|  |  |  |  |
| --- | --- | --- | --- |
| **Name** | **Data Type** | **Scope** | **Description** |
| Var alreadyPlayed | Integer | Global | Determines whether the first audio file has already been played or not. |
| Var audioFirst |  | Global | An audio file. |
| Var alreadyExist | Integer | Global | Determines whether a background image is already present or not. |
| Var timeControl | Integer | Global | Determines whether the previous text has fully appeared or not. |
| Var timeManage | Integer | Global | Determines whether another text should replace the current one. |
| Var characterName | Integer | Global | Determines if the name of the character should appear or not, and whether it should be replaced or not. |
| Var characterDisappear | Integer | Global | Determines if the name of the character should disappear or not. |
| Var nextSceneControl | Integer | Global | Determines if the next scene should be called. |
| Var count | Integer | Global | Determines if the next following function should be called. |
| Var audioSecond | String | Global | An audio file. |
| Var timeCount | Integer | Global | Determines how long the processing of the code should be delayed for. |
| Var numberCount | Integer | Global | Represents the current index of the array that should appear. |
| Var numberCount2 | Integer | Global | Represents the current index of the array that should disappear. |
| Var inspection | Integer | Global | Determines whether the following code should be initiated/performed or not. |
| arrCover | Array | Global | Used to record transition images. |
| arrTransition | Array | Global | Used to record transition images. |
| Var createHTML | String | Local | Creates a HTML line to be appended to the HTML page. |
| Var coverID | String | Local | Represents the ID of an element that could be selected. |
| Var secondID | String | Local | Represents the ID of an element that could be selected. |
| Var restriction | String | Local | Represents the ID of an element that could be selected. |
| Var character | String | Local | Represents the ID of an element that could be selected. |
| Var audioFile |  | Local | Represents the ID of any audio file declared in the parameter of the function. |
| arrStudy | Array | Global | Used to record transition images. |
| Var insertHTML | String | Local | Creates a HTML line to be appended to the HTML page. |
| Var audioThird | String | Global | An audio file. |
| Var HTML | String | Local | Creates a HTML line to be appended to the HTML page. |
| Var firstElement | String | Local | Represents the ID of an element that could be selected. |
| Var secondElement | String | Local | Represents the ID of an element that could be selected. |
| Var thirdElement | String | Local | Represents the ID of an element that could be selected. |
| Var fourthElement | String | Local | Represents the ID of an element that could be selected. |
| Var fifthElement | String | Local | Represents the ID of an element that could be selected. |
| Var sixthElement | String | Local | Represents the ID of an element that could be selected. |
| Var displayID | String | Global | Represents an empty string. |
| Var apikey | String | Global | String for storing the APIKEY of the online database. |
| Var url | String | Global | String for storing the URL of the online database. |
| Var settings |  | Local | Combination of Boolean and string. |
| Var characterItem | String | Local | Creates a HTML line to be appended to the HTML page. |
| Var characterCropped | String | Local | Creates a HTML line to be appended to the HTML page. |
| Var serviceURL |  | Local | Represents the combined result of the url plus the ItemID (both are parameters). |
| Var HTML1 | String | Local | Creates a HTML line for the character selected to be appended to the HTML page. |
| Var displayInfo | String | Local | Represents the ID for the identity of the character selected. |
| Var female1 |  | Local | Constructor of the female subclass for the selected character. |
| Var destinationWent | Integer | Global | Used to control the time phase of the day (the destination images) and whether the image should be called. |
| Var className  (currently not used) | String | Local | Represents the ID for the created character. |

Following the latest technological trends within recent decades, modern video games emerges as a prominent aspect of entertainment, ranging across genres including but not limited to sandbox, real-time strategy (RTS), shooters (FPS and TPS) and role-playing games (RPG). Whereby social and ethical concernments regarding the theory of response moralism – the idea of emotions in response to fiction being realistic and morally accessible – and ethical significance arise as a potential issue urgent for consideration. RPG, particularly PC versions, involves the dominant feature of controlling a particular character in which the user identifies themselves as. Followed by purposeful user decisions determining character development and plot progress, often certain moral values were reflected. And behind the morally laden decisions, ethical significance then becomes primarily accentuated. In certain extreme cases, in order to completely reflect the corresponding moral traits identified through given choices, the plotline proceeds at opposite ends. Such progression evokes questioning regarding whether decisions made in a moral situation reflect aspects of personal moral character and whether experiencing moral choices would result in negative influences upon the user. Applying this social and ethical issue, in order to eliminate such unethical influences, this project categorises the meaning behind moral laden decisions as informative. Educating for the social stereotypes and norms of Ancient China within various phases of the history instead for pure entertainment. Even if the undesired choice was chosen, the user would be guided back to the right choice with relevant background information explaining the cause of such result. Hence minimising unwanted ethical significance. In addition to this, issues surrounding ergonomics and particularly interface design also exists within the field of RPG. As more and more research have been conducted regarding user usability and effectiveness of design. The importance of an aesthetically pleasing yet simple to use interface is a targeted consideration when developing software applications and projects. Since the interface represents the connection between the user and the software, therefore an unwanted outcome could potentially lead to problems such as inclusivity. Within the major project, this was minimised as most triggers related to the storyline is able to be initiated through clicking any parts of the webpage screen.

IPO Chart

|  |  |  |
| --- | --- | --- |
| Input | Process | Output |
| Click on screen | Triggers the function for next line | Calls for the next line and corresponding functions |
| Click on bamboo slip | Triggers transition to next line | Calls for the next line |
| Click on beginning (start) button | Deletes irrelevant elements | Clears cache  Calls on following lines of code |
| Click on images belonging to the class of normal background | Transitions to next line of code | Calls for a ‘new’ scene |
| Click on image titled context | Deletes irrelevant elements | Clears cache  Calls on following lines of code |
| Click on confirmation button | Deletes irrelevant elements | Clears cache  Calls on following lines of code |
| Click on street button | Triggers following code | New code gets processed |
| Click on tea house button | Triggers following code | New code gets processed |
| Click on temple button | Deletes irrelevant elements  Triggers following code | Clears cache  New code gets processed |