Janitor (Data Cleaning and Tidying):

1. You have a messy dataset with column names containing spaces, special characters, and uppercase letters. How can you use the janitor package to clean and standardise these column names?
2. Your dataset has multiple variables with missing values. Using janitor, what function would you use to calculate the total number of missing values in each column?
3. You need to reshape your dataset from wide to long format, particularly for a survey data with multiple questions as columns. Which janitor function can help you achieve this transformation?
4. You want to reorder the rows in your dataset by a specific variable's values. How can you use janitor to reorder rows based on a numeric or character variable?
5. Your dataset contains duplicate rows, and you want to remove them. What janitor function should you use to eliminate duplicate records?
6. Suppose you have a dataset with a column named "gender" that contains values like "M" and "F." How can you use janitor to recode these values to "Male" and "Female"?
7. You have a dataset with a datetime column, and you want to extract the year, month, and day as separate columns. Which janitor function can help you achieve this?

Forcats (Categorical Data Handling):

1. You have a categorical variable, "Education_Level," with multiple levels. How can you use forcats to reorder the levels based on the frequency of each level in your dataset?
2. You want to create a bar plot of a categorical variable to visualize the distribution of its levels. How can you use forcats to reorder the levels for a meaningful visualization?
3. Your dataset has a categorical variable with a large number of levels. You want to group some of the less frequent levels into a single "Other" category. What forcats function can help you achieve this?
4. You're working with a dataset that includes a "Rating" column, and you want to create a new categorical variable, "Rating_Category," based on specific cutoff values. How can forcats assist in this task?
5. You have a dataset with a column that contains messy text data, and you want to standardize the capitalization for consistent analysis. How can forcats help you achieve this?
6. Your dataset includes a categorical variable that represents days of the week. How can you use forcats to set a custom order for the levels so that the days are ordered according to your preference?
7. You need to reorder the levels of a categorical variable so that they appear in a specific order you define. What forcats function allows you to specify a custom order for the levels?
8. You're working with a dataset that has a categorical variable, "Customer_Type," with some levels that are synonyms for each other. How can forcats help you consolidate these synonymous levels into a single category?

Answers:

1. To clean and standardize column names with spaces, special characters, and uppercase letters, you can use the clean_names() function from the janitor package.
2. To calculate the total number of missing values in each column, you can use the missing_stats() function from janitor.
3. To reshape a dataset from wide to long format, you can use the gather() function from janitor.
4. To reorder rows based on a specific variable's values, you can use the arrange() function from the dplyr package, not janitor.
5. To remove duplicate rows from a dataset, you can use the get_dupes() function from janitor.
6. To recode values in a column, you can use the recode() function from the dplyr package, not janitor.
7. To extract the year, month, and day from a datetime column, you can use the ymd(), year(), month(), and day() functions from the lubridate package, not janitor.

Forcats (Categorical Data Handling):

1. To reorder the levels of a categorical variable based on the frequency of each level, you can use the fct_infreq() function from the forcats package.
2. To reorder the levels of a categorical variable for a meaningful visualization, you can use the fct_reorder() function from the forcats package.
3. To group less frequent levels into a single "Other" category, you can use the fct_lump() function from the forcats package.
4. To create a new categorical variable based on specific cutoff values, you can use the cut() function from base R or other packages like dplyr.
5. Forcats is not typically used to standardise capitalization. You can use functions like tolower(), toupper(), or str_to_title() from the stringr package for this purpose.
6. To set a custom order for the levels of a categorical variable, you can use the fct_relevel() function from the forcats package.
7. To specify a custom order for the levels of a categorical variable, you can use the fct_relevel() function from the forcats package.
8. To consolidate synonymous levels into a single category, you can use the fct_collapse() function from the forcats package.