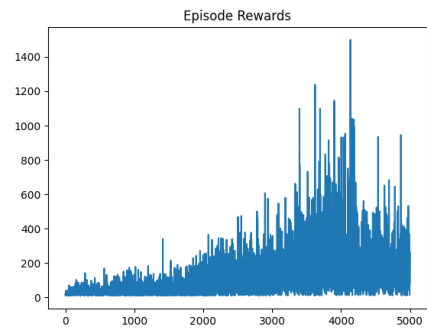


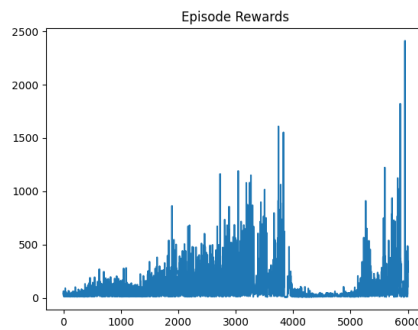
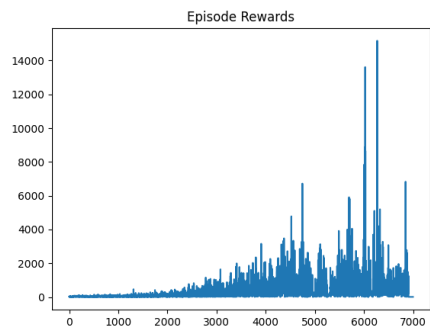
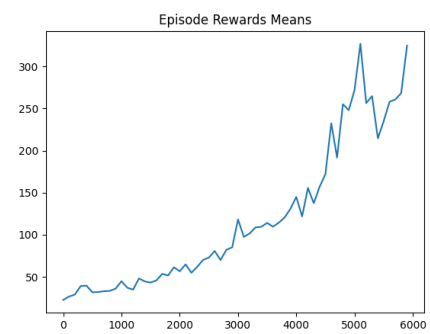
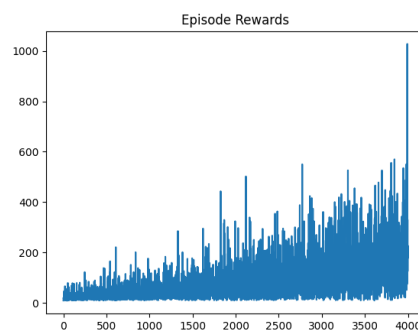
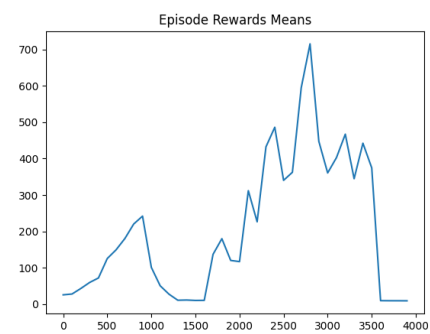
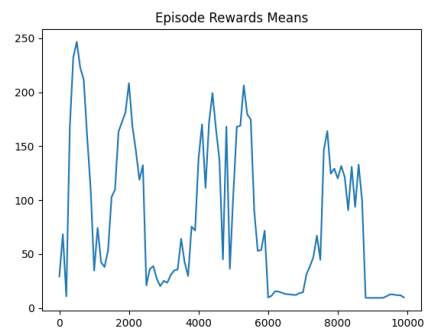
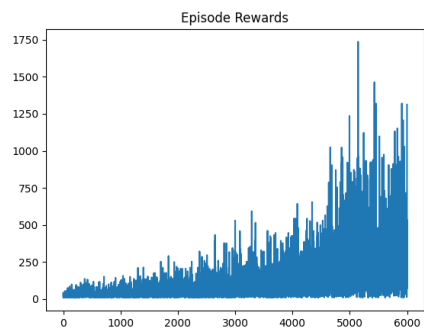
## CartPole

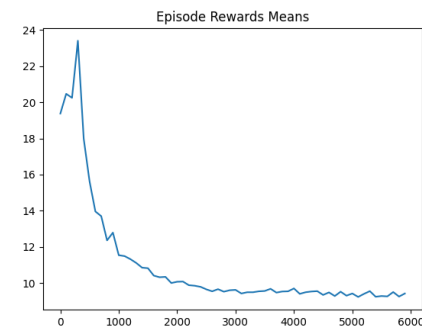
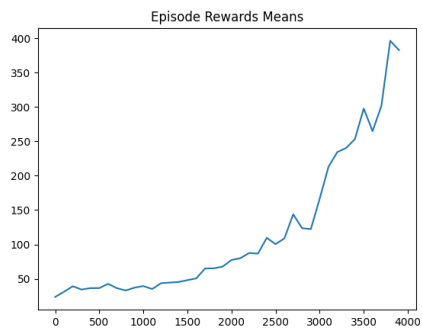
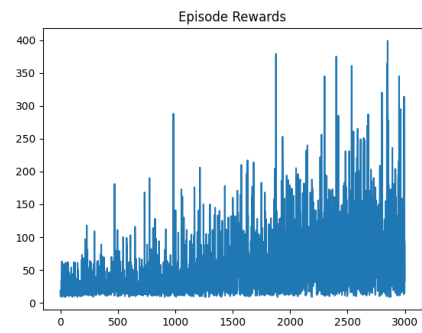
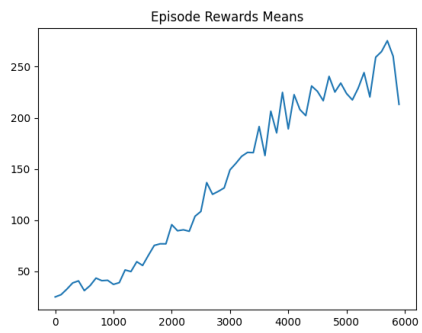
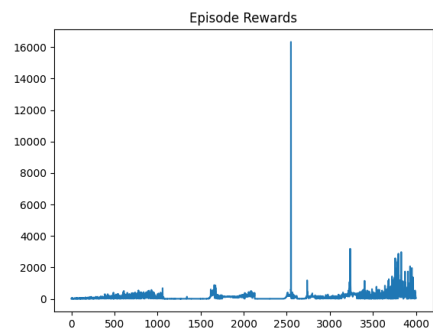
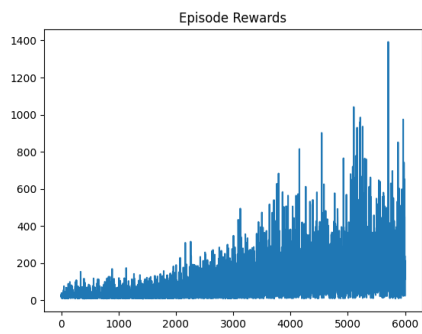
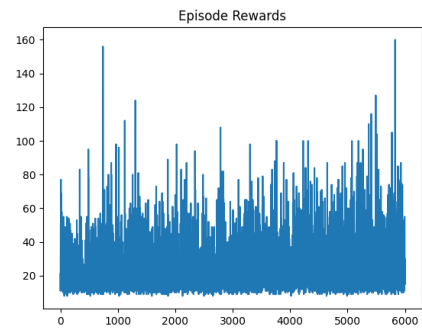
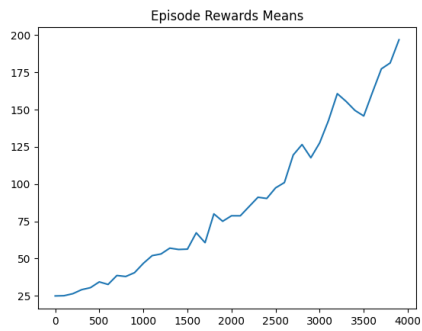




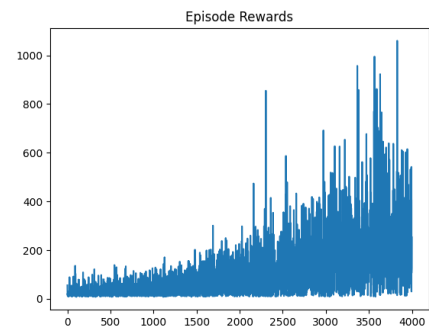
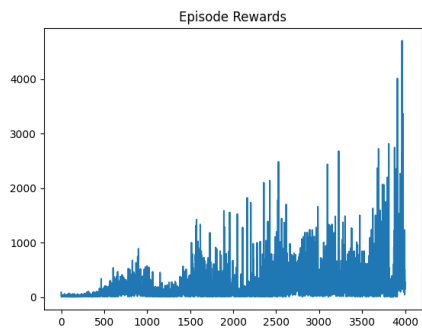
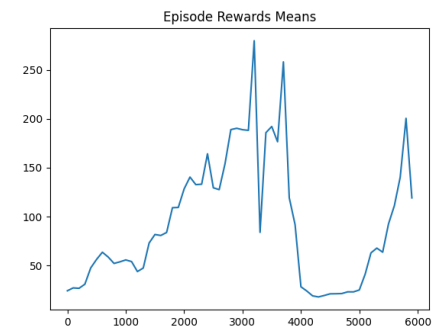
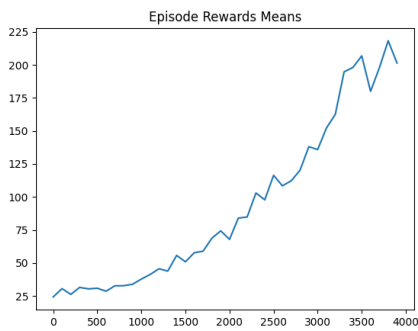
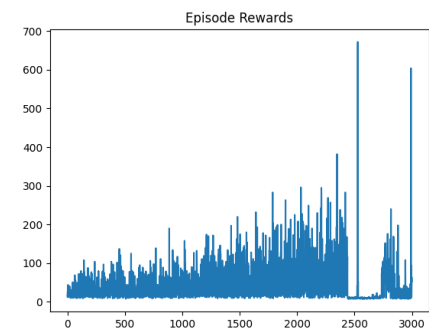
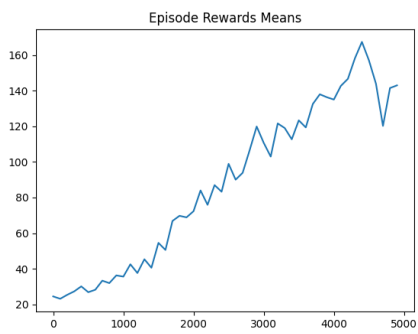
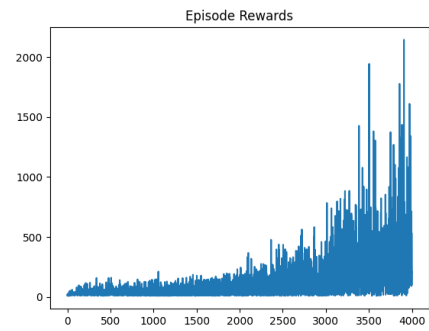


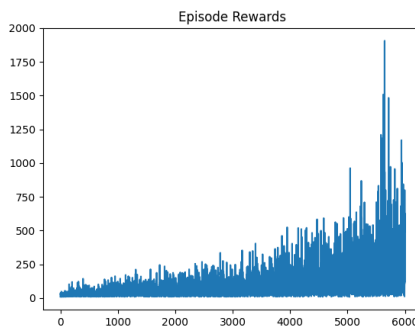
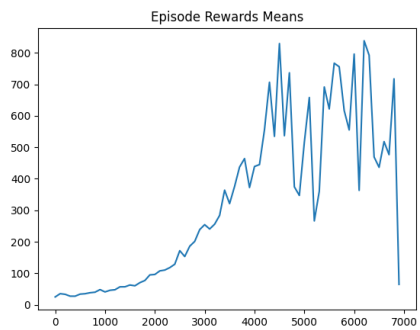
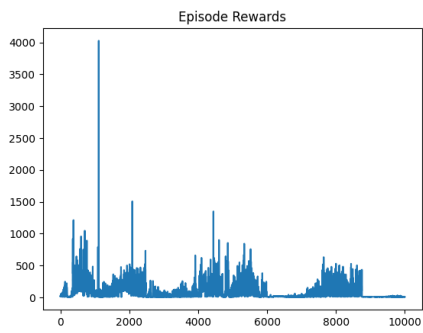
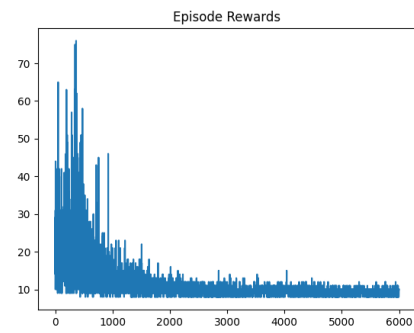
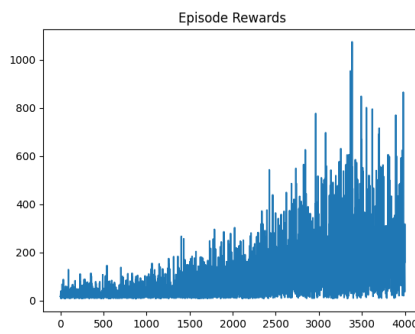
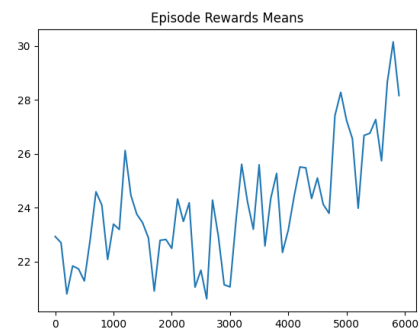
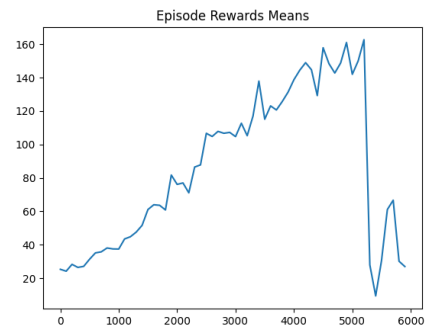
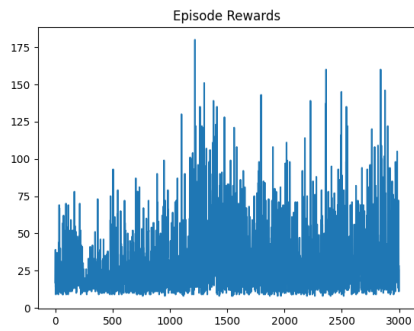












## 5 Conclusión

Concluimos que las redes neuronales pueden ser una herramienta eficaz para aproximar la solución de la ecuación de movimiento de un péndulo simple. Sin embargo, es importante considerar la complejidad del modelo y la calidad de los datos de entrada para obtener resultados precisos.