Initialize history as an empty list  
  
Function clear\_screen():  
If OS is Windows:  
Clear console screen using "cls"  
Else:  
Clear console screen using "clear"  
  
Function digital\_loader(message):  
For 3 iterations:  
Print message with progressive dot  
Sleep for 0.4 seconds  
Print message with "Done!"  
  
Function print\_header(title):  
Clear screen  
Print a stylized header with a separator and title centered  
  
Function delayed\_print(label, value):  
Sleep for 0.8 seconds  
Print label and value in a formatted manner  
  
Function display\_history()  
Call print\_header with "History of Sessions"  
If history is NOT empty:  
Print: "No previous sessions found."  
Else: Iterate over each session in history.  
Print session details: numbers entered, total sum, even/odd counts and sums  
Wait for user input before returning to main menu  
  
Function process\_numbers():  
Loop until user wants to return to the main menu:  
Call print\_header with "Welcome to the Digital Processor"  
Initialize empty list 'numbers'

Ask user to input 5 integers:  
For each of the 5 numbers:  
If user input is not a valid integer, prompt again  
Add the valid input to 'numbers'  
Call digital\_loader with "Processing numbers"  
Calculate total sum of numbers  
Initialize variables for odd/even counts and sums

For each number in 'numbers':  
If the number is even:

update even count and sum  
Else:

update odd count and sum

Call print\_header with "Processing Complete"  
Display summary results (numbers, sum, even/odd details)  
Add session data (numbers, total sum, counts, and sums) to history

Ask user if they want to continue:  
If '1' is selected, reload for new input  
If '2' is selected, return to main menu  
Else, print error and break  
  
Main program loop:  
Call print\_header with "MAIN MENU"  
Display main menu options (Analyze Numbers, View History, Exit)  
Prompt user for choice:  
If '1' is selected, call process\_numbers  
If '2' is selected, call display\_history  
If '3' is selected, print exit message and break from the loop  
If invalid choice, print error and prompt again