Using Views
(**Programming tip:** drop view if it exists, <u>before</u> creating each view)

Log in to *YOUR* database using MySQL Workbench, <u>AND</u> through SSH.

<mark>***First: drop *ALL* existing tables!*** Then…</mark>

source db/gmc.sql

--or...

 \. db/gmc.sql

Reports:
<mark>**Note:** All currency values should be formatted with $ sign, including 2 decimal places. All zip codes and phone numbers should include hyphens.</mark>

1. Create a view that only displays vehicle's types, makes, models, and years in ascending order of years (name it v_veh_info).

2. Create a view that only displays customers' first names, last names, and balances, in descending order of balances (name it v_cus_info).

3. Create a view that only displays sales reps' first names, last names, and commissions, for those sales reps' who have total sales of at least $500,000, in ascending order of last names, (these sales reps' commissions should display a 10% bonus over their stored values, name it v_srp_comm). Show commission values before/after calling view.

4. Create a view that only displays dealership id's, phone numbers, zip codes, and year-to-date sales, for dealerships beginning with zip code "981," in descending order of year-to-date sales (name it v_dlr_ytd).

5. Create a view that displays the types of vehicles, and the number of vehicles for each type (name it v_veh_type). Can use alias columns instead (e.g., type, num)

6. Create a view that displays all <u>purchased</u> vehicle information, with the vehicle's price, in ascending order of price (name it v_veh_info_all).

7. Create a view that only displays sales reps' first names, last names, and ages, in descending order of ages (name it v_srp_info).

8. Create a view that displays dealerships' ytd_sales, ytd_sales commissions of 5% (unstored derived attribute), for dealerships with ytd_sales of more than $1,000,000 (name it v_dlr_comm):