PS10P1:
Input:

- User's choice to continue the program (Yes or No)
- Last name (if the choice is Yes)
- Month (if the choice is Yes)
- Sales (if the choice is Yes)

Process:

- Check if the user wants to continue the program
- If Yes, get last name, month, and sales as input
- Pass month and sales to the nextMonthsForecast function
- Determine the forecast percent based on the month
- Calculate next month's sales using the formula: sales x (1 + forecast percent)
- Return the next month's sales
- Display the next month's sales for the user
- If No, exit the program

Output:

- Next month's sales forecast for the given last name
- Exit message when the user chooses to stop the program
- Error messages for invalid input (if any)

PS10P2:
Input:

- User's choice to continue the program (Yes or No)
- Length of the room (if the choice is Yes)
- Width of the room (if the choice is Yes)
- Height of the room (if the choice is Yes)

Process:

- Check if the user wants to continue the program
- If Yes, get the length, width, and height of the room as input
- Pass the length, width, and height to the computeSquareFootage function
- Calculate the square footage of the room using the formula: 2 x length x width (floor and ceiling) + 2 x length x height (2 of the walls) + 2 x width x height (the other 2 walls)
- Return the square footage of the room
- Calculate the number of gallons of paint needed to cover the room using the formula: square footage / 50
- Display the number of gallons needed
- If No, exit the program

Output:

- Number of gallons of paint needed to paint the room
- Exit message when the user chooses to stop the program
- Error messages for invalid input (if any)

PS10P3:
Input:

- User's choice to continue the program (Yes or No)
- Make of the automobile (if the choice is Yes)
- Model of the automobile (if the choice is Yes)
- Electric vehicle code (Y or N) (if choice is Yes)
- MSRP (sticker price) of the automobile (if the choice is Yes)

Process:

- Check if the user wants to continue the program
- If Yes, get the make, model, electric vehicle code, and MSRP as input
- Pass the MSRP, make, model, and electric vehicle code to the outTheDoorPrice function
- Determine the percent off MSRP based on the make, model, and electric vehicle code
- Calculate the new MSRP using the formula: MSRP * (1 - percent off MSRP)
- Add 7% sales tax to the new MSRP to get the total price
- Return the total cost and new MSRP
- Display the door price for the automobile
- Keep track of the sum of all MSRPs and the sum of all sales prices
- If No, exit the program

Output:

- Out-the-door price for each automobile
- Total MSRP of all automobiles
- The total sales price of all automobiles
- Exit message when the user chooses to stop the program
- Error messages for invalid input (if any)

PS10P4:
Input:

- User's choice to continue the program (Yes or No)
- Last name (if the choice is Yes)
- Miles from downtown Chicago (if the choice is Yes)

Process:

- Check if the user wants to continue the program
- If Yes, get the last name and miles from downtown Chicago as input
- Pass the miles from downtown Chicago to the computeTicketPrice function
- Determine the ticket price based on the miles from downtown Chicago
- Return the ticket price

- Display the train ticket price for the user
- Keep track of the sum of all ticket prices
- If No, exit the program

Output:

- Train ticket price for each user
- The total price of all tickets
- Exit message when the user chooses to stop the program
- Error messages for invalid input (if any)

PS10P5:
Input:

- User's choice to continue the program (Yes or No)
- County (if the choice is Yes)
- The market value of a home (if the choice is Yes)

Process:

- Check if the user wants to continue the program
- If Yes, get the county and market value of the home as input
- Pass the county and market value to the computeAssessedValue function
- Determine the assessed value percent based on the county
- Compute the assessed value using the formula: market value * assessed value percent
- Return the assessed value
- Display the assessed value for the home in the given county
- Keep track of the sum of all market values and assessed values
- If No, exit the program

Output:

- The assessed value for each home in the given county
- The total market value of all homes
- The total assessed value of all homes
- Exit message when the user chooses to stop the program
- Error messages for invalid input (if any)