

Credit Sussie - Coding Challenge

- Prerequisites

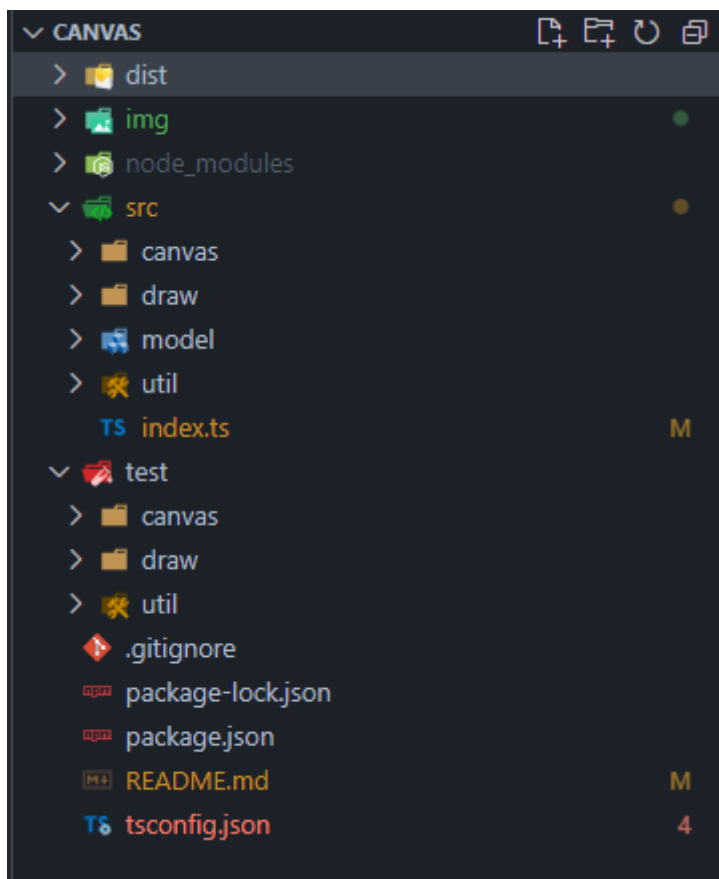
* Node.js

- Project structure

`dist`: Contains the transpiled JavaScript files

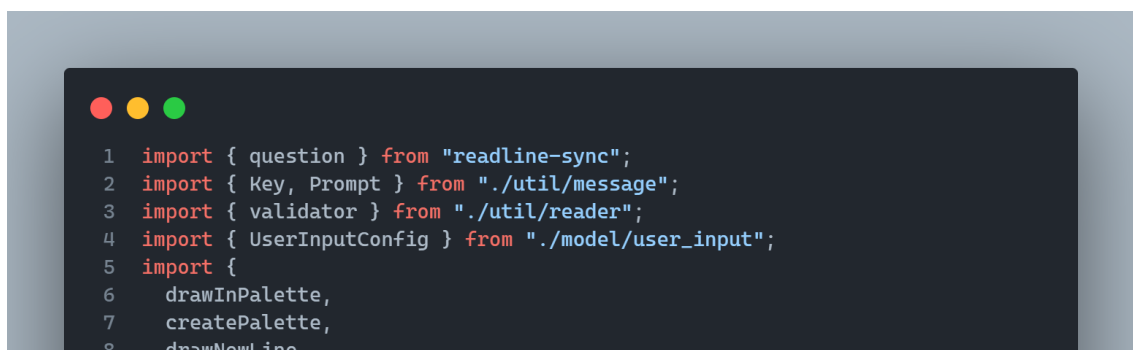
`src`: TypeScript files

`test`: Unit-Test files



- Initializing the project

Project execution starts with `init()` function execution



```

9     drawNewRectangle,
10    fillPalette,
11 } from "../canvas/palette";
12
13 let sessionPaletteData;
14
15 init();
16
17 function init(): void {
18     const command: string = question(Prompt.MSG_ENTER_COMMAND);
19     const input: UserInputConfig = validator(command);
20
21     handler(input);
22     console.log("\n");
23
24     init();
25 }
26
27 function handler(input: UserInputConfig): void {
28     // console.log(input);
29     // console.log("\n");
30
31     // Invalid command
32     if (input.command == Key.INVALID_COMMAND) {
33         console.log(Prompt.MSG_INVALID_COMMAND);
34         return;
35     }
36
37     // Canvas creation
38     if (input.command == Key.DRAW_NEW_CANVAS) {
39         sessionPaletteData = createPalette(input);
40         if (sessionPaletteData == null) {
41             console.log(Prompt.MSG_CANVAS_CREATION_FAILED);
42         } else {
43             drawInPalette(sessionPaletteData);
44         }
45         return;
46     }
47
48     // Draw a line
49     if (input.command == Key.DRAW_LINE && isPaletteSessionAvailable()) {
50         let _data = drawNewLine(input, sessionPaletteData);
51         if (_data == null) {
52             console.log(Prompt.MSG_INVALID_LINE_COORDINATES);
53         } else {
54             sessionPaletteData = _data;
55             drawInPalette(sessionPaletteData);
56         }
57         return;
58     }
59
60     // Draw a rectangle
61     if (input.command == Key.DRAW_RECTANGLE && isPaletteSessionAvailable()) {
62         let _data = drawNewRectangle(input, sessionPaletteData);
63         if (_data == null) {
64             console.log(Prompt.MSG_INVALID_LINE_COORDINATES);
65         } else {
66             sessionPaletteData = _data;
67             drawInPalette(sessionPaletteData);
68         }
69         return;
70     }
71
72     // Fill the canvas with colour
73     if (input.command == Key.BUCKET_FILL && isPaletteSessionAvailable()) {
74         let _data = fillPalette(input, sessionPaletteData);
75         if (_data == null) {
76             console.log(Prompt.MSG_FILL_FAILED);
77         } else {
78             sessionPaletteData = _data;
79             drawInPalette(sessionPaletteData);
80         }
81         return;
82     }
83

```

```

84 // Exit command issued
85 if (input.command == Key.QUIT) {
86     console.log(Prompt.MSG_QUIT);
87     process.exit();
88 }
89 }
90
91 function isPaletteSessionAvailable() {
92     if (sessionPaletteData) {
93         return true;
94     }
95
96     console.log(Prompt.MSG_NO_PALETTE);
97     return false;
98 }

```

• Execution

1. Navigate to project folder

The screenshot shows a VS Code terminal window with the 'Canvas' directory selected. The terminal displays the output of the 'ls' command, showing the directory structure and file details.

Mode	LastWriteTime	Length	Name
d----	11/28/2021 9:48 PM		dist
d----	11/28/2021 10:01 PM		img
d----	11/28/2021 7:34 PM		node_modules
d----	11/28/2021 10:01 PM		src
d----	11/28/2021 9:02 PM		test
-a---	11/28/2021 7:03 PM	88	.gitignore
-a---	11/28/2021 7:34 PM	75599	package-lock.json
-a---	11/28/2021 7:44 PM	1004	package.json
-a---	11/28/2021 10:54 PM	86	README.md
-a---	11/28/2021 7:37 PM	190	tsconfig.json

2. To build:

```
> npm run build
```

```
Canvas master +1 ~2 npm run build

> canvas_drawing@1.0.0 build
> npm test && tsc

> canvas_drawing@1.0.0 test
> mocha -r ts-node/register test/**/*.spec.ts

Check Palette Creation Function
  ✓ should return a multidimensional array
  ✓ should return null if both width and height is zero
  ✓ should return null if atleast width or height is zero

Check Filling Function
  ✓ should return a multidimensional array

Check Creation of line Function
  ✓ should return a multidimensional array of the palette

Check Creation of Rectangle Function
  ✓ should return a multidimensional array of the palette

Check User Input Validator - Success Scenarios
  ✓ Successfully Accpected Quit Command
  ✓ Successfully Accpected Create Canvas Command
  ✓ Successfully Accpected Draw Line Command
  ✓ Successfully Accpected Draw Rectangle Command
  ✓ Successfully Accpected Fill Canvas with Colour Command

Check User Input Validator - Failure Scenarios
  ✓ Invalid Quit Command
  ✓ Successfully Accpected Create Canvas Command
  ✓ Successfully Accpected Draw Line Command
  ✓ Successfully Accpected Draw Rectangle Command
  ✓ Successfully Accpected Fill Canvas with Colour Command

16 passing (20ms)

Canvas master +1 ~2
```

3. To run the Unit-Tests:

```
> npm test
```

```
Canvas master +1 ~2 npm test

> canvas_drawing@1.0.0 test
> mocha -r ts-node/register test/**/*.spec.ts

Check Palette Creation Function
  ✓ should return a multidimensional array
  ✓ should return null if both width and height is zero
  ✓ should return null if atleast width or height is zero

Check Filling Function
  ✓ should return a multidimensional array

Check Creation of line Function
  ✓ should return a multidimensional array of the palette

Check Creation of Rectangle Function
  ✓ should return a multidimensional array of the palette

Check User Input Validator - Success Scenarios
  ✓ Successfully Accepted Quit Command
  ✓ Successfully Accepted Create Canvas Command
  ✓ Successfully Accepted Draw Line Command
  ✓ Successfully Accepted Draw Rectangle Command
  ✓ Successfully Accepted Fill Canvas with Colour Command

Check User Input Validator - Failure Scenarios
  ✓ Invalid Quit Command
  ✓ Successfully Accepted Create Canvas Command
  ✓ Successfully Accepted Draw Line Command
  ✓ Successfully Accepted Draw Rectangle Command
  ✓ Successfully Accepted Fill Canvas with Colour Command

16 passing (20ms)

Canvas master +1 ~2
```

4. Enter following command

```
> npm start
```

5. Enter commands to prompt

```
Canvas master ~1 npm start

> canvas_drawing@1.0.0 start
> tsc && node dist/index.js

enter command:
```

6. To create a canvas:

```
> c 20 4
```

Height: 4

```
> canvas_drawing@1.0.0 start  
> tsc && node dist/index.js
```

enter command: C 20 4

enter command:

7. To draw a line:

> L 1 2 6 2

x1: 1

 $y_1: 2$

x2: 6

 $y_2: 2$

```
enter command: L 1 2 6 2  
-----  
|                      |  
|xxxxxxx              |  
|                      |  
|                      |  
|                      |  
-----
```

```
enter command:
```

8. To draw a rectangle:

```
> R 14 1 18 3
```

x1:14

 $y_1: 1$

x2: 18

$y_2: 3$

```
enter command: R 14 1 18 3
-----
|               xxxxx |
| xxxxxx      x   x  |
|               xxxxx |
|               |
|               |
-----

enter command:
```

9. To fill the canvas with colour:

> B 10 3 0

x1: 10

y1:3

$$\text{C}:\text{O}$$

```
enter command: B 10 3 o
-----
|ooooooooooooooooxxxxxoo|
|xxxxxxxxooooooooox   xoo|
|ooooooooooooooooxxxxxoo|
|ooooooooooooooooooooooooo|
-----
enter command:
```

10. To quit:

 $\gamma > Q$

```
enter command: Q
drawing ends.
```

Repository.