**PRACTICAL # 09**

**OBJECT:**

JDBC Connectivity and Operations

**THEORY:**

Java connects to different databases through the DB connectors or drivers. Such drivers are available for major DBMS like Oracle, MySQL, SQLServer etc.

In this lab, we’ll work with MySQL database connectivity and performing CRUD operations.

**Program:**

.model small

.stack 100d

.data

.code

main proc

mov bx, 5

mov ax, 'a'

push bx

push ax

call printArg

mov ah, 2

mov dl, 'z'

int 21h

ret

main endp

printArg proc

push bp

mov bp, sp ; store sp current value for relative addressing

sub sp, 2 ; reserve 2 bytes/1 words on stack for local variables

; save the registers values on stack before use

push ax

push bx

push dx

mov [bp-1], 2 ; initialize local variable

mov [bp-2], 3

mov dx, [bp+6] ; access first argument

add dx, 48

mov ah, 2

int 21h

mov dx, [bp+4] ; access second argument

int 21h

mov al, [bp-1] ; access local vaiable1

mov bl, [bp-2] ; access local vaiable1

mul bl

mov dl, al

add dl, 48

mov ah, 2

int 21h

;restore the registers original values

pop dx

pop bx

pop ax

add sp, 2 ; unallocate reserved bytes on stack

pop bp ; restore bp original value

ret ; restore ip to caller function

printArg endp

end main

**ACTIVITIES**

**Activity 1**

Write a simple function that gets two numbers as arguments and returns their sum in a register.

**Activity 2**

Find out a way to print the string in reverse order and write a program to that.

**Activity 3**

Write a program that finds sum of elements of an array of size three.

**REVIEW QUESTIONS**

1. What is the purpose of SP register and SS register in stack management?
2. In which order does the stack grow as data is pushed into it?
3. What will be the stack overflow and underflow conditions?
4. How are the parameters to a function accessed?

1. How are local variables created and accessed in a function?
2. What are the function **calling conventions**?

1. How is the current register data saved before using the register in some called function?