1. Execute the query "SELECT DATE(date\_created) as sales\_date, SUM(amount) as total\_sales FROM sales\_list GROUP BY DATE(date\_created) ORDER BY SUM(amount) DESC, DATE(date\_created) ASC" using prepared statements and fetch the result.

2. Create an empty array `$sales\_data\_all\_time` to store the sales data.

3. Iterate over the result using a `while` loop:

- Assign the `total\_sales` value to `$sales\_data\_all\_time` with the `sales\_date` as the key.

4. Calculate the total sales for all time:

- Use `array\_reduce()` on `$sales\_data\_all\_time` to sum up all the sales values.

5. Calculate the all-time high and low sales:

- Use `max()` and `min()` functions on `$sales\_data\_all\_time` to find the maximum and minimum values.

6. Sort the sales data array in descending order by key using `krsort()`.

7. Get the latest month's sales data:

- Assign the first key of `$sales\_data\_all\_time` to `$current\_month\_date`.

- Retrieve the sales amount for the `$current\_month\_date`.

8. Calculate the average sales for all time:

- Divide the total sales by the number of entries in `$sales\_data\_all\_time`.

9. Calculate the current year's sales:

- Initialize `$current\_year\_sales` to zero.

- Iterate over `$sales\_data\_all\_time`:

- Check if the first four characters of the date match the first four characters of `$current\_month\_date`.

- If true, add the sales amount to `$current\_year\_sales`.

- If false, break the loop.

10. Generate a suggestion based on the sales data:

- Initialize `$suggestion` as an empty string.

- Use conditional statements to determine the suggestion based on the comparisons between sales values.

- Set `$suggestion` accordingly.

11. Generate a sales message:

- Create the `$sales\_message` string by concatenating various sales-related information and the suggestion.

12. Output the sales message:

- Echo the `<div>` element with the ID "sales-suggestion" containing the formatted `$sales\_message`.