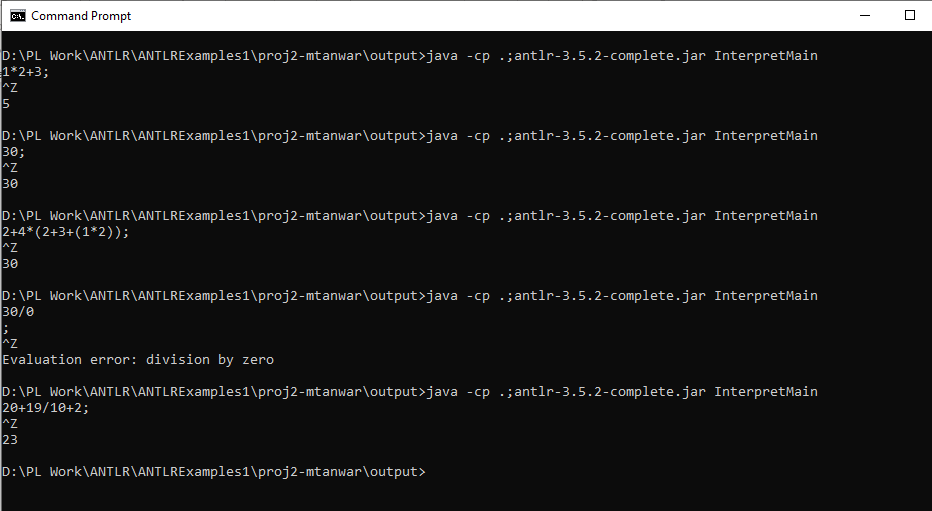
**Test Cases for Project Part2**

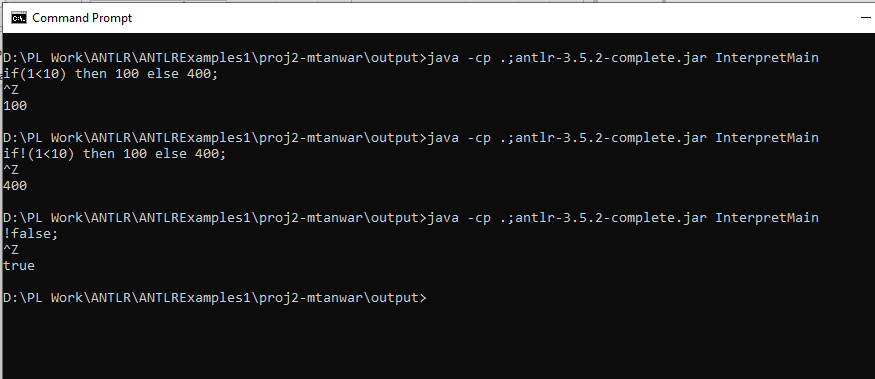
**Case1: Arithmetic expression**

1. **1\*2+3;**
2. **30;**
3. **2+4\*(2+3+(1\*2));**
4. **30/0**
5. **20+19/10+2;**



**Case 2: Not Expression**

1. **if!(1<10) then 100 else 400;**
2. **!false;**



**Case 3: Relational Expression**

(i) let val x=10

in

let fun foo(x)=

if(x<10) then 100 else 500

in foo(20)

end

end;

(ii)

let val x=10

in

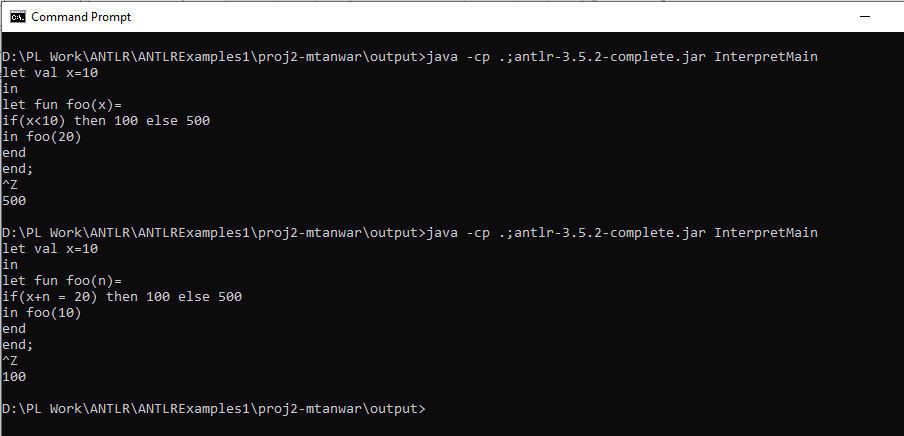
let fun foo(n)=

if(x+n = 20) then 100 else 500

in foo(10)

end

end;



**Case 4: If then else expression**

(i) let val a=1

in

if(a<5) then 200 else 500

end;

let fun foo(n)=

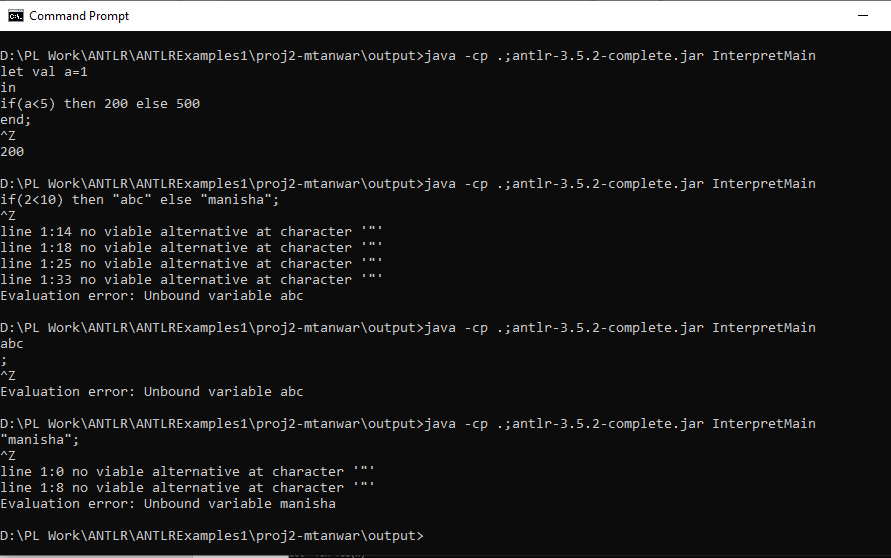
if (! (n< 10)) then 100 else 200

in

foo(10)

end;

(ii) if(2<10) then "abc" else "manisha";



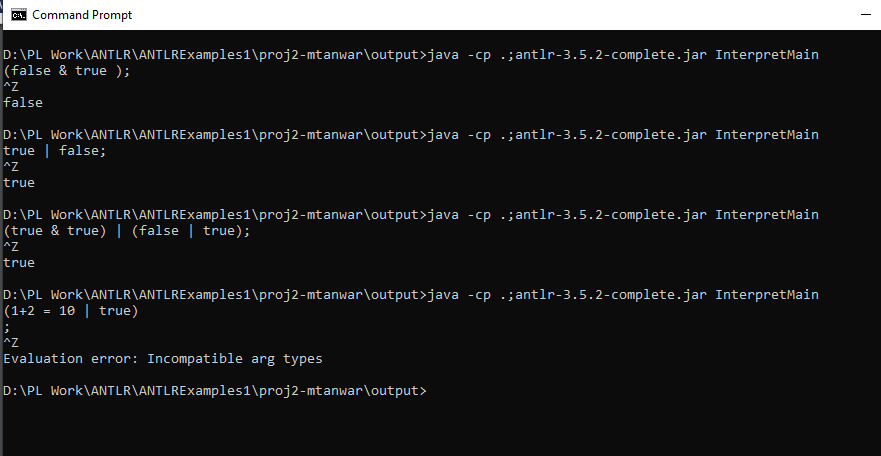
**Case 5: Logical expression**

**(i) (true & true) | (false | true);**

**(ii) true | false;**

**(iii) (false & true );**

**(iv) (1+2 = 10 | true);**



**Case 6: Let val expression**

(i) let val x=10

in

let val y=20

in

x+y

end

end;

(ii) let val x=100

in

let fun sum(b)=

x+b

in sum(200)

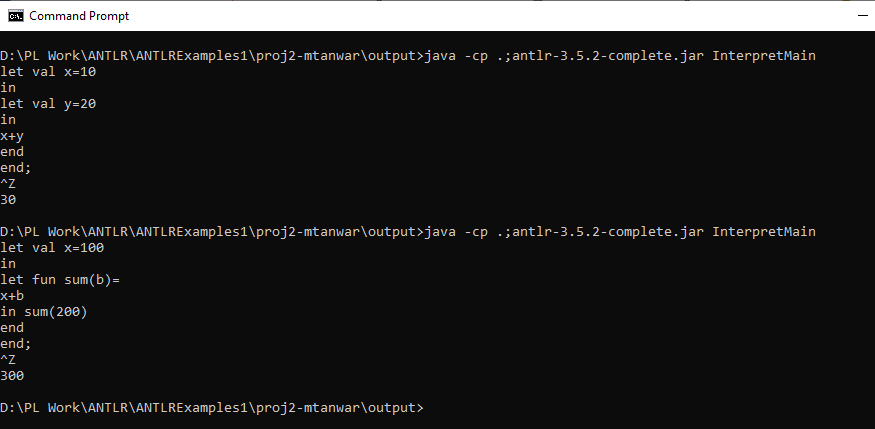
end

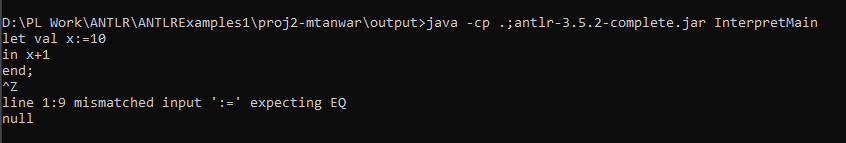
end;

(iii) let val x:=10

in x+1

end;





**Case 7 : Assignment Expression**

(i) let val x=10

in x:=x+20

end;

(ii) let val x=10

in y:=x+20

end;

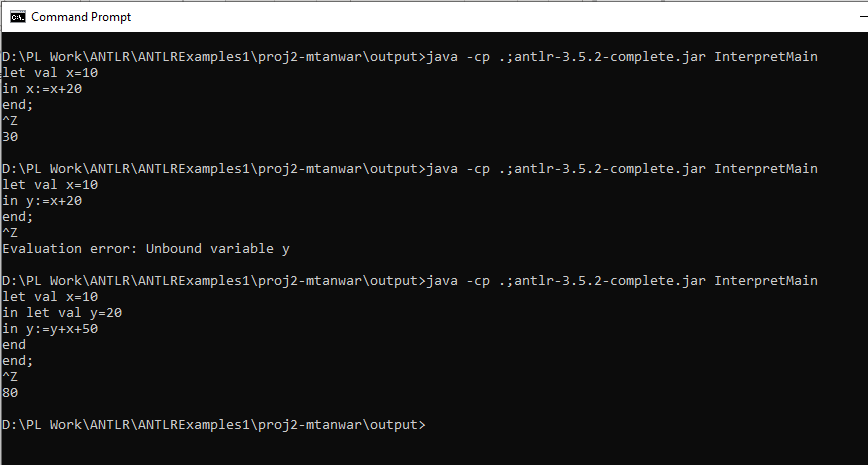
(iii) let val x=10

in let val y=20

in y:=y+x+50

end

end;



**Case 8: Sequence Expression**

(i) let val a=10

in let val b=20

in

{

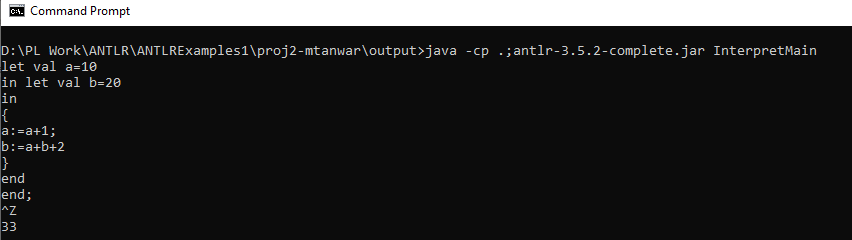
a:=a+1;

b:=a+b+2

}

end

end;



**Case 9: Let fun Expression**

**(i) let fun fact(n)=**

**if (n=0) then 1**

**else n\*fact(n-1)**

**in fact(4)**

**end;**

**(ii) let val x= 100**

**in**

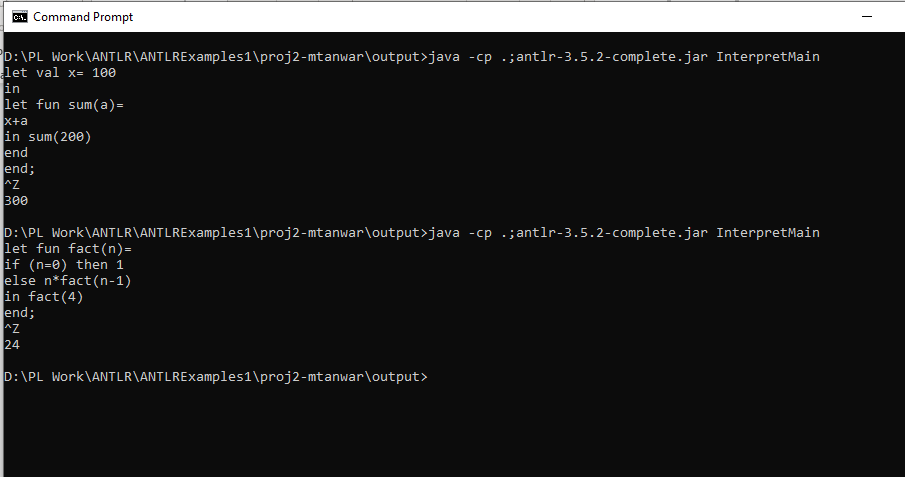
**let fun sum(a)=**

**x+a**

**in sum(200)**

**end**

**end;**



**Case 10: While Expression**

**(i) let val num=10**

**in let val a=0**

**in**

**{**

**while(0 < num) do**

**{**

**a :=a+num;**

**num :=num-1**

**};**

**a**

**}**

**end**

**end;**

**(ii) let val num=10**

**in{**

**while(2 < num) do**

**{**

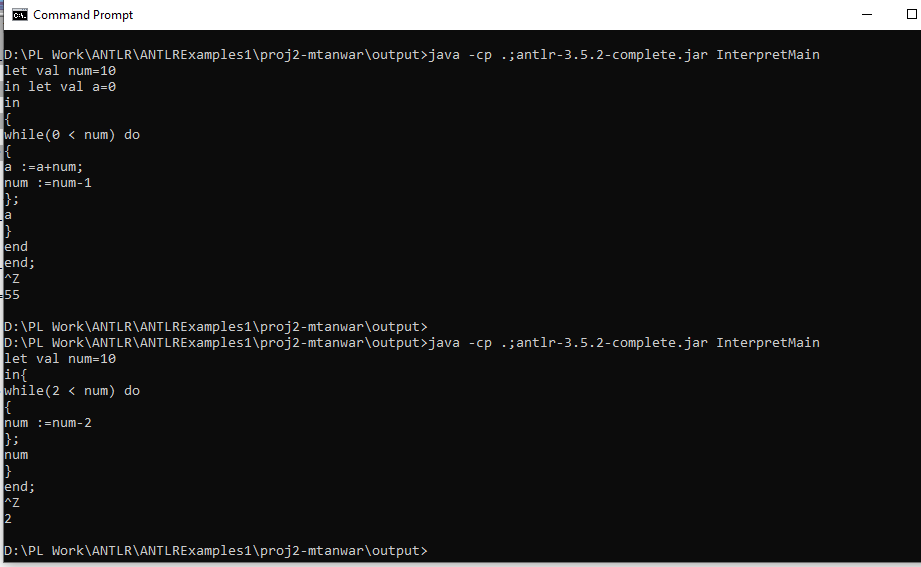
**num :=num-2**

**};**

**num**

**}**

**end;**



**Case 11: Application Expression**

**(i) let fun twice(n)=**

**2\*n**

**in twice(4)**

**end;**

**(ii) let fun fib(n)=**

**if n<2 then n**

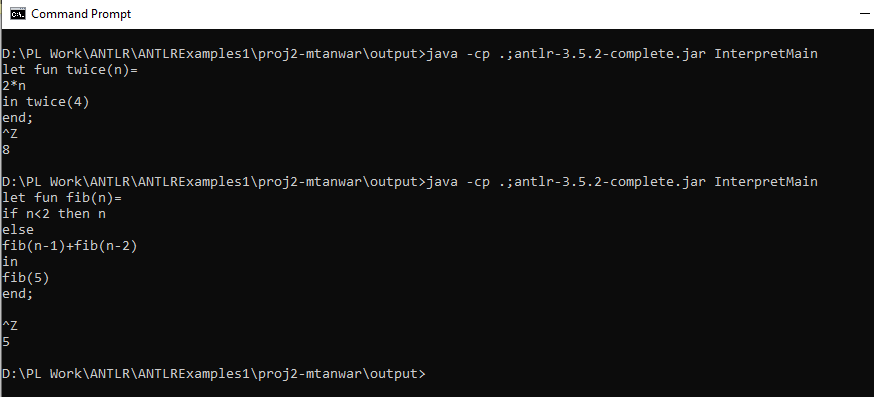
**else**

**fib(n-1)+fib(n-2)**

**in**

**fib(5)**

**end;**



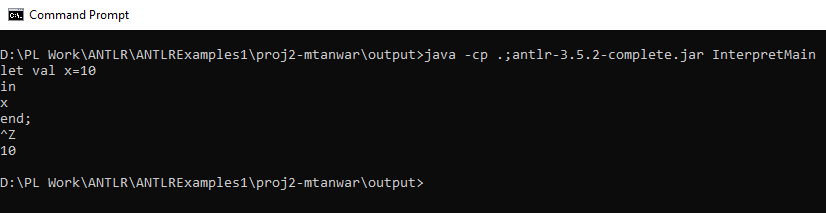
**Case 12: Variable Expression**

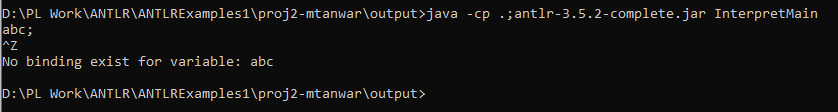
**let val x=10**

**in**

**x**

**end;**





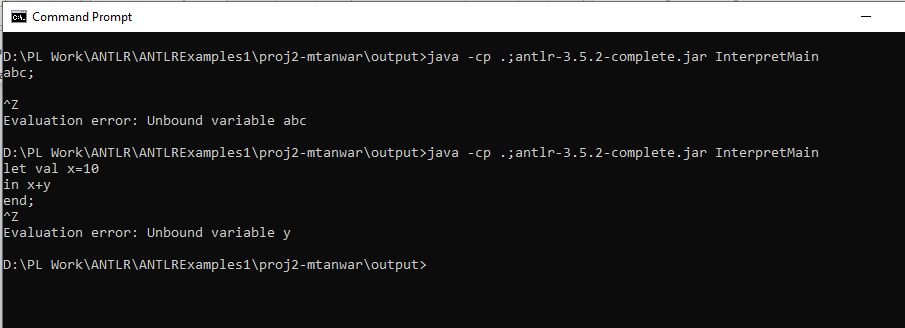
**Case 13: Unbound Error**

**(i) let val x=10**

**in x+y**

**end;**

**(ii) abc;**



**Case 14: Eval Error**

* **2=2 | false | true ;**
* **if( 3 > 2) then 100 else 200;**
* **if(100) then 1 else 2;**

