Report

**Test Ya Memory**

* **Inspiration**
  + To make a **fun and time pass game** which can be useful to **develop a good memory.**
  + Playing this game regularly will **increase one’s short-term memory** which is likely to improve their **spontaneous performance and thinking reflexes** while a complex task in a time bound situation.
  + It also helps develop concentration, which is necessary to avoid silly mistakes as one usually makes when in hurry.
* **What It Does**
  + The game has two difficulty levels, **“ez”** and **“hard”.**
  + It is a well-known game of **finding matching cards** and to complete the pair.
  + In the “ez’ level, the matched pair is locked and will remain on display throughout the rest of the game play.
  + In the “hard” level, the game is meant to be finished in one go, i.e., once you find a pair, it will disappear if you mess up the streak. It becomes tough as you will have to remember the location of the pairs while finding new pairs, with this feature, it will also increase you **multitasking ability**.
* **How It was Made**
  + I used two famous libraries of the Python Language, i.e., **Tkinter** and **Pillow**.
  + Tkinter is used to create the front end of the project.
  + Pillow is used to resize the images used in the project as the Tiles in the game.
  + The logic behind the game is quite simple, whenever a tile is clicked, Click function is called. If the tile clicked is a Question Mark, the image at the location will appear.
  + This happens for two consecutive clicks on different tiles until you find a pair.
  + If a pair is not found, the entire deck is reset. (referring the hard mode)
  + A score counter is there to determine the number of clicks you did, which is basically the number of times the Click function was called.
* **Challenges I Faced**
  + I faced many challenges while making this game and I remember only few of them
  + While I was using the grid() attribute to assign the locations to the widget, I was constantly facing an issue displaying that the current widget is of **<class NoneType>.**
  + It took me almost 4 hours to find that using grid() right after the declaration of the widget makes it a NoneType object.
  + It took a lot of time to make the logic work. And I had to do that twice as I wanted to make two difficulty levels.
  + I wanted to make the interface a bit more appealing because I learnt Tkinter for my Python Project and really wanted to add some cool background or buttons, but it was time bound.
  + The challenge while playing the game is mainly that you will be able to see yourself committing silly mistakes again and again. I asked my mum and dad to try it out and it was a really funny experience that they forgot the locations of the pairs so quickly. On an average, **it took every one more than 100 clicks** to get through the hard level.

**(I thought of putting demeaning comments whenever player forgets a matching pair, but I dropped the plan).**