

**NAME-ANUBHAV ANAND**

**ENROLLMENT NUMBER-2020CSB102**

**SUBJECT-ASSIGNMENT-8 OF DBMS**

**DEPARTMENT-COMPUTER SCIENCE AND  
TECHNOLOGY**

**G-SUITE ID-**

**[2020CSB102.anubhav@students.iiests.ac.in](mailto:2020CSB102.anubhav@students.iiests.ac.in)**

**Q1. Find the maximum and minimum of three given numbers.**

**Ans-**

**DECLARE**

**a NUMBER := 46;**

**b NUMBER := 67;**

**c NUMBER := 21;**

**BEGIN**

**IF a >= b**

**AND a >= c THEN**

**dbms\_output.Put\_line('Maximum number is ' || a);**

**ELSIF b >= a**

**AND b >= c THEN**

**dbms\_output.Put\_line('Maximum number is ' || b);**

**ELSE**

**dbms\_output.Put\_line('Maximum number is ' || c);**

**END IF;**

**IF a <= b**

**AND a <= c THEN**

**dbms\_output.Put\_line('Minimum number is ' || a);**

**ELSIF b <= a**

**AND b <= c THEN**

**dbms\_output.Put\_line('Minimum number is ' || b);**

**ELSE**



**dbms\_output.Put\_line('Minimum number is ' || c);**

**END IF;**

**END;**

**--End program**

## Output plus program-

 Live SQL



SQL Worksheet

```
1 DECLARE
2   a NUMBER := 46;
3   b NUMBER := 67;
4   c NUMBER := 21;
5 BEGIN
6   IF a >= b
7   AND a >= c THEN
8     dbms_output.Put_line('Maximum number is '
9                           ||a);
10  ELSIF b >= a
11  AND b >= c THEN
12    dbms_output.Put_line('Maximum number is '
13                          ||b);
14  ELSE
15    dbms_output.Put_line('Maximum number is '
16                          ||c);
17  END IF;
18  IF a <= b
19  AND a <= c THEN
20    dbms_output.Put_line('Minimum number is '
21                          ||a);
22  ELSIF b <= a
23  AND b <= c THEN
24    dbms_output.Put_line('Minimum number is '
25                          ||b);
26  ELSE
27    dbms_output.Put_line('Minimum number is '
28                          ||c);
29  END IF;
30
31 END;
32 --End program
```

SQL Statement Output

© 2022 Oracle · Live SQL 22.3.1, running Oracle Database 19c Enterprise Edition - 19.14.0.0.0 · Databases  
Built with ❤️ using Oracle APEX · Privacy · Terms of Use

## Output-

 Live SQL

SQL Worksheet

```
Statement processed.
Maximum number is 67
Minimum number is 21
```

**Q2. Find the factorial of a given number.**

**Ans-Code-**

**DECLARE**

**factorial number :=1;**

**n number := 7;**

**BEGIN**

**while n > 0 loop**

**factorial:=n\*factorial;**

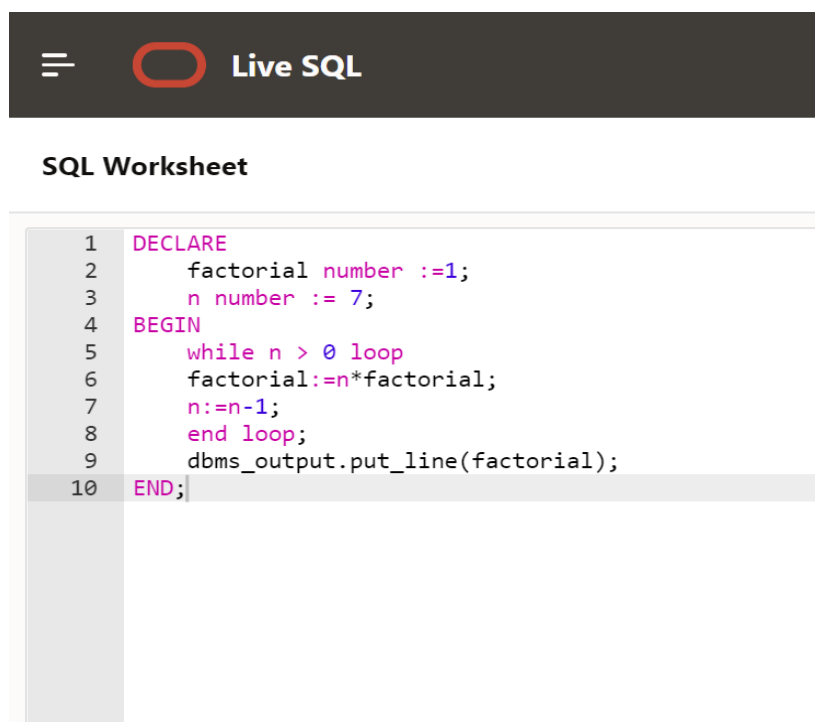
**n:=n-1;**

**end loop;**

**dbms\_output.put\_line(factorial);**

**END;**


**Code-**



The screenshot shows a web-based SQL editor interface. At the top, there is a dark header bar with a hamburger menu icon, an orange oval logo, and the text "Live SQL". Below the header, the page is titled "SQL Worksheet". The main area contains a code editor with a line number column on the left (1 to 10) and the SQL code on the right. The code is as follows:

```
1 DECLARE
2     factorial number :=1;
3     n number := 7;
4 BEGIN
5     while n > 0 loop
6         factorial:=n*factorial;
7         n:=n-1;
8     end loop;
9     dbms_output.put_line(factorial);
10 END;
```

## Output-

 Live SQL

### SQL Worksheet

```
1 DECLARE
2     factorial number :=1;
3     n number := 7;
4 BEGIN
5     while n > 0 loop
6         factorial:=n*factorial;
7         n:=n-1;
8     end loop;
9     dbms_output.put_line(factorial);
10 END;
```

Statement processed.  
5040

**Q3. Reverse a given string.**

**Ans-Code-**

**DECLARE**

**str VARCHAR2(30);**

**final VARCHAR2(40);**

**len NUMBER;**

**BEGIN**

**str:='ANUBHAV';**

**len:=length(str);**

**for i in reverse 1..len**

**loop**

**final:=final || substr(str,i,1);**



**end loop;**

**dbms\_output.Put\_line('Reverse:' || final);**

**END;**

**--End Program**



## Code+Output-

 Live SQL

SQL Worksheet

```
1 DECLARE
2     str VARCHAR2(30);
3     final VARCHAR2(40);
4     len NUMBER;
5 BEGIN
6     str:='ANUBHAV';
7     len:=length(str);
8     for i in reverse 1..len
9     loop
10        final:=final||substr(str,i,1);
11    end loop;
12    dbms_output.Put_line('Reverse:' || final);
13 END;
14 --End Program
```

## Output-

 Live SQL

SQL Worksheet

```
1 DECLARE
2     str VARCHAR2(30);
3     final VARCHAR2(40);
4     len NUMBER;
5 BEGIN
6     str:='ANUBHAV';
7     len:=length(str);
8     for i in reverse 1..len
9     loop
10        final:=final||substr(str,i,1);
11    end loop;
12    dbms_output.Put_line('Reverse:' || final);
13 END;
14 --End Program
```

Statement processed.  
Reverse:VAHBUNA

**Q4. Consider a banking database. Accept an account number from the user, check if the**

**balance in the account is less than the minimum balance to be kept in bank account, only**

**then deduct Rs. 100/= from the balance. The process is fired on the ACCT\_MSTR table.**

**ACCT\_MSTR (acct\_no, type, curbal, ststus)**

**'type' can hold the values:**

**'CA' for Current Account,**

**'SB' for Savings Bank Account.**

**'status' can hold the values:**

**'A' for Active account,**

**'S' for suspended account,**

**'T' for Terminated account.**



**Ans-Code-**

```
create table ACCOUNT_MASTER(acct_no number(5) primary key,  
                             type varchar2(10),  
                             curbal number(10),  
                             status varchar(10));
```

```
insert into ACCOUNT_MASTER values(1, 'CA', 1000,'A');  
insert into ACCOUNT_MASTER values(2, 'SB', 100,'S');  
insert into ACCOUNT_MASTER values(3, 'CA', 1100,'T');  
insert into ACCOUNT_MASTER values(4, 'CA', 700,'A');  
insert into ACCOUNT_MASTER values(5, 'SB', 1700,'A') ;
```

**DECLARE**

```
    xacct_no number(5);  
    xmin_bal number(5):=1000;  
    xbalance number(5);
```

**BEGIN**

```
xacct_no:=4;  
select curbal into xbalance  
from ACCOUNT_MASTER  
where acct_no=xacct_no;
```

```
IF(xbalance < xmin_bal) THEN  
update ACCOUNT_MASTER  
set curbal=curbal-100
```

```
where acct_no=xacct_no;  
xbalance:=xbalance-100;  
dbms_output.put_line('Rs 100 is deducted  
and current balance is ' || xbalance);
```

```
ELSE  
dbms_output.put_line('Current balance is ' || xbalance);  
END IF;  
END;
```

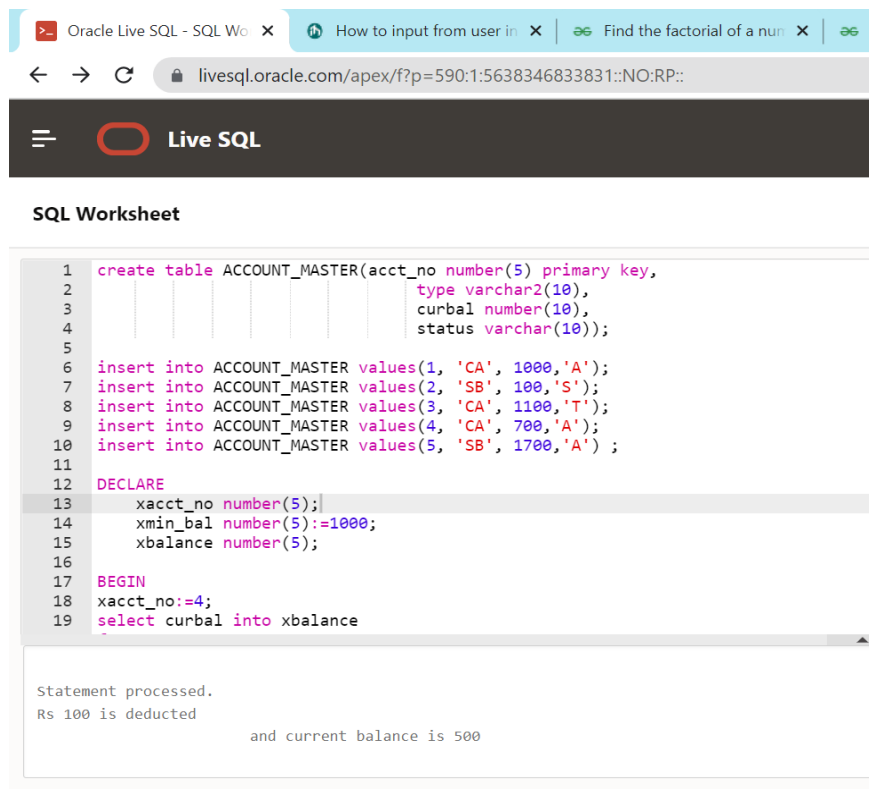
## Code+output-



### SQL Worksheet

```
1  create table ACCOUNT_MASTER(acct_no number(5) primary key,
2                                type varchar2(10),
3                                curbal number(10),
4                                status varchar(10));
5
6  insert into ACCOUNT_MASTER values(1, 'CA', 1000, 'A');
7  insert into ACCOUNT_MASTER values(2, 'SB', 100, 'S');
8  insert into ACCOUNT_MASTER values(3, 'CA', 1100, 'T');
9  insert into ACCOUNT_MASTER values(4, 'CA', 700, 'A');
10 insert into ACCOUNT_MASTER values(5, 'SB', 1700, 'A') ;
11
12 DECLARE
13     xacct_no number(5);
14     xmin_bal number(5):=1000;
15     xbalance number(5);
16
17 BEGIN
18     xacct_no:=4;
19     select curbal into xbalance
20     from ACCOUNT_MASTER
21     where acct_no=xacct_no;
22
23     IF(xbalance < xmin_bal) THEN
24         update ACCOUNT_MASTER
25         set curbal=curbal-100
26         where acct_no=xacct_no;
27         xbalance:=xbalance-100;
28         dbms_output.put_line('Rs 100 is deducted
29                                and current balance is '||xbalance);
30
31     ELSE
32         dbms_output.put_line('Current balance is '||xbalance);
33     END IF;
34 END;
```

## Output-



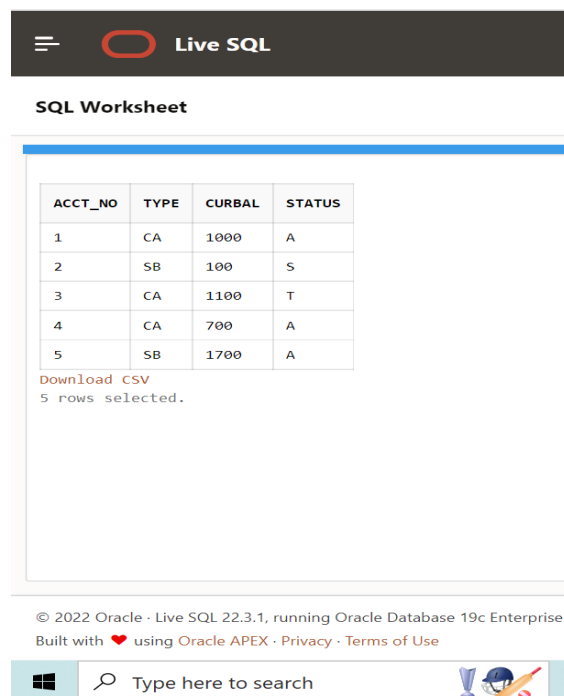
The screenshot shows the Oracle Live SQL interface. The browser tabs include "Oracle Live SQL - SQL Wo...", "How to input from user in...", and "Find the factorial of a num...". The address bar shows the URL "livesql.oracle.com/apex/?p=590:1:5638346833831::NO:RP:". The page header says "Live SQL". The "SQL Worksheet" section contains the following SQL code:

```
1 create table ACCOUNT_MASTER(acct_no number(5) primary key,
2                               type varchar2(10),
3                               curbal number(10),
4                               status varchar(10));
5
6 insert into ACCOUNT_MASTER values(1, 'CA', 1000,'A');
7 insert into ACCOUNT_MASTER values(2, 'SB', 100,'S');
8 insert into ACCOUNT_MASTER values(3, 'CA', 1100,'T');
9 insert into ACCOUNT_MASTER values(4, 'CA', 700,'A');
10 insert into ACCOUNT_MASTER values(5, 'SB', 1700,'A') ;
11
12 DECLARE
13     xacct_no number(5);
14     xmin_bal number(5):=1000;
15     xbalance number(5);
16
17 BEGIN
18     xacct_no:=4;
19     select curbal into xbalance
```

Below the code, the output is displayed:

Statement processed.  
Rs 100 is deducted  
and current balance is 500

## Table pic after insertion-



The screenshot shows the Oracle Live SQL interface. The browser tabs include "Oracle Live SQL - SQL Wo...", "How to input from user in...", and "Find the factorial of a num...". The address bar shows the URL "livesql.oracle.com/apex/?p=590:1:5638346833831::NO:RP:". The page header says "Live SQL". The "SQL Worksheet" section contains the following SQL code:

```
1 create table ACCOUNT_MASTER(acct_no number(5) primary key,
2                               type varchar2(10),
3                               curbal number(10),
4                               status varchar(10));
5
6 insert into ACCOUNT_MASTER values(1, 'CA', 1000,'A');
7 insert into ACCOUNT_MASTER values(2, 'SB', 100,'S');
8 insert into ACCOUNT_MASTER values(3, 'CA', 1100,'T');
9 insert into ACCOUNT_MASTER values(4, 'CA', 700,'A');
10 insert into ACCOUNT_MASTER values(5, 'SB', 1700,'A') ;
11
12 DECLARE
13     xacct_no number(5);
14     xmin_bal number(5):=1000;
15     xbalance number(5);
16
17 BEGIN
18     xacct_no:=4;
19     select curbal into xbalance
```

Below the code, the output is displayed:

Statement processed.  
Rs 100 is deducted  
and current balance is 500



The table data is shown in a table with 4 columns: ACCT\_NO, TYPE, CURBAL, and STATUS. The data is as follows:

ACCT_NO	TYPE	CURBAL	STATUS
1	CA	1000	A
2	SB	100	S
3	CA	1100	T
4	CA	700	A
5	SB	1700	A

Below the table, the text "Download CSV" and "5 rows selected." is displayed.

© 2022 Oracle - Live SQL 22.3.1, running Oracle Database 19c Enterprise Edition  
Built with ❤️ using Oracle APEX - Privacy - Terms of Use

## Table pic after deduction-

 **Live SQL**

### SQL Worksheet

ACCT_NO	TYPE	CURBAL	STATUS
1	CA	1000	A
2	SB	100	S
3	CA	1100	T
4	CA	600	A
5	SB	1700	A

[Download CSV](#)

5 rows selected.

**Q5. Calculate the area of a circle for a value of radius varying from 3 to 7.**

**Store the radius**

**and the corresponding values of calculated area in an empty table named 'Areas',**

**containing two columns 'Radius' and 'Area'.**

**Ans-Code-**

**create table areas ( r number(2), area number (14,2));**

**DECLARE**

**r number(5);**

**area number(14,2);**

**pi constant number (4,2):=3.14;**

**BEGIN**

**r:=3;**

**while r<=7**

**loop**

**area:=pi\*power(r,2);**

**insert into areas values(r,area );**

**r:=r+1;**

**end loop;**

**END;**

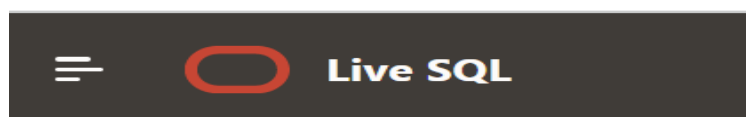
**select \* from areas;**

Code-



## SQL Worksheet

```
1 create table areas ( r number(2), area number (14,2));
2 DECLARE
3     r number(5);
4     area number(14,2);
5     pi constant number (4,2):=3.14;
6 BEGIN
7     r:=3;
8     while r<=7
9     loop
10    area:=pi*power(r,2);
11    insert into areas values(r,area );
12    r:=r+1;
13    end loop;
14 END;
15 select * from areas;
```



## SQL Worksheet

R	AREA
3	28.26
4	50.24
5	78.5
6	113.04
7	153.86

