Data Link: https://www.kaggle.com/datasets/spscientist/students-performance-in-exams/data

Here are 8 fundamental coding questions involving a dataset in R that students who have just learned the syntax should be able to do. Problems that are highlighted you may not be able to answer until further in the course and can currently skip. These questions are designed to help students practice basic data manipulation and analysis tasks using the R programming language:

1) **Load a Dataset:** Ask students to load a dataset from a CSV file (e.g., using **read.csv**) into R.
   a) Please download and read in the dataset from the above link. Please be sure that your first row of data does not become the column names.
2) **View the First Rows:** Have students display the first few rows of the dataset to get a quick overview of its structure.
   a) Using the head() function, visualize the first few rows of the data.
   b) Make appropriate univariate plots for each of the data variables
   c) Choose three separate pairs of variables, and view the joint distribution of the data
3) **Summary Statistics:** Ask students to calculate and print summary statistics (mean, median, min, max) for a specific column in the dataset (e.g., using **summary** or **quantile**).
   a) Find the 50th, 90th, and 99th percentile for each of the grade score variables
4) **Create New Variables:** Have students create a new variable that's a transformation of existing variables using if/else statement (e.g., creating a new column that's the sum of two existing columns).
   a) Turn the "parental.level.of.education" variable into an ordered factor variable using **factor.**
   b) Create a new variable called "agg_score". The variable will be constructed as such:
      i) 0 if the student did not score in the top 60th percentile for any of the subjects
      ii) 1 if the student scored in the top 60th percentile for exactly 1 course
      iii) 2 if the student scored in the top 60th percentile for exactly 2 courses
      iv) 3 if the student scored in the top 60th percentile for all courses.
   c) Select two variables, that do not include grade, and create a plot for each of the joint distributions with "agg_score"
5) **Filter Data:** Have students filter the dataset to extract rows that meet specific criteria using **which** (e.g., filtering data for a specific category).
   a) Create two separate data sets
      i) Students who did not score in the top 60th percentile for any subject
      ii) Students who scored in the top 60th percentile for at least one subject
   b) Whichever variables you chose to view the joint distribution of in 4) with the created variable, view them again univariately for each created data set with the plot of your choice
      i) Did you notice anything with the variables between the two data?
6) **Sorting Data:** Challenge students to sort the dataset based on a particular variable (e.g., using **order**).
   a) Using the sort() function, print out the values of every variable for the person who scored the highest in each subject separately

        i) Pick a separate grade variable to break all tie breakers

    b) Using the sort() function, print out the values of every variable for the person who scored the lowest in each subject separately

        i) Pick a separate grade variable to break all tie breakers

7) **Export Data:** Have students save a subset of the dataset or the entire dataset to a new CSV file (e.g., using **write.csv**).
    a) Subset the data to those who have free/reduced lunch and export it to a csv file.
8) **Encourage Scientific Curiosity:** Encourage the students to ask their very own questions within the scope of the data, and to provide insights.
    a) After working with the data extensively, create another reasonable variable you think may be interesting to look at and explore. To be clear, you must deeply explore the variable as has been done throughout the document via multiple plots and analysis against several different variables.