1. Build a Drug Store Ecommerce website

\*) Use MERN stack to build the website

\*) Set up a MongoDB or any NO-SQL DB

\*) Set a Node js Express Server

\*) Create a React.JS frontend

\*) Create three Collections namely User and Products and order history in MongoDb

\*) Upload the relevant data to those tables

\*) Users are already registered with email and password

\*) Make sure you hash the password and store it in DB

\*) Build a login page :- A user can login using emailId and password with usual validations

\*) Build a profile page :- Display all user information. A user can update his information. Make sure that all the emails are unique.

\*) If the credit card field is blank show ask the user to input the credit card number

\*) Upon login a user is displayed all the Drugs name and their company name

\*) User is displayed Item detail page where price and current stocks are displayed

\*) Have a mock add to cart button. Upon clicking that button stock for that product is depleted

\*) Build an order history page that shows all the order history and stock of the product when it was ordered

1. Given a 2D screen, location of a pixel in the screen and a color, replace color of the given pixel and all adjacent same colored pixels with the given color.

screen[M][N] = {{1, 1, 1, 1, 1, 1, 1, 1},

{1, 1, 1, 1, 1, 1, 0, 0},

{1, 0, 0, 1, 1, 0, 1, 1},

{1, 2, 2, 2, 2, 0, 1, 0},

{1, 1, 1, 2, 2, 0, 1, 0},

{1, 1, 1, 2, 2, 2, 2, 0},

{1, 1, 1, 1, 1, 2, 1, 1},

{1, 1, 1, 1, 1, 2, 2, 1},

};

x = 4, y = 4, newColor = 3

The values in the given 2D screen indicate colors of the pixels.

x and y are coordinates of the brush, newColor is the color that

should replace the previous color on screen[x][y] and all surrounding

pixels with same color. You can only move down and right

Output:

Screen should be changed to following.

screen[M][N] = {{1, 1, 1, 1, 1, 1, 1, 1},

{1, 1, 1, 1, 1, 1, 0, 0},

{1, 0, 0, 1, 1, 0, 1, 1},

{1, 3, 3, 3, 3, 0, 1, 0},

{1, 1, 1, 3, 3, 0, 1, 0},

{1, 1, 1, 3, 3, 3, 3, 0},

{1, 1, 1, 1, 1, 3, 1, 1},

{1, 1, 1, 1, 1, 3, 3, 1},

};