Group Project: Minesweeper

A coding project by Clément Guyot de La Pommeraye and Alexander Meyer

Introduction  
For our group project, we wanted to create a code which we would both enjoy working on and playing. Both of us had an interest in creating a game and we believed that this project task would be the perfect opportunity to try it out. Once we were set on creating a game we had to decide which game would actually suit our interests and our skill level. We had a few options such as tic tac toe, hangman, snake, etc.. Ultimately we decided on creating a Minesweeper Clone due to its interesting visual interface and its underlying game code intricacies.

What is Minesweeper?

According to Wikipedia, Minesweeper is a game where the objective is “*to clear a rectangular board containing hidden "mines" or bombs without detonating any of them, with help from clues about the number of neighbouring mines in each field*”. Minesweeper is thus mainly a game of strategy and intellect (with a bit of chance for the first few field picks).

How to get our game to work

Put all the files in the same folder and run the file main.py. In order to move the cursor you have to use the up and down arrow keys. To select an menu item you have to press enter and to return back to the main menu at any point in the menu you press backspace.

Brief overview of our game  
Once you start the game the menu interface will open and you will be able to view the game menu with “Start”, “Difficulty” and “Credits”. If you select the menu item “Difficulty” you will be given the choice between “Easy”, “Medium”, “Hard” and “Extreme”. By selecting the difficulty you will return back to the main menu automatically. If you then start the game you will play minesweeper with the difficulty you previously selected. Note: You can also start the game without choosing your difficulty (the difficulty “medium” is set by default).

Once the game has started, you have a timer in the top-left corner and the number of mines you have to find (according to the number of mines you flagged) in the top-right corner. Note that if you have flagged too many areas this number will be negative.

Our advice to beat the game

There is always the possibility that you hit a bomb on your first pick. This is why we recommend you start by clicking on the four corners. If you lose there you can restart immediately without losing any time and if you are still in the game you would have eliminated the biggest part of the game linked to luck.

Coding process

At first we had a look at other successful minesweeper clones that were made using python (+pygame). By changing/altering lines of code of these applications we received a more holistic view on how game code actually works. After understanding and comparing different coded versions of Minesweeper we selected the one from Harrelix (check source below), on which our code is largely based. We decided to add the option of selecting levels, which change the colour of the backscreen, the size of the minefield and the number of mines.

Secondly, we created the menu interface which consists of a button starting the game (pre-selected level of hard), one selecting the difficulty and credits. The core difficulty in this part was to actually understand what we were doing. Since we basically created the “Difficulty” Sub Menu from scratch it required us to understand how the main menu code worked together.

Finally, we had to combine the two codes, which was undoubtedly the hardest part. This was primarily due to the fact that you need to really understand what each line of code actually does in order to alter, add or remove elements of your application (including transferring variables from the menu code to the actual game – which was something we were really struggling with).

Difficulties/challenges

It was our first coding project and we had no previous knowledge of Pygame so there were a lot of times when we had to check on the net to understand certain specific situations.

The fact that we were working on this project together was also another challenge as we could not be working on the same code at the same time. This required us to communicate and coordinate better in order to get anything done.

We spent a lot of time trying to figure out how to export variables from one file to the other (especially the variable “level”).

In order to really be able to synchronise the different codes and to add some stuff to it we really needed to understand it in depth which was something harder than expected.

Possible improvements

We could have added an explosion at the end when you lose. However we wanted to create a “live” explosion, like in a video, but we believe that we may not really have the skills to do it and that it would have added too many lines of codes only for a minor part of the game.

We also wanted to create a leader board (times + names) that could be accessed from the menu, but we would have needed to create a database that could be modified and kept when we close the code. This would have been very challenging as we wrote the code in such a way that we restart it each time you launch main.py.

We could also have added more option controls in the menu such as volume, music, light & dark modes,… to make it look more like a “true game” but it was not really part of the project itself and would also have required a lot of lines.

Our thoughts

We really enjoyed working on this project. There were times when we felt completely lost, but that only made it more satisfying when we finally got the code to work. In our opinion the structure of the course gave us the necessary basic knowledge to successfully complete our first python project.

Sources

ChristianD37. (2020, July 24). Menu System. GitHub. Retrieved December 13, 2021 from https://github.com/ChristianD37/YoutubeTutorials/tree/master/Menu%20System

foxyblue. (2020, August 31). Minesweeper. GitHub. Retrieved December 12, 2021 from https://github.com/foxyblue/minesweeper

Harrelix. (2021, June 14). Minesweeper. GitHub. Retrieved November 29, 2021, from https://github.com/Harrelix/Minesweeper

pyGuru123. (2021, January 20). MineSweeper. GitHub. Retrieved December 12, 2021 from https://github.com/pyGuru123/Python-Games/tree/master/MineSweeper