

## ASSIGNMENT ON SQL:

### 1) Create A Table With Four Columns Of Your Choice, Making That They Have Null Constraints

```
create database assignment1;
```

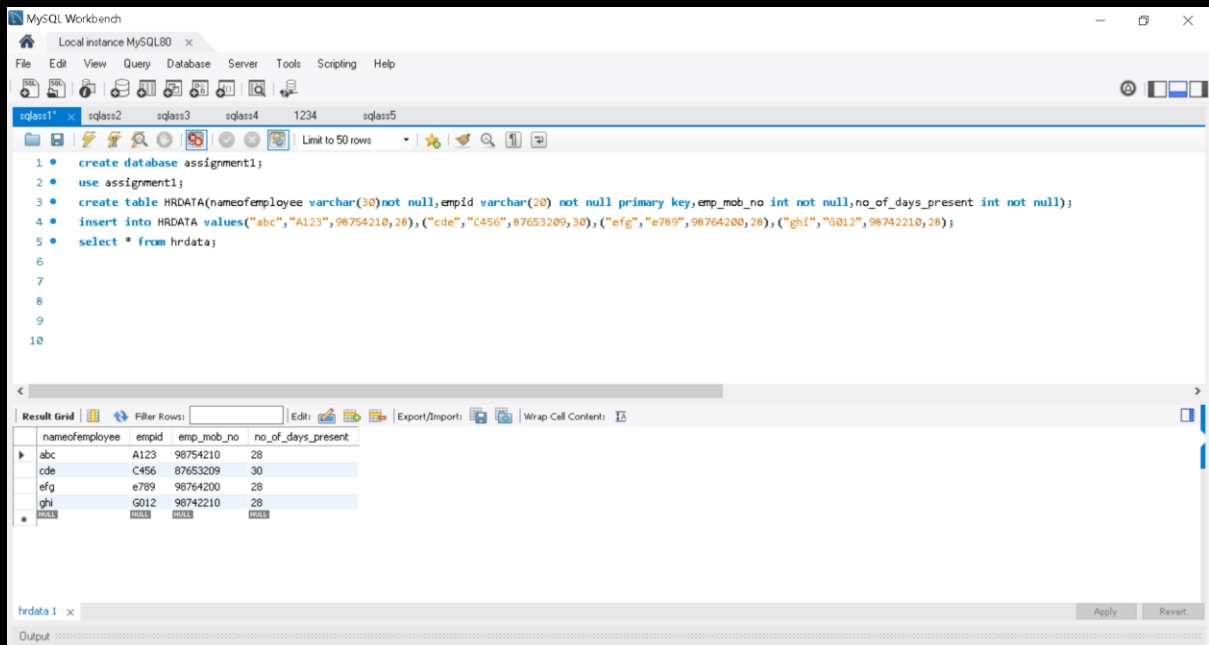
```
use assignment1;
```

```
create table HRDATA(nameofemployee varchar(30)not null,empid varchar(20) not null primary  
key,emp_mob_no int not null,no_of_days_present int not null);
```

```
insert into HRDATA
```

```
values("abc","A123",98754210,28),("cde","C456",87653209,30),("efg","e789",98764200,28),("ghi","  
G012",98742210,28);
```

```
select * from hrdata;
```



The screenshot shows the MySQL Workbench interface. The top toolbar includes icons for file operations, query execution, and database management. The main window displays a SQL script with the following queries:

```
1 • create database assignment1;  
2 • use assignment1;  
3 • create table HRDATA(nameofemployee varchar(30)not null,empid varchar(20) not null primary key,emp_mob_no int not null,no_of_days_present int not null);  
4 • insert into HRDATA values("abc","A123",98754210,28),("cde","C456",87653209,30),("efg","e789",98764200,28),("ghi","G012",98742210,28);  
5 • select * from hrdata;
```

Below the script, the 'Result Grid' shows the output of the SELECT query. It contains four rows of data:

nameofemployee	empid	emp_mob_no	no_of_days_present
abc	A123	98754210	28
cde	C456	87653209	30
efg	e789	98764200	28
ghi	G012	98742210	28

The bottom status bar shows 'hrdata 1' and 'Output'.

2) Create A Sales Table Having Columns ID, Product name, Price Per Unit And Quantity, and Then Create A View Which Will Show The Total Cost Per Each Product And Product Name

```
create database assignment2;
```

```
use assignment2;
```

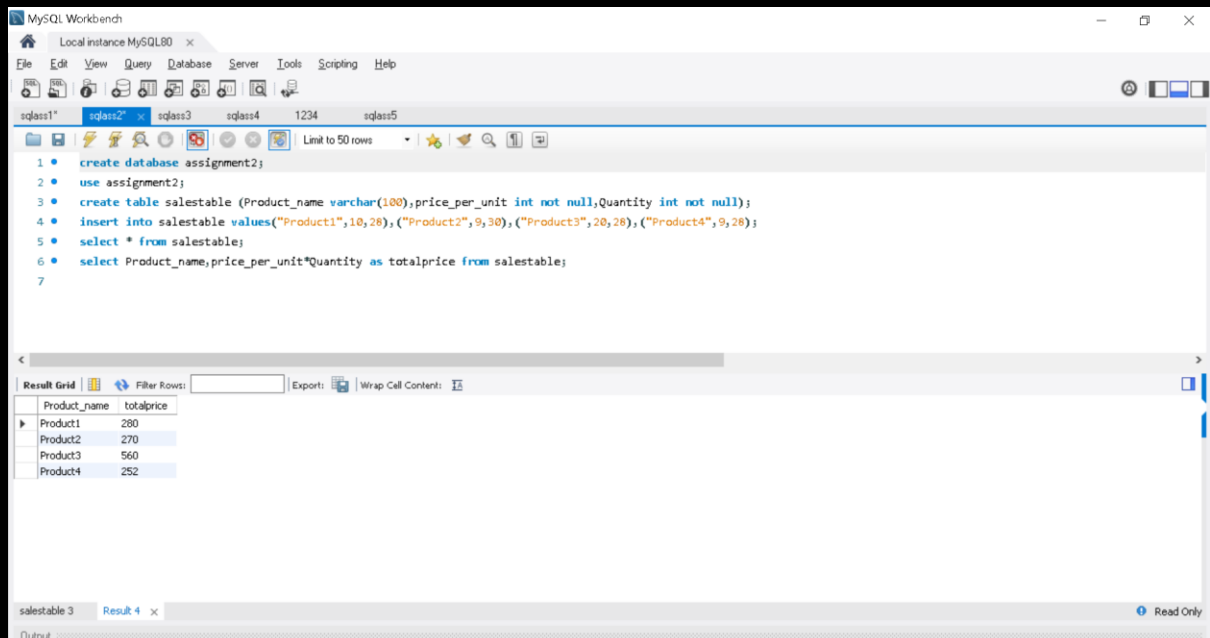
```
create table salestable (Product_name varchar(100),price_per_unit int not null,Quantity int not null);
```

```
insert into salestable
```

```
values("Product1",10,28),("Product2",9,30),("Product3",20,28),("Product4",9,28);
```

```
select * from salestable;
```

```
select Product_name,price_per_unit*Quantity as totalprice from salestable;
```



The screenshot shows the MySQL Workbench interface. The top toolbar includes icons for file operations, editing, and database management. The main editor window contains the following SQL script:

```
1 • create database assignment2;
2 • use assignment2;
3 • create table salestable (Product_name varchar(100),price_per_unit int not null,Quantity int not null);
4 • insert into salestable values("Product1",10,28),("Product2",9,30),("Product3",20,28),("Product4",9,28);
5 • select * from salestable;
6 • select Product_name,price_per_unit*Quantity as totalprice from salestable;
7
```

Below the editor, the 'Result Grid' tab is active, displaying the output of the last query. The grid shows four rows of data:

Product_name	totalprice
Product1	280
Product2	270
Product3	560
Product4	252

The bottom status bar indicates the current view is 'Result 4' and is in 'Read Only' mode.

3)

3)

For this challenge, use the same table again:

id	Name	Age
1	Bob	21
2	Sam	19
3	Jill	18
4	Jim	21
5	Sally	19
6	Jess	20
7	Will	21

Your task for this challenge is to return a sum of all the ages in the table. Your query should

```
create database assignment3;
```

```
use assignment3;
```

```
create table table1(id int,Name_ Varchar(100),age int);
```

```
insert into table1
```

```
values(1,"Bob",21),(2,"Sam",19),(3,"Jill",18),(4,"Jim",21),(5,"Sally",19),(6,"Jess",20),(7,"Will",21);
```

```
select * from table1;
```

```
select sum(age) from table1;
```

The screenshot shows the MySQL Workbench interface. The SQL editor contains the following queries:

```
1 • create database assignment3;
2 • use assignment3;
3 • create table table1(id int,Name_ Varchar(100),age int);
4 • insert into table1 values(1,"Bob",21),(2,"Sam",19),(3,"Jill",18),(4,"Jim",21),(5,"Sally",19),(6,"Jess",20),(7,"Will",21);
5 • select * from table1;
6 • select sum(age) from table1;
7
8
```

The Results tab shows the output of the last query:

sum(age)
417

The bottom status bar indicates the current table is 'table1' and the result is 'Result 2'.

4)

4)

This is the table structure that you'll use for this SQL challenge:

id	Name	Age
1	Bob	21
2	Sam	19
3	Jill	18
4	Jim	21
5	Sally	19
6	Jess	20
7	Will	21

The challenge is to write a query that'll group all the people by their age, along with a count of the people that are the same age. Here's

```
create database assignmentgroup;
```

```
use assignmentgroup;
```

```
create table table1(id int,Name_ Varchar(100),age int);
```

```
insert into table1
```

```
values(1,"Bob",21),(2,"Sam",19),(3,"Jill",18),(4,"Jim",21),(5,"Sally",19),(6,"Jess",20),(7,"Will",21);
```

```
select * from table1;
```

```
select age,count(age) from table1 group by age ;
```

The screenshot shows the MySQL Workbench interface. The SQL editor contains the following queries:

```
1 • create database assignmentgroup;
2 • use assignmentgroup;
3 • create table table1(id int,Name_ Varchar(100),age int);
4 • insert into table1 values(1,"Bob",21),(2,"Sam",19),(3,"Jill",18),(4,"Jim",21),(5,"Sally",19),(6,"Jess",20),(7,"Will",21);
5 • select * from table1;
6 • select age,count(age) from table1 group by age ;
7 •
```

The Results window shows the output of the last query:

age	count(age)
21	6
19	4
18	2
20	2

5)

5)

In this challenge, your database table is:

Division id	Year	Revenue
1	2018	60
1	2021	40
1	2020	70
2	2021	-10
3	2018	20
3	2016	40
4	2021	50

Your task is to write a query for this table that'll return just the division ids for all the divisions that had positive revenue in 2021.

```
create database assignment5;
```

```
use assignment5;
```

```
create table table2(Division_id int,Year_ int,revenue int);
```

```
insert into table2 values(1,2018,60),(1,2021,40),(1,2020,70),(2,2021,-10),(3,2018,20),(3,2016,40),(4,2021,50);
```

```
select * from table2 where revenue>=1 and year_=2021;
```

The screenshot shows the MySQL Workbench interface. The SQL editor contains the following queries:

```
1 • create database assignment5;
2 • use assignment5;
3 • create table table2(Division_id int,Year_ int,revenue int);
4 • insert into table2 values(1,2018,60),(1,2021,40),(1,2020,70),(2,2021,-10),(3,2018,20),(3,2016,40),(4,2021,50);
5 • select * from table2 where revenue>=1 and year_=2021;
6
```

The Result Grid shows the output of the final query:

Division_id	Year_	revenue
1	2021	40
4	2021	50