LAB TASK – 7

1. Write a program in assembly language to take a single-digit integer from the user and print it on the screen.

CODE:

```
ORG 100h        ; Origin, to specify that the program starts at 100h (COM file format)

; Display message "Enter a digit: "
MOV DX, OFFSET msg_input  ; Load the address of the input message
MOV AH, 09h       ; Function 09h of INT 21h is used to display a string
INT 21h         ; Call DOS interrupt to print the message

; Read a single character from the user
MOV AH, 01h       ; Function 01h of INT 21h is used to read a character
INT 21h         ; Call DOS interrupt to get the character
MOV BL, AL       ; Store the input character in AL (This can be removed, keeping it as is)

; Check if the character is a digit (0-9)
CMP AL, '1'       ; Compare AL with '0'
JL NotDigit       ; If the input is less than '0', it is not a valid digit
CMP AL, '9'       ; Compare AL with '9'
JG NotDigit       ; If the input is greater than '9', it is not a valid digit

; Print the message "The entered digit is: "
MOV DX, OFFSET msg_output  ; Load the address of the output message
MOV AH, 09h       ; Function 09h of INT 21h is used to display a string
INT 21h         ; Call DOS interrupt to print the output message
```

```
; Print the entered digit
MOV DL, BL       ; Move the entered digit to DL (for printing)
MOV AH, 02h      ; Function 02h of INT 21h is used to print a single character
INT 21h          ; Call DOS interrupt to print the character
JMP EndProgram    ; Jump to the end of the program

NotDigit:
; If the input is not a valid digit, display an error message
MOV DX, OFFSET msg_error  ; Load the address of the error message
MOV AH, 09h      ; Function 09h of INT 21h is used to display a string
INT 21h          ; Call DOS interrupt to print the error message

EndProgram:
; Terminate the program
MOV AH, 4Ch      ; Function 4Ch of INT 21h terminates the program
INT 21h          ; Call DOS interrupt to exit

msg_input  DB 'Enter a single  digit: $'             ; Input message
msg_output DB 0Dh, 0Ah, 'The single digit is: $'  ; Output message
msg_error  DB 0Dh, 0Ah, 'Error: Not a digit! $'   ; Error message

END              ; End of program
```

OUTPUT::::

2. Write a program in assembly language to take two single-digit integers from the user and print the result of subtraction on the screen.

CODE:

```
ORG 100h


_start:
  ; Prompt for the first digit
  MOV DX, OFFSET msg_input1
  MOV AH, 09h
  INT 21h

  ; Get the first digit from the user
  MOV AH, 01h
  INT 21h
  CMP AL, '0'       ; Check if input is less than '0'
```

```
JL InvalidInput     ; Jump to error if input is not valid
CMP AL, '9'         ; Check if input is greater than '9'
JG InvalidInput     ; Jump to error if input is not valid
SUB AL, '0'         ; Convert ASCII to number
MOV BL, AL          ; Store first digit in BL

; Prompt for the second digit
MOV DX, OFFSET msg_input2
MOV AH, 09h
INT 21h

; Get the second digit from the user
MOV AH, 01h
INT 21h
CMP AL, '0'         ; Check if input is less than '0'
JL InvalidInput     ; Jump to error if input is not valid
CMP AL, '9'         ; Check if input is greater than '9'
JG InvalidInput     ; Jump to error if input is not valid
SUB AL, '0'         ; Convert ASCII to number
MOV BH, AL          ; Store second digit in BH

; Print the result message and move to next line
MOV DX, OFFSET msg_result
MOV AH, 09h
INT 21h
```

```
    ; Subtract the two digits (BL - BH)
    SUB BL, BH

    ; Check if the result is negative
    JS NegativeResult

    ; Convert the result to ASCII and display
    ADD BL, '0'
    MOV DL, BL
    MOV AH, 02h
    INT 21h
    JMP EndProgram

NegativeResult:
    ; If result is negative, print '-' first
    MOV DL, '-'
    MOV AH, 02h
    INT 21h

    ; Convert the absolute value of result to ASCII and display
    NEG BL
    ADD BL, '0'
    MOV DL, BL
    MOV AH, 02h
    INT 21h
    JMP EndProgram
```

```
InvalidInput:
   ; Display error message
   MOV DX, OFFSET msg_error
   MOV AH, 09h
   INT 21h
   JMP EndProgram

EndProgram:
   ; Terminate the program
   MOV AH, 4Ch
   INT 21h

msg_input1  DB 'Enter first digit: $'           ; Input message for the first digit
msg_input2  DB 0Dh, 0Ah, 'Enter second digit: $'  ; Input message for the second digit
msg_result  DB 0Dh, 0Ah, 'The result is: $'      ; Output message (newline before result)
msg_error   DB 0Dh, 0Ah, 'Error: Invalid input! $' ; Error message

END
```

OUTPUT::::


emulator screen (80x25 chars)
Enter first digit: 7
Enter second digit: 2
The result is: 5
clear screen
change font