First tutorial for designing the game: JavaScript snake game tutorial

- Did not find the tutorial well explained
- At the end of the tutorial, the end game condition does not work
- I'll have to find another tutorial to hopefully troubleshoot my issues
- I used a classic Nokia snake game design and colour for my inspo
    - Tool I used to grab the colour codes: Pix Spy
- I like the 800x800 grid visually, but having such a large grid will mean much longer games and therefore much more time spent training. I will reduce the game board by half
- I have a "score" variable but I will have to find a way to keep track of the final score outside of the script...
- I really want to model my food after the classic design but not sure how hard that'll be and whether it'll be worth the effort, but we'll see
- After a nice long and frustrating conversation with my trusty pal ChatGPT, I found my game end solution. I needed another function to reset the game
- Added an alert before restarting the game so it doesn't run endlessly, got the idea from this tutorial:
    - Create a snake game using HTML, CSS and JavaScript
- Instead of trying to save the score with the game script, I'll see about saving it with the AI script
    - Within the JS, I could send the data to MySQL db or potentially a csv file which would obv be the better option probably
    - But the benefit of only keeping track of the score via the AI is that runs that I do manually won't be tracked and that is definitely preferable
- Another semi-frustrating conversation with my pal ChatGPT got me the food design I was looking for, yay!
- I believe that is a wrap on designing the game... Onto next steps
- Did an intro course on OpenCV to get an idea of how it works and some of the library functionalities
- After finishing the course I'm still not sure how to get the game state into my bot so I'll have to do some more research
- Resources I am exploring to figure out my next step:
    - r/computervision Real-time Image Processing on computer screen data
    - General Video Game AI: Learning from Screen Capture
        - Deep Q-Network - receives screen information and produces actions as it's output
            - Consists of 2 parts; a deep CNN and Q-learning
            - CNN responsible for getting key features and determining best action
            - Q-Learning responsible for evaluating state-actions of each frame
        - Section on Q-Learning was very enlightening:

- Q-learning is an off-policy temporal difference learning algorithm in Reinforcement Learning [19] (RL), which is a Machine Learning paradigm that aims to find the best policy to react in the problem it is solving. This is done by maximising reward signals given from the environment for each available action in a given situation. Q-learning is a model-free RL technique that allows online learning, and it updates the current state values according to maximum return values of all states after providing available actions. A main challenge in Reinforcement Learning is to balance exploitation and exploration. Exploitation chooses the best action found so far, whilst exploration selects another alternative option to improve the current policy. A simple solution for this is to apply an ▯-greedy policy, which selects an action at random with probability ▯. In our implementation, ▯ was initialised at 1 and decreased by 0.1 in every time step until it stabilizes at 0.1.
- r/learnmachinelearning AI play T rex game based on screenshot, using reinforcement learning.
  - Q: How do u get game data like the score? A: use selenium chrome webdrive to grab the JavaScript element. As for which element to grab, I refer to other people's code
  - Resources OP provided:
    - An introduction to Deep Q-Learning: let's play Doom
    - The other 2 were dead… The post was 4 years old after all
      - Actually one was a github page that has been taken down but the repo still exists
        - TF-Rex
- Screen Capturing and Streaming With Python
  - Except it was of course members only… This definitely would've helped
  - Found out it was originally posted on his personal blog
    - Screen Capturing and Streaming With Python
- I can use Pillow to grab pictures of the screen I just need to figure out how to open and feed the game to it
- Okaaayyyyyyy based on some chats with chatGPT and what I've seen here + others over the course of my research
  - Webdriver Screenshot in Python]
  - How to open an HTML file in the browser from Python?
  - Python PIL | Image.crop() method
- I decided to go with a Selenium screenshot + Pillow crop and save method
- Since Selenium is literally built for browser stuff, it is the most "obvious" choice to open / interact with my browser-driven game
- Pillow is a simple way to crop the images before saving them
  - I was playing around with the idea of making the images grayscale to "simplify" them for the model but not sure it's necessary since they're already only 2 colours. I found the

necessary code here

    - Using PIL to turn a RGB image into a pure black and white image

- Now I just need to set it up so that it continually takes images while the browser is open and stack the images for consumption by the model
- Putting it in a try/while is allowing it to loop now the hard(er) part..... I've got pics of the game... now what???
- Also, the alert is causing the game to close but Selenium has some functions for dealing with alerts so it shouldn't be a problem, I can probably get the agent script to use the Selenium fxn to continue
  - Google Search
- Alright, I've got the screenshots going, images are deleted once the game is closed, the game over alert closes the game and restarts it, keyboard interrupt also closes the game
- So so far everything is working as expected so I think I'm ready to move on to the really tough part... the model and also how tf I'm going to handle the images that are being taken...
- Also need to consider deleting images after they're used so I don't just have an infinitely growing images folder
- Looking at this FreeCodeCamp tutorial and the respective GitHub:
  - An introduction to Deep Q-Learning: let's play Doom
  - Deep Q Learning/Doom/Deep Q learning with Doom.ipynb
- It's a different game and there's a lot of differences between the projects but it seems to be doing the image-taking/feeding that I'm looking to do
- Works episodically, each episode is a game, you receive a reward after each action and at the end of the game you get the total reward
- Images are preprocessed to reduce the size and complexity
- Frames are stacked to give the model a sense of motion, they do 4 and that seems to be the most common # of frames
- Frames are stacked in a deque so that when a new one is added, the oldest one is pushed out
  - This may be an opportunity for me to delete the one pushed out of the deque
- Actually, it looks like in their implementation, the images are never saved and are just passed to the stack function
- I suppose I'll modify my code to do this as well and implement their stacking function
- I found this Pytorch tutorial for DQN: Reinforcement Learning (DQN) Tutorial
- I'm trying to adapt it for my needs.... Not as easy as it seems
- Currently I'm just running it and dealing with the errors as they come up
- My biggest problem though is that the rewards function is not defined as it was built into the game? Same with the FCC one too
- Looks like I've made my life a bit more difficult for myself by not doing a game through a library like pygame where I would've had access to all of the game states very easily
- I ended up adding a score and new_score variable to check for when the score increases
- This is how I implemented the reward function

- Turns out the tutorial I was using was doing a 1D input and of course the one I'm doing is 3D(?)
- So I needed to change the SQN class from using linear functions to conv2d first and then linear
- I also had to permute the inputs because they were [1, 45, 45, 4] and the function was expecting [1, 4, 45, 45]
- Once those issues were straightened out I had a new problem, the alert was throwing exceptions that weren't being caught by the try/except block
- I decided to just remove the alert because it was causing more problems than it was solving
- I was able to modify the reward function to punish the agent when the new_score went to 0
- Otherwise it returns 0 and +1 when the score increases
- Another new problem, the snake wasn't actually moving
- This was caused because the send_keys() needed to send the action.item() to the actions list when the key names are listed
- At this point, the snake is moving!!! YAY!
- Its actually running! Its a freaking miracle
- except...........
- He's not getting a single apple and he keeps committing suicide
- I just realized my reward function is permanently punishing the agent because when the new_score is 0 it returns -10
- I'll need to fix that....
- I had to add the alert back as it's a good way to see that the game has ended but I have to change the way it's handled
- I'm trying a wait + expected condition alert_is_present
- Then I can do all of the tidying up stuff and punish the agent then accept the alert
- The wait + expected condition try/except didn't work so I'm reverting back to the unexpectedalertexception try/except but putting it inside the main loop
- That seems to be working but the agent isn't going for the apple... it doesn't realize that it's the "goal"
- So I've troubleshot all of my try/excepts, they seem to be working now which is great
- I can expect 50 loops as cuda is not available
- Just got a random attributeError while testing... very unexpected....
- I had to add WebdriverWaits to both the score and the sending keys action because there were instances where the script was trying to access them before the game was ready
- Trying a full run through to test endgame expectations
- Got another error from the sending keys action, I'm modifying the wait to wait until its clickable instead of located...
- That did not fix it I got the error again.... Going to try adding an exception for it that just prints and goes on

- It also didn't seem as mobile with the WebDriverWait on the action element if that makes sense, it was going straight much more often
- Second attempt at a full run through went to almost 40 episodes with 4 ElementNotInteractableErrors. It seems to be happening right at the beginning of a new episode after a failed game
- I got another error for the new score not existing so I have to add a wait there too…
- Made it to 37 and then got an AttributeError… For the love of god why
- Only one interaction error this time though
- I had my first successful full run through
- Only one interaction error and the attribute error didn't crop up this time
- The durations actually got worse over time and the longest episode was the first one
- Looks like I'll need to improve my reward function and/or the model
- Feed newScore to the plot_durations function to plot the score over the course of training
- Modified the plot_durations function to include the score
- Also changed my torch version to include cuda so hopefully training will run more smoothly and I'll run it for longer, starting with 1000 iterations
- I am considering changing the rewards function to reward movement slightly (+1), and a higher reward for getting the apple (+10) with the hope that that will lead to longer durations and therefore higher scores? We'll see, I haven't tried it yet
- Possibly add a line at 0 to show the times when the score goes above 0 OR plot it on a separate sheet
- Had to increase the WebDriverWait time for the score b/c I got a NoneType TypeError…
- But training is running a lot nicer now that I've given it access to the GPU. Lovely
- The AttributeError returned, it's related to an issue with sending keys to the body element. I'll add the WebDriverWait back here and match the wait time to the current score wait time
- Increased the wait time again…. new_score NoneType error this time….
- Got ANOTHER NoneType error
- Decided to look into some other expected_conditions which led me to visibility_of_element_located
- Looked up the difference between that and presence_of_element_located, found out that especially since w the score I need to access innerHTML, I should be using the visibility one
- Changing both score webdriverwaits to visibility and reducing them to their original wait time. I'll increase as needed
- Changing body element webdriverwait to element_to_be_clickable and also reducing the wait time
- Increased score wait time
- Increased again
- These attribute and type errors are completely ruining my life. I have to find a way around them… I don't think increasing the wait time is the answer really

- I made some more changes to the web driver waits and interactions with the score. I'll keep making changes as needed
- I've been working on getting around/fixing the element errors for several hours now. It is not going well
- The element problems have completely derailed my project. I am not finding any concrete solutions and I can't get a full run through
- If I reduced the number of episodes I'd have a better chance of getting a run through but that doesn't solve the problem
- I did modify the graphing though. I created a subplot for the score so I can see both duration and score over the course of the episodes
- I'll do one short run so I can save an image of what my new graphs look like
- I believe I found a more stable way of interacting with the game score
- From this post:
  - How I built an AI to play Dino Run
- I found the driver.execute_script() function and that seems to be working much much better
- I also believe I've found the solution to my issue of the game skipping episodes entirely, I believe it was due to the game starting up again before the alert was closed so I added a try/except for that in the game over exception
- But I'm still having the issue of the alert not triggering the exception from time to time. When this happens the data for that episode is lost so I need to figure out why it's happening
- I'm biting the bullet and setting up a VM for running the script. I'm really hoping that will stop these weird issues related to the alert or other pervasive problems
- VirtualBox doesn't have GPU passthrough… I cry
- None of the windows VMs do…
- My thought process behind putting it in a VM was that hopefully isolating the program would avoid any potential interactions that might be causing some of the "unexplainable" issues I've been experiencing
- I did try closing my other Chrome tabs to see if that was causing issues but the alert issue still happened
- Last ditch, I will put the program in the VM anyways just to see what will happen
- Too many technical roadblocks with the VM situation. I'm going to move on for now
- Since I'm currently passed the date I was aiming for originally, I'm going to focus on improving the agent learning and then sort out the technical problems later
- Did a full run before making changes to the rewards function, my first one apparently
- I'm going to make some changes to the rewards and do a few short runs to see the impact then choose a "best" case and leave it running overnight and hope for the best because 1000 episodes takes like 4 hours

Current rewards:
0 while moving

1 for eating
-10 for game over

I want to try:
1 while moving
10 for eating, 100 for eating
-101 for game over

- Combining 100 for eating, 1 while moving, and -101 for game over seems to give the best results
- I'm going to continue with that for a full run through
- Things to try to fix my learning issues:
    - Increase EPS_END to improve exploration - this could explain my issues with the duration decreasing over time
    - Increase EPS_DECAY to slow down the reduction in exploration
    - Try playing around with replay memory
    - Try adjusting the learning rate, optimizer (AdamW), gamma and tau
    - Look at loss values over time
- Playing around with the above mentioned variables to see the effects

Problems:

- Not sure that the agent is actually learning as intended
- Instances of the alert not triggering the exception... I need to find the cause and address it. The best way would be using print statements so I can see what has happened in the terminal
- Element not interactable error. Need to find a better way around it than the exception