Space Invader

An HTML5 Game by Vineet Yadav

Space Invader is an html5 based game collaborated with JavaScript, CSS, jQuery, Bootstrap.

It consists of enemies falling from the sky, and a player who must kill them by firing bullets to survive in the game. Player can move left and right using the arrow keys and can fire bullets using the space bar, Mouse left key or Up arrow key. Player should make sure that no enemy more than the lives of player touches the ground, else he will lose.

Implementation

Space invader game is implemented using a single .html file backed with few JavaScript files.

HTML File

GamePlay.html file contains all the five pages- Main Menu, Canvas, Settings, Instructions, Game Over, in five different elements. Only one tag can be active at a time.

Page hiding and displaying is controlled by the JavaScript code.

JavaScript Files

1. **canvasScript.js** : It provide all the functionality related to buttons, page-switching, background music, canvas game environment.

2. **jquery.hotkeys.js** : It provides the functionality for binding the keyboard keys to a particular function.

3. **sound.js** : It provide the code to add sound in the game.

4. **sprite.js** : It provide the code to access images from the directory and add them inside the game.

5. **Other JavaScript files**: Files like bootstrap.min.js, feather.min.js, jquery-3.3.1.js, popper.min.js are also included to provide offline working of code.

Source code Organization

GamePlay.html

In GamePlay.html, pages are designed using tag and Bootstrap is used to style the pages according to their specific job.

canvasScript.js

canvasScript.js is the main file that include all the canvas functionality code. Commonly the effect of animation is achieved by a rapid succession of sequential images that minimally differ from each other.

In this file an interval is added that creates this kind of rapid succession. It rapidly clear and redraw the contents of canvas that creates an animation effect.

In this file first code for entities like Player, Bullets, Laser, Rewards, Enemy, Stars and Life is written. These are the entities which are updating in every loop of interval at a decided FPS rate.

After that, functionalities for updating the game environment is written like function for updating player position after resizing the game window, function for game over, function for collision detection, function for updating and redrawing the canvas contents.

After that, code for binding keyboard keys to a particular function is written.

After that, code that will execute after a click on an html button is written. For example, when a user click on play button, settings button or on an exit button a specific lines of code will execute.

Other JavaScript files provide functionalities as their name suggest like sound.js provide sounds to the game, sprite.js provide images to the game and so on.