TM Sources

Book: Apprendre la programmation par jeu: Découvrir Pygame avec de nouveaux jeux en Python, Vincent Maille, éd. ellipses

Summary:

This book talks about simple 2D video game creation in detail. This book is separated into many parts/chapters. This book has many game ideas and shows step by step tools and methods to create various games. In addition to all of this, this book has a very educational aspect because it has explanations, as of theory, exercises, and hints for them in case we are not able to figure that out, but it also contains answers and very detailed explanations for it.

The first chapter is not a game, but a revision of how maths work and the logic and skills that are needed to start coding.

But the first actual game is Pong, which is part of the “Classical Games” because of how old, popular, and simple it is. In this game, the book suggests methods and ways of creating our own space, using images, moving with keyboard touches, using dictionaries and so on. With the Pong game, we can take a dive into the Pygame methods and explore it as we like while still completing a working game. But to have an actual game of Pong, you need to use simple artificial intelligence that tracks the ball at its positions and moves accordingly. Moreover, maths is extremely crucial for this game and this chapter also covers how we can use simple trigonometry to have efficient physics. Having sub-parts in the chapter is useful to navigate and find what we need. But sometimes we need to access and use a certain and particular method that we want to call in our code and the book does just that by having a small table that contains few methods that have been used throughout the sub-part and their “meaning”.

The second chapter is also a very famous game entitled “Minos”, which is a maze escape game. Even though many of the starting sub-parts of each chapter are very similar, the result game is completely different. In this game, the reader learns about linked lists and tuples, as well as the concept behind a queue principal. So, in order to create perfect mazes and to actually get out of one, the reader is taught about how to manage the windows, the surfaces, and key events. There are great examples and exercises that help everyone understand the topics that the book is talking about. The key concept of the Minos game being, is the creation of the maze, and in order to make that, the reader gets to learn how to use the lists (and link them together) to create complicated mazes. With the help of functions, dictionaries, lists and more, and of course this book, it is possible to create interactive and fun games.

This next one is more of a mouse pointing game… Selem-Stom, or also known as Crossword. This game is designed to find certain words with letters that are mixed up in a box. In this chapter many of the previous methods and ideas are used, but new things are also added, such as length of a list and groups and so on. With this game, the reader experiences a whole new side of Pygame, not with the keyboard, but with the mouse and its functions that are related to it.

The next chapter is a game that is not part of the classic games or even the “known” games. It is a new 2D concept game that introduces image groups and very interesting topics about Pygame. This game is called “Cupid”, or “Cupidon” in French, what makes this game different and eye-catching is that it has a very fine story behind it even though we would be able to code the game right away with everything that has been mentioned until now in this book. The concept of the whole game is killing enemies that are servants of the devil, that are sent by Jupiter because of his jealousy towards Cupid. As mentioned earlier, this game has many images, per player, for his different movements, and the enemies and their movements, and the arrows and their animations. Even if we would be able to code this game with our current knowledge, the code will be considered too heavy and difficult to understand. To solve this problem, OOP is introduced into the chapter. OOP, also known as Object Oriented Programming, is a very useful, easy, and important method used not only in Python, but in most coding languages. Using this class() method, the person who will be coding can make his life easier by introducing rectangle collisions, as well as fixing the image as a rectangle property, which is the next concept that is introduced. And by always using vectorial mathematics for the movement of the player, the game becomes easier and easier to understand and code. So logic and simplicity are some of the very crucial aspects of coding, especially in Pygame.

All of these games are running on something we call “Front-end”. It is basically the result (the visual representation) of back-end programming (code that we “do not” see the result in a visual form) software. So, this chapter talks about coding and using a game launcher, or game window, but not a specific type of game. So, by using some particular maths (analytical equations), we are able to create circles to ellipsis to some other tangled shapes that are easy to code. A new concept is introduced, and it is the animations, even though in reality it is an illusion of many different screens with different sprites that appear and disappear (for simple one screen animation). And using some other maths, it is possible to change the shapes of the rectangles we create, like for example transforming them into cylinders.

The next game is a second-hand version of the very famous “Pac-Man” called “EHPAD-RUN”. It has roughly the same goals and functions as the original Pac-man, but the yellow circle eater is replaced by a dentist trying to get all the dental glue and being chased by some enemies. With this game, the reader is able to learn and understand how to modify the sizes and rotations of images and so on. By using all these methods, the reader is able to, for example, draw country flags or mazes for the game. A new notion is introduced and is referred to as “heritage” and helps with separating or grouping elements of certain sprites with one and another.

The next chapter is data processing and at first would seem like a boring topic or uninteresting, but in fact can be very useful with Pygame and/or other coding languages. But for the sake of Pygame and the topics that we need in this book, we are going to skip its summary.

The final chapter’s use is to actually be able to code very hard games. And in order to do that, new lessons are taught. How to read and analyse values from a database with Python, the properties of OOP and manage binary code in Pygame. Many examples are given to explain these topics, but they can be used only in specific contexts.

Finally, the book ends with a massive table with every single method that has been used during the entirety of the book, the references for the pages and some brief explanations concerning the methods themselves.