

Vue.js

Images

Review

- Up until now you have communicated between your html and Vue.js using properties, events, and computed properties
- button clicks were attached to their event handlers using `v-on:click`
- Properties were bound bi-directionally to form elements using `v-model`
- One way binding from a property to a form element was achieved with `v-bind:value`

Images

- This session we will investigate images
- We can bind an image url to an html img tag using `v-bind:src`
- For example:


```

```
- You can also bind other attributes such as height and width


```

```

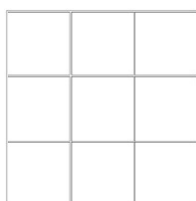
Exercise: Images

- Create a page which has a single button at the top a horizontal line an image
- Initially display the scene.jpg image
- When the button is pressed toggle the scene image with the meerkat image
- Add height and width to the image, 800 x 292 for the scene image and 192 x 254 for the meerkat



Exercise: TicTacToe

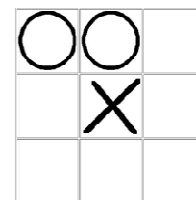
- Create a tictactoe game
- Create a 3 x 3 table with each cell initially displaying the blank.jpg image
- Under the table is a status message saying "Game in progress"
- There are two players X and O and their turn alternates



Game in Progress

Exercise: TicTacToe

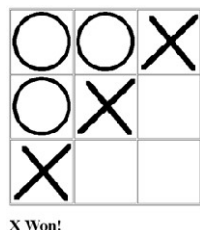
- Each time a blank cell is clicked by a player display in that cell the image for that player
- Clicks on a blank cell should be ignored



Game in Progress

Exercise: TicTacToe

- If a player completes a line horizontally, vertically or on a diagonal then that player has won and an appropriate message should be displayed
- If all 9 cells have been clicked and no one has won the message should be displayed declaring a draw



Behind the scenes

- Declare a property which is an array of 9 integers (initially all 0)
tiles: [0, 0, 0, 0, 0, 0, 0, 0, 0],
- Add an integer or a boolean to indicate whose turn it is
- Add a counter for how many cells have been clicked
- Add a string which will be displayed on the web page

Behind the scenes: Image

- Add a method which is passed an integer with a value from 0 to 8, and which looked up that number in the array to see if it is a 0, 1 or a 2
- If it is a 0 then the corresponding image should be set to a blank, if it is a 1 then set the image to an X, if it is a 2 then set the image to an O.
- Attach this to the v-bind:src attribute for the image passing the image number
``

Behind the Scenes: Click

- Add a onClick (v-on:click) event handler to each of the images on the page which is passed a number from 0 to 8 corresponding to the image
- When the image is clicked check to see what the array value for that image currently is.
- If the value is a 0 then change it to a 1 for the X player, or a 2 for the O player

Behind the Scenes: Array

- If you change the array entry in the usual way
`list[index] = 2;`
Vue will not notice the change, and nothing will happen
- Use this method to change the array entry
`this.$set(this.tiles, index, 2);`

Behind the Scenes: End Game

- After each click increment the counter and check to see if a player has won or if the game has resulted in a draw