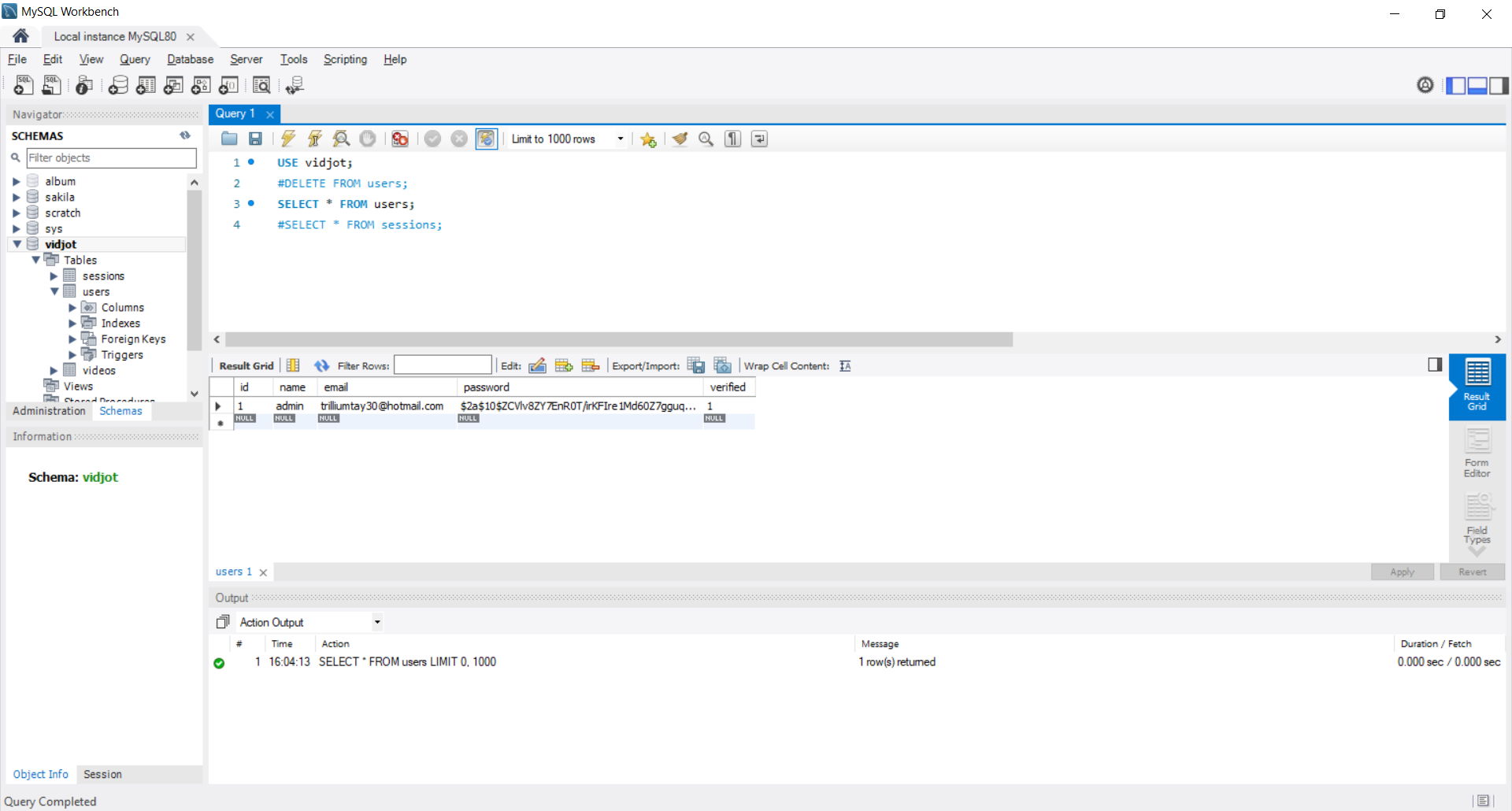
Problems

1. How to differentiate Users and Admins
   1. Create ‘Admin’ Column that accepts True and False values,
      1. check on every page whether he’s an admin or not
      2. Somehow store in session? Questions regarding how safe session is (unsure how session works, question 4)
         1. Practical 3 Activity 3 – app.js line 83
      3. Store it in database???

Passport – Google strategy

Use req.user.userType attribute, that stores ‘Admin’, ‘Default’

1. How would cart attributes be stored



* Cart **MUST** contain furniture/product Id, and each items quantity, so
  + {‘furnitureId’: 1, ‘quantity’:1} object
* Cart attributes should be permanent, so stored somewhere in database, it has to be in
  + Users Table
    - If users table, the entire list of objects would be stored in string, and retrieved back the same way(?) (question on converting string and objects directly)
    - Retrieved by selecting cart where userId == req.userId
  + Or Cart Table (user -> cart / 1..\* relationship, so cart contains userId foreignkey) \*USE THIS
    - Consists of [furnitureId, quantity, userId]
    - This means for every cart item stored there will be another row, will take up a lot of data for each row created
    - Retrieved by selecting all cart rows where userId == req.userId

Consider  
Shopping Cart and all prices   
<https://www.npmjs.com/package/@ecomplus/shopping-cart>

<https://github.com/ecomplus/utils>

<https://developers.e-com.plus/utils/ecomUtils.html>

Might be useful Methods:  
findByName – get object from list of objects based on ‘name’ key  
formatDate – But we already have our own formatDate in hbs.js and npm ‘moment’