LAB ASSIGNMENT 08

1. Create the following schema and insert some tuples in these tables shown below.
2. **Author (ID, Name, Birth\_Year, Death\_Year (NULL in case of Author is alive)**

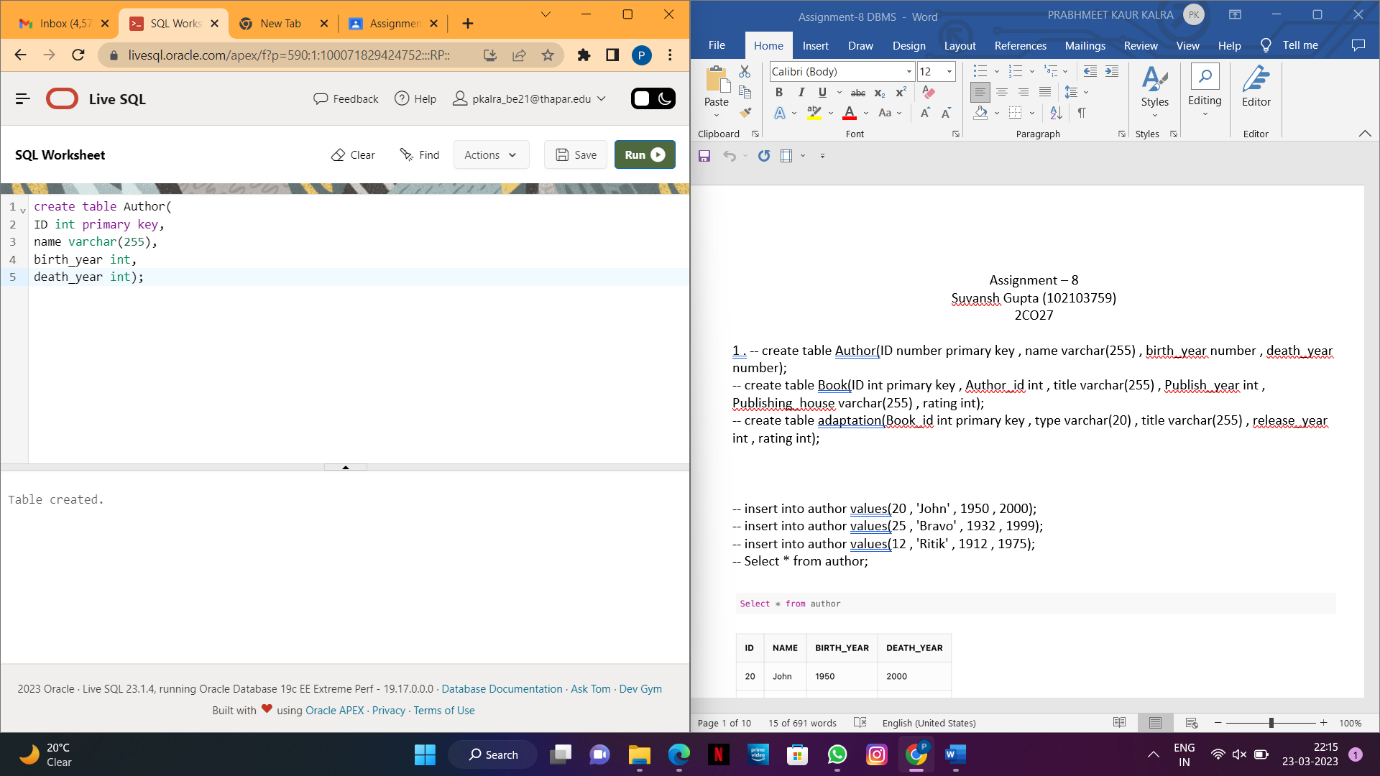
create table Author(

-- ID int primary key,

-- name varchar(255),

-- birth\_year int,

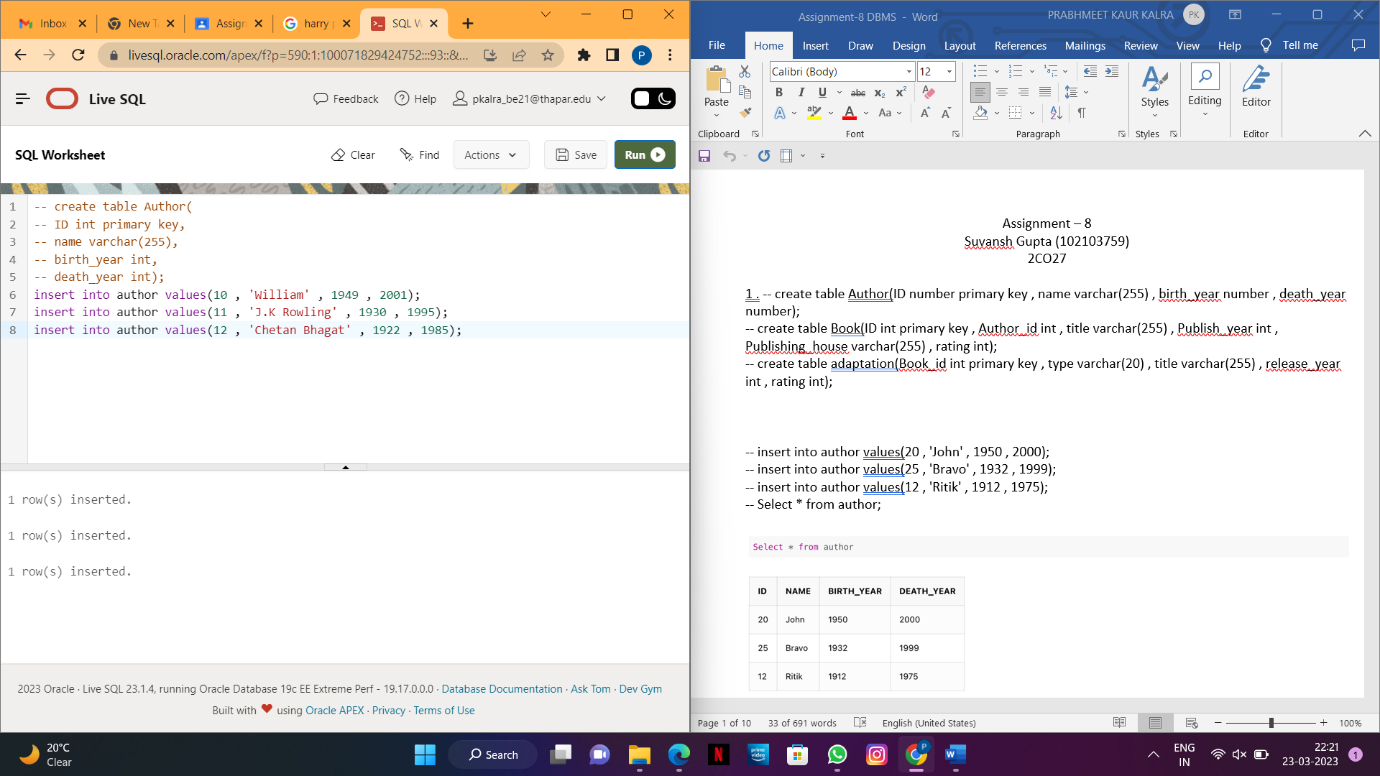
-- death\_year int);



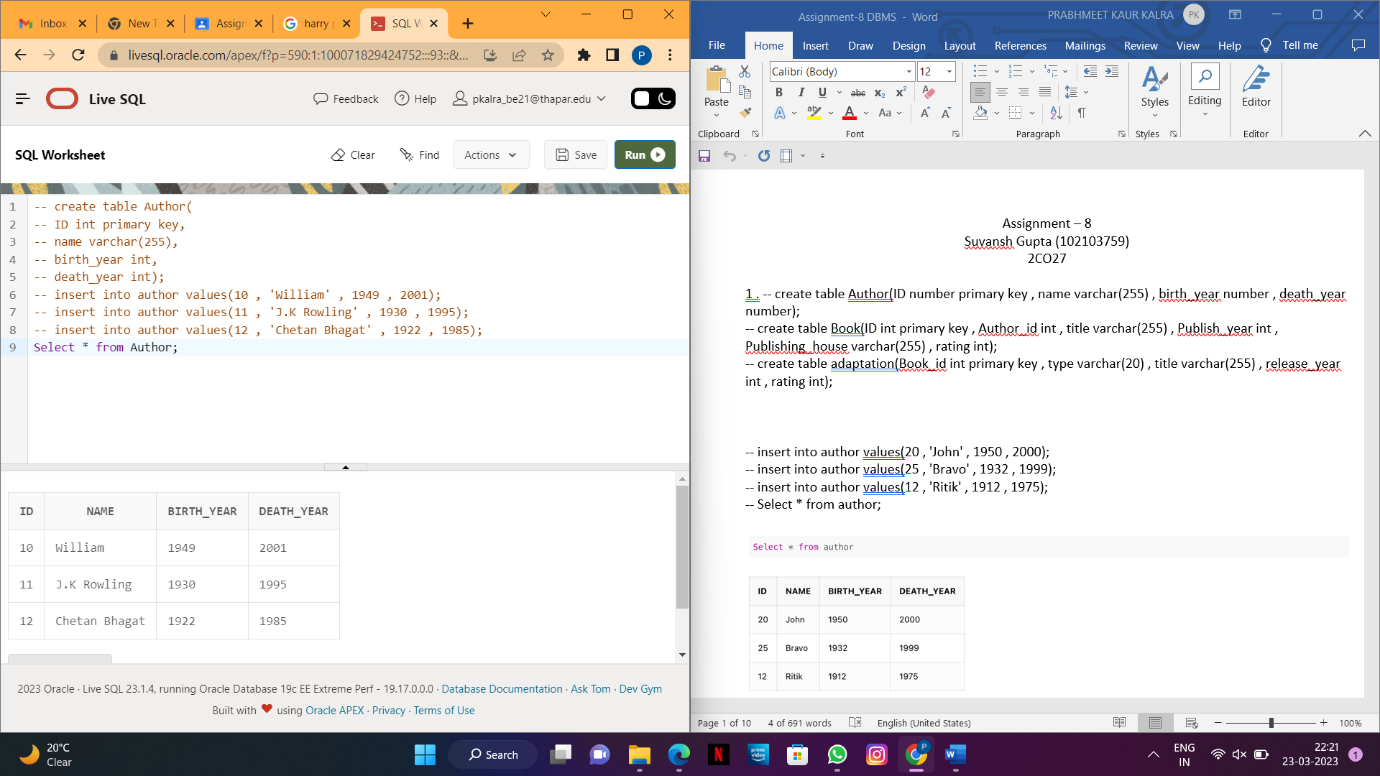
-- insert into author values(10 , 'William' , 1949 , 2001);

-- insert into author values(11 , 'J.K Rowling' , 1930 , 1995);

-- insert into author values(12 , 'Chetan Bhagat' , 1922 , 1985);



-- Select \* from Author;



1. **Book (ID, Author\_ID, Title, Publish\_Year, Publishing\_House, Rating))**

create table Book(

-- ID int primary key,

-- Author\_id int,

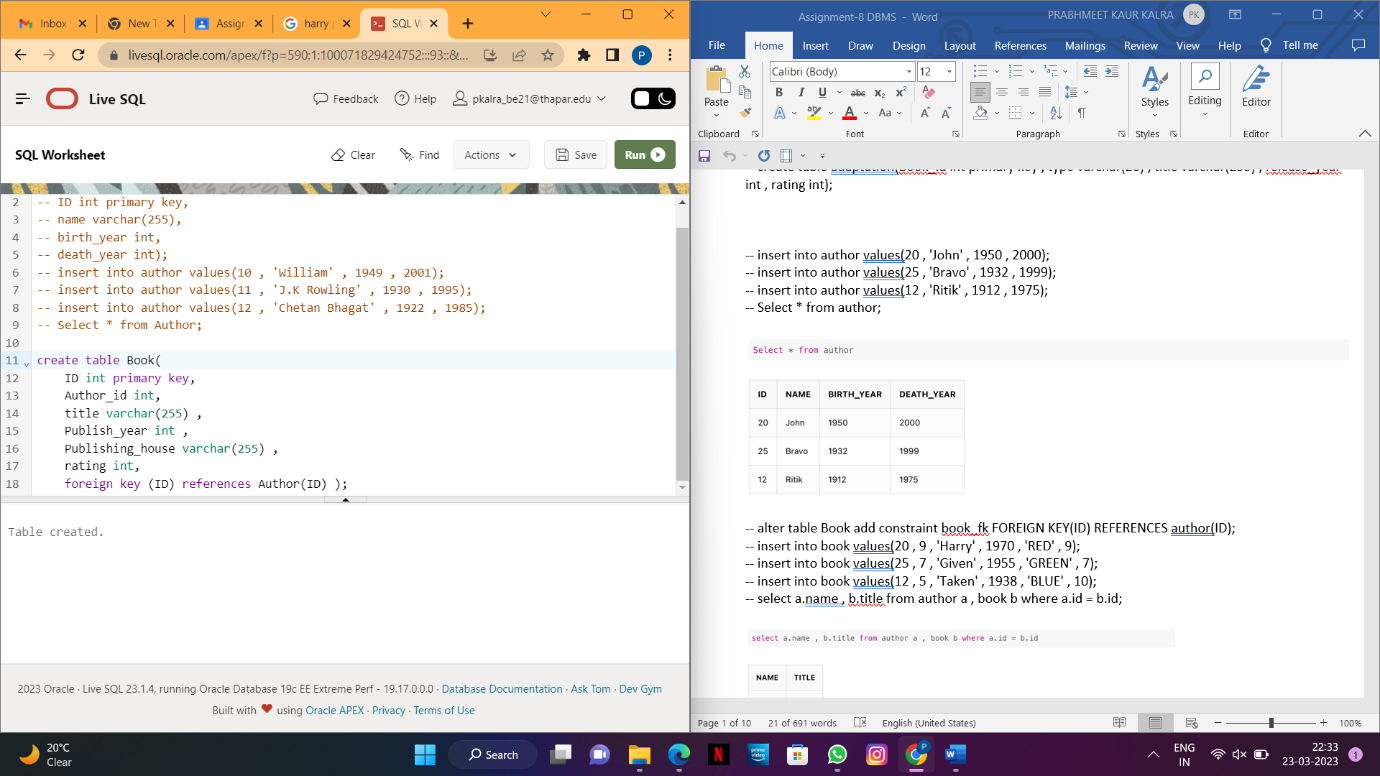
-- title varchar(255) ,

-- Publish\_year int ,

-- Publishing\_house varchar(255) ,

-- rating int,

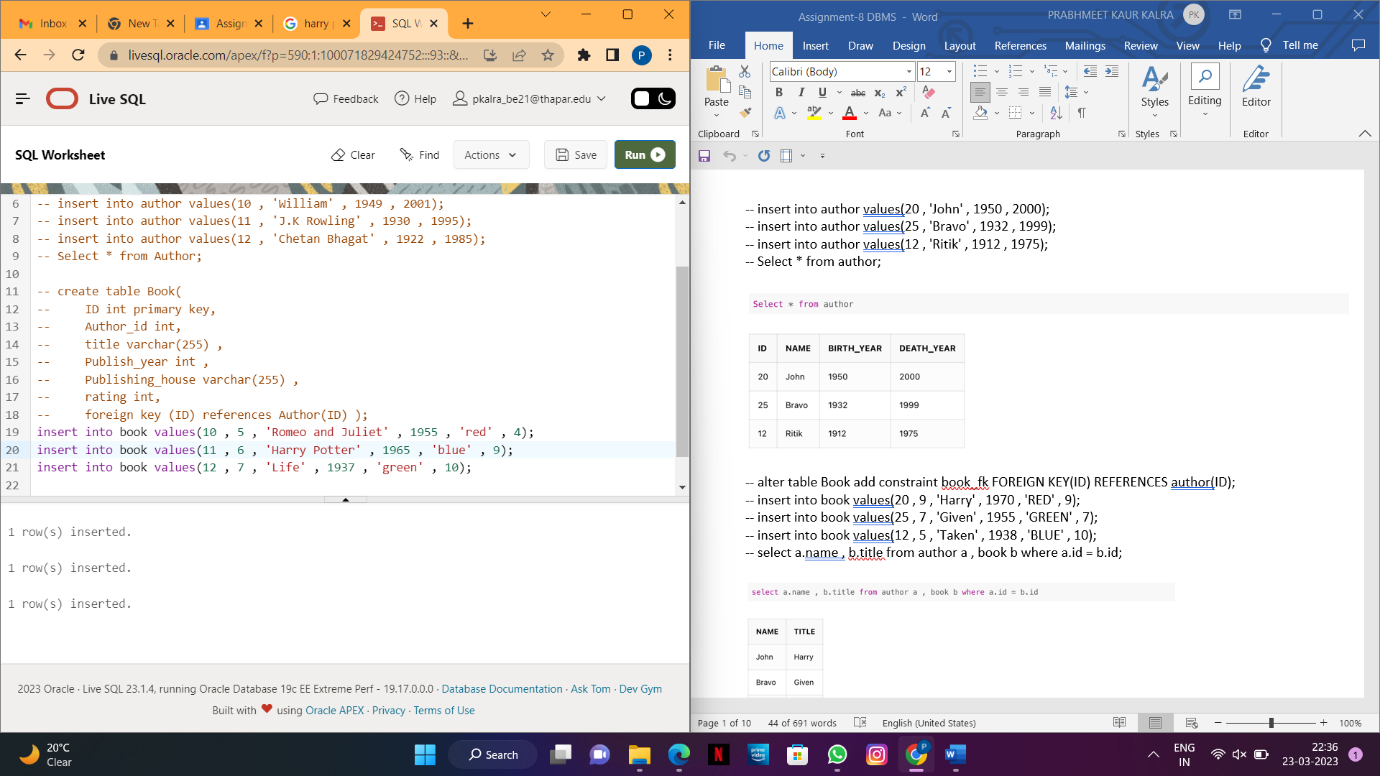
-- foreign key (ID) references Author(ID) );



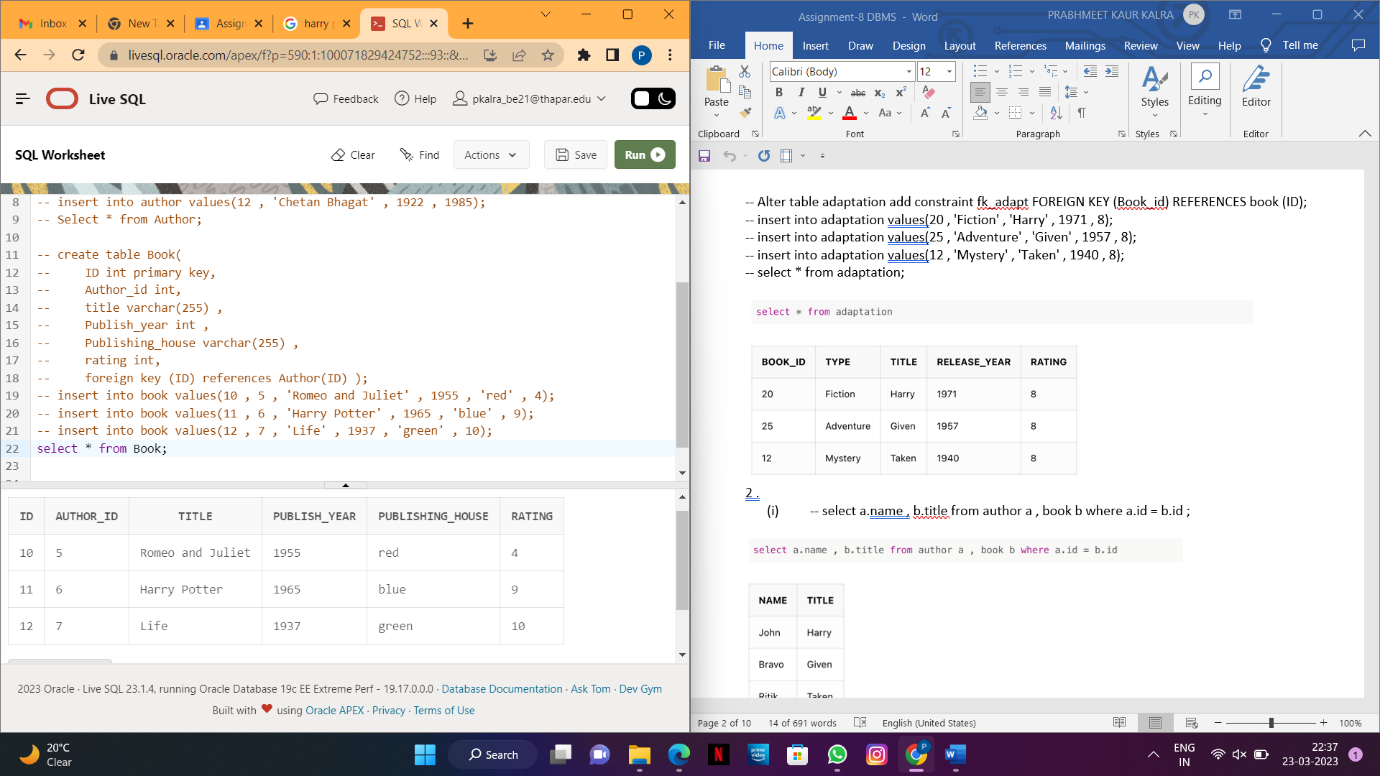
-- insert into book values(10 , 5 , 'Romeo and Juliet' , 1955 , 'red' , 4);

-- insert into book values(11 , 6 , 'Harry Potter' , 1965 , 'blue' , 9);

-- insert into book values(12 , 7 , 'Life' , 1937 , 'green' , 10);



-- select \* from Book;



1. **Adaptation (Book\_ID, Type, Title, Release\_Year, Rating)**

-- create table Adaptation(

-- Book\_id int primary key ,

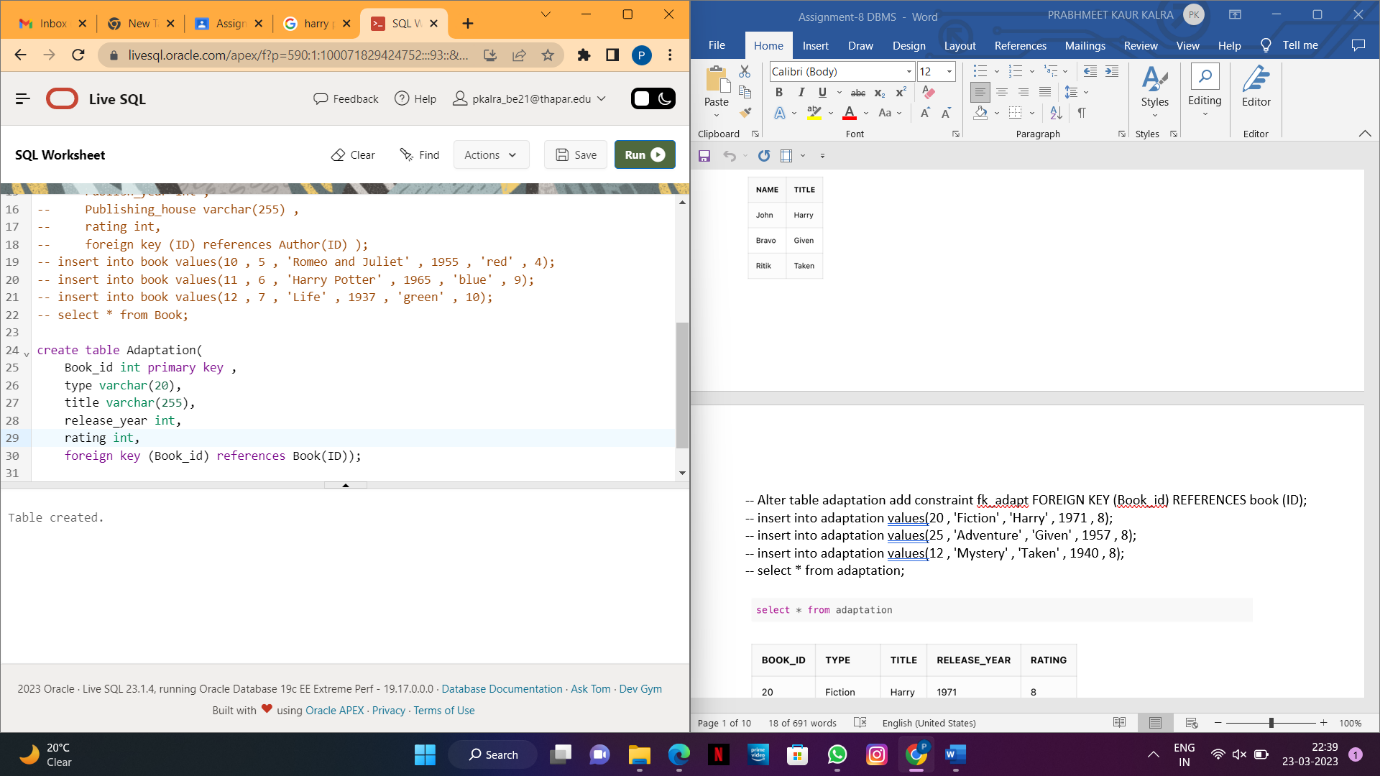
-- type varchar(20),

-- title varchar(255),

-- release\_year int,

-- rating int,

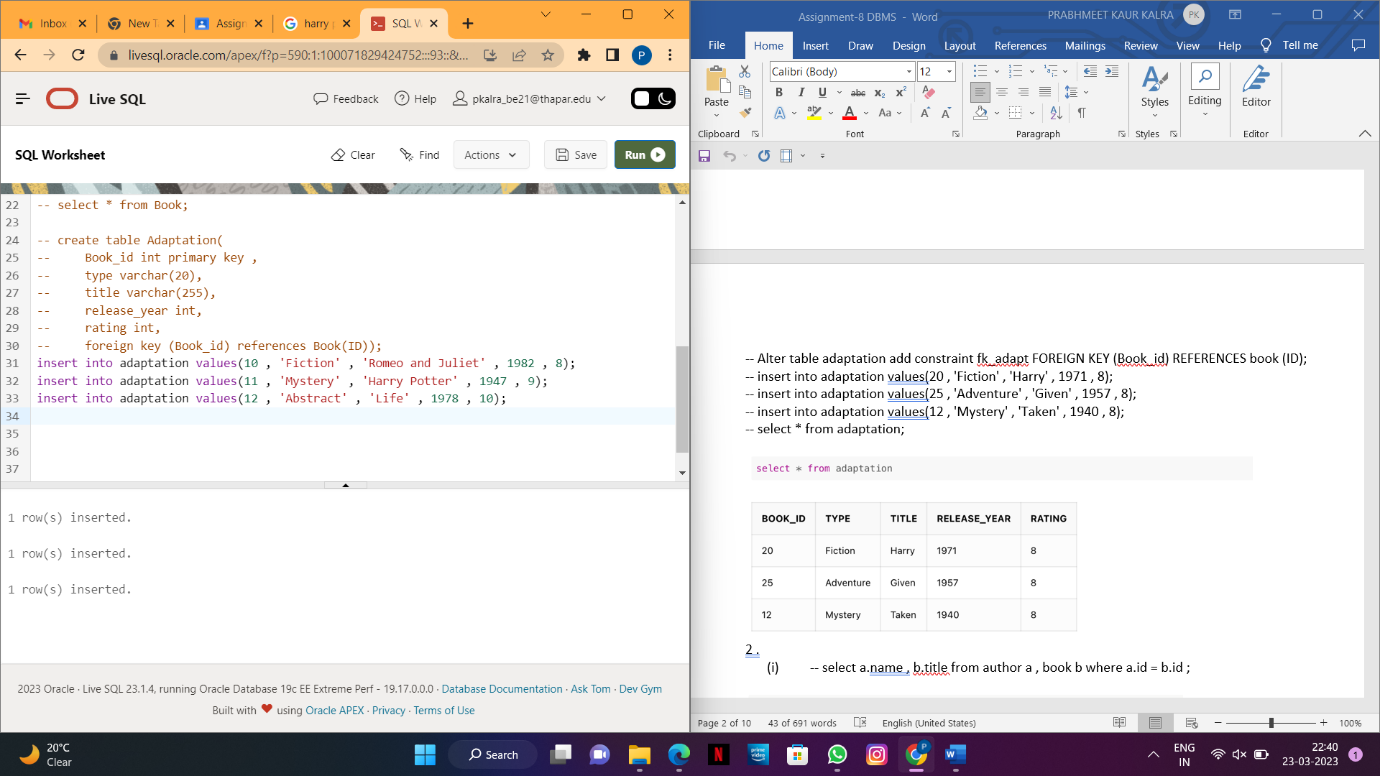
-- foreign key (Book\_id) references Book(ID));



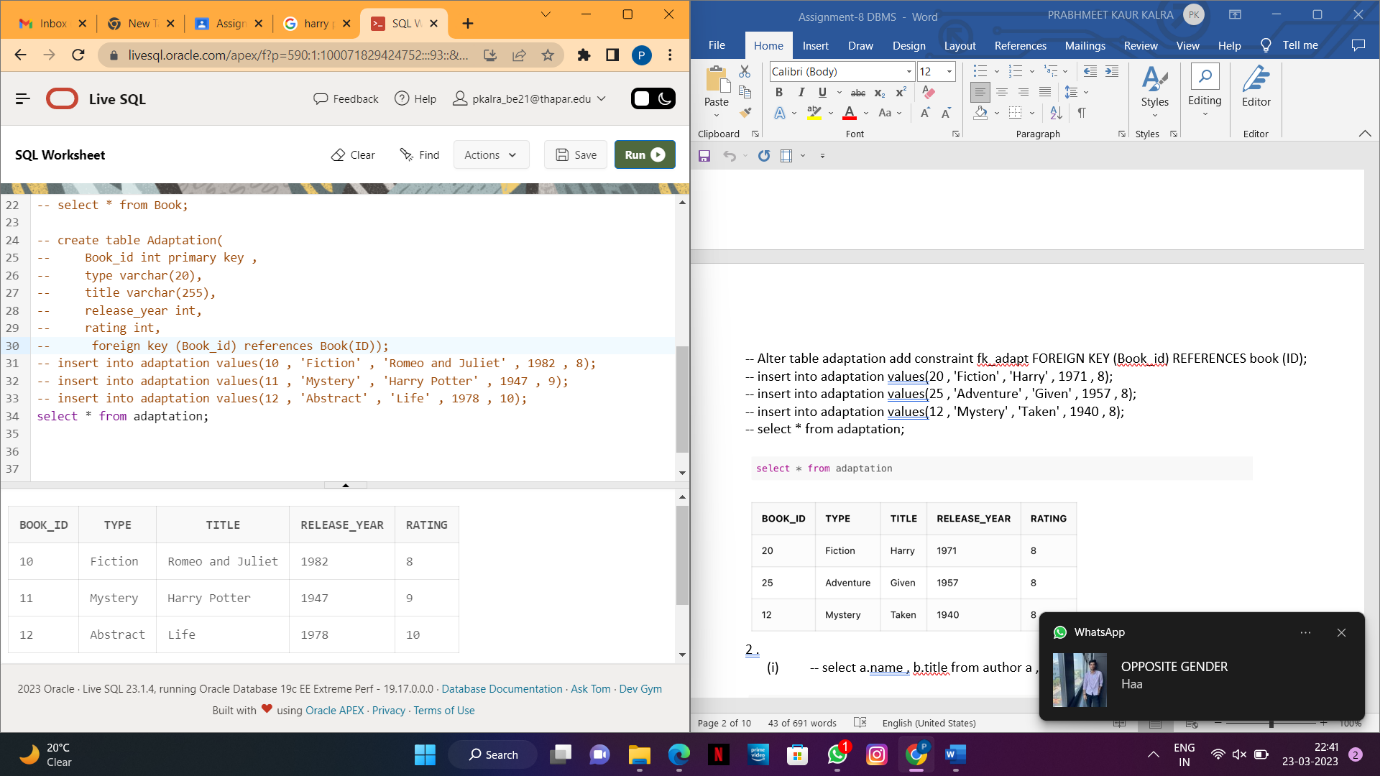
-- insert into adaptation values(10 , 'Fiction' , 'Romeo and Juliet' , 1982 , 8);

-- insert into adaptation values(11 , 'Mystery' , 'Harry Potter' , 1947 , 9);

-- insert into adaptation values(12 , 'Abstract' , 'Life' , 1978 , 10);



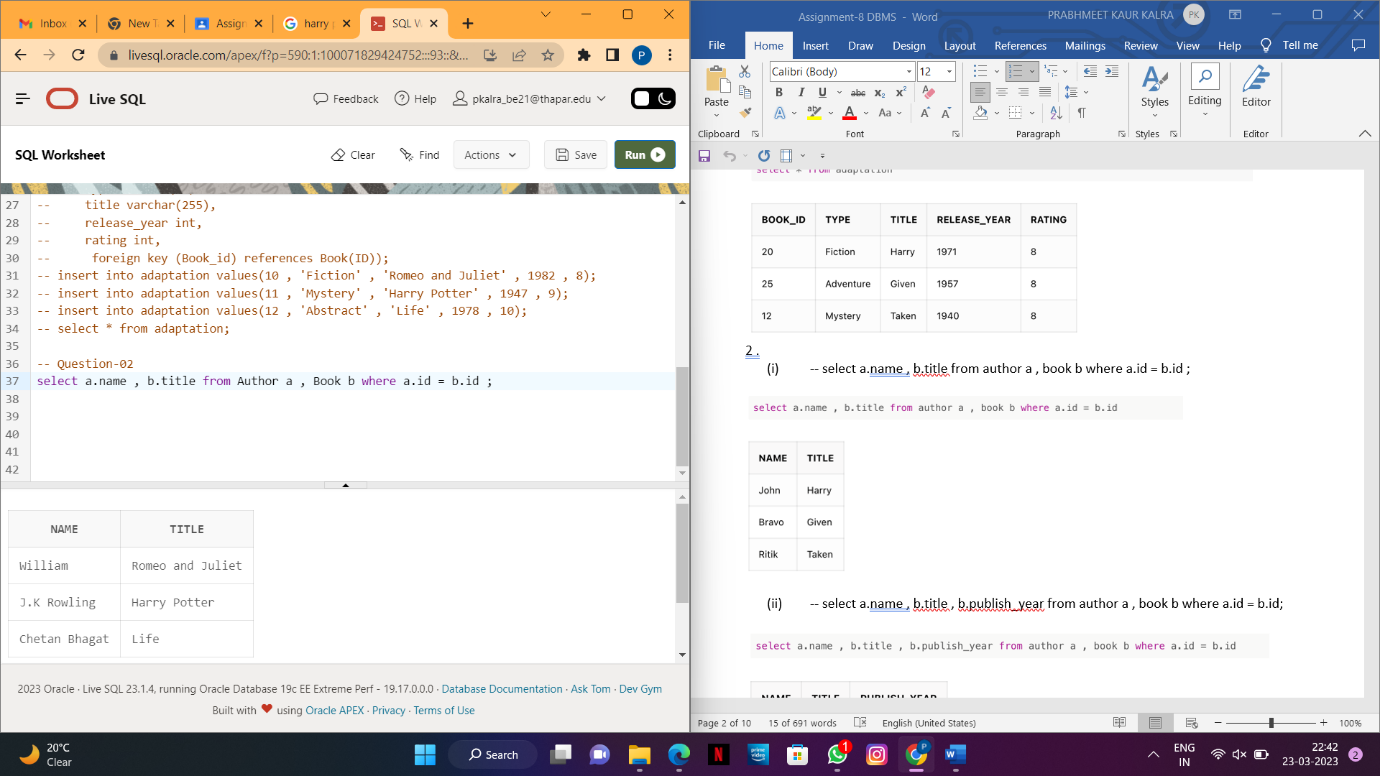
-- select \* from adaptation;



**2. Write SQL command for the following using Join:**

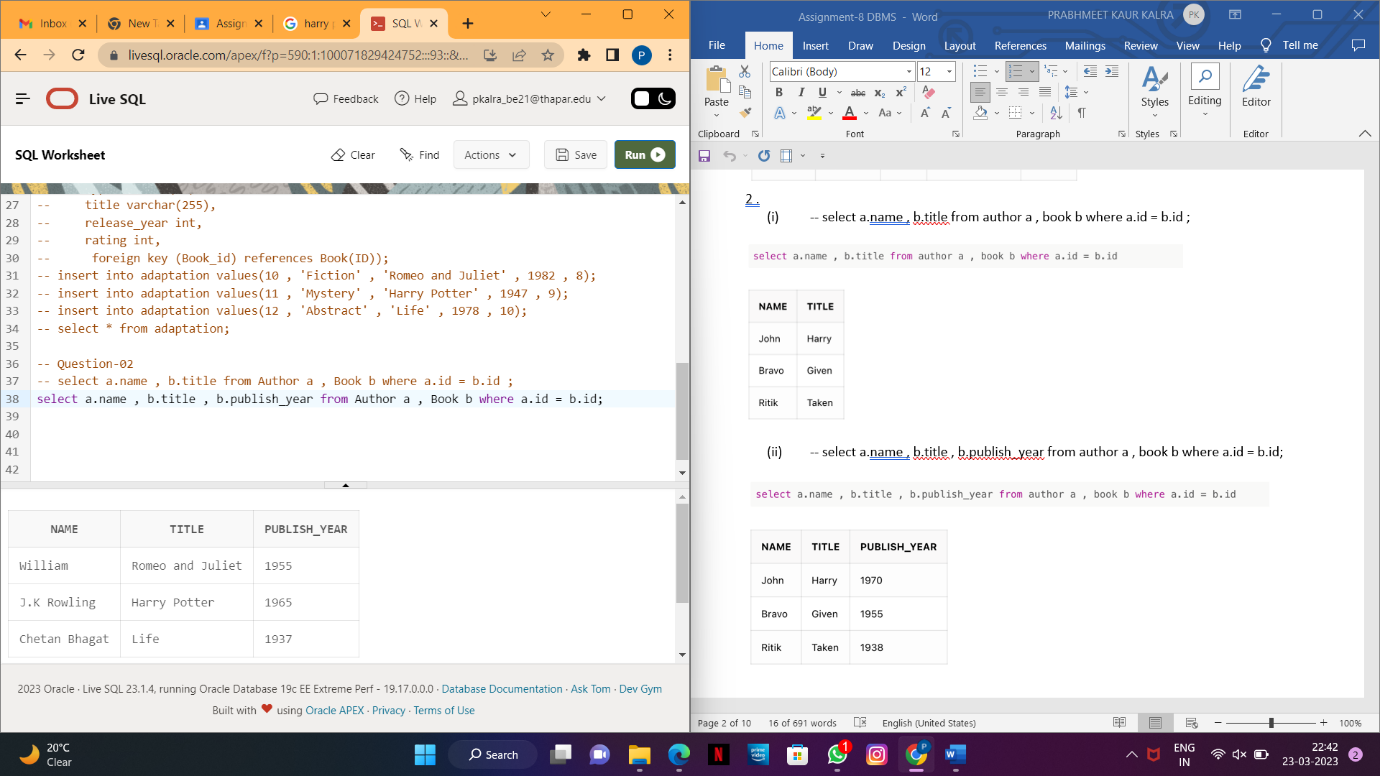
**i) Display the title of each book and the name of its author.**

**--** select a.name , b.title from Author a , Book b where a.id = b.id;



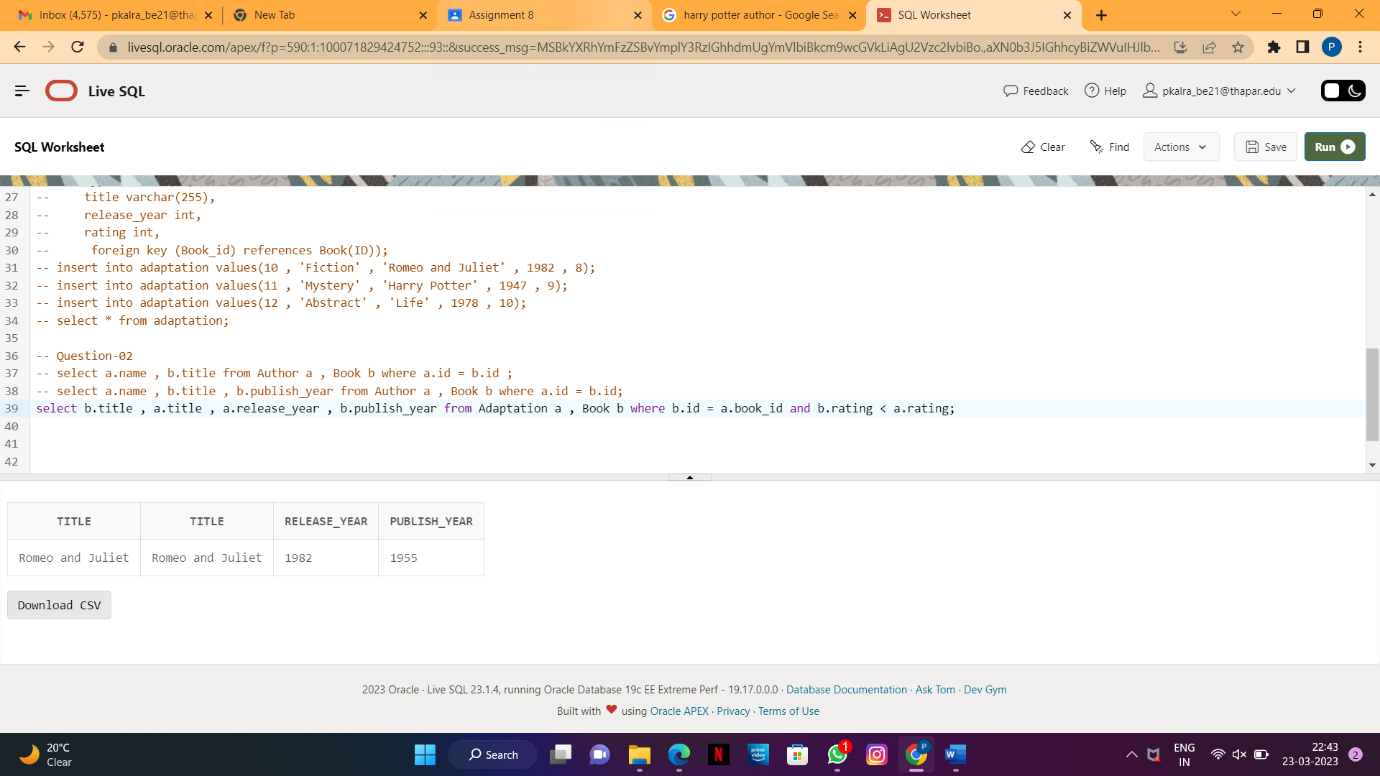
**ii) Display the name of each author together with the title of the book they wrote and the year in which that book was published**

--select a.name , b.title , b.publish\_year from Author a , Book b where a.id = b.id;



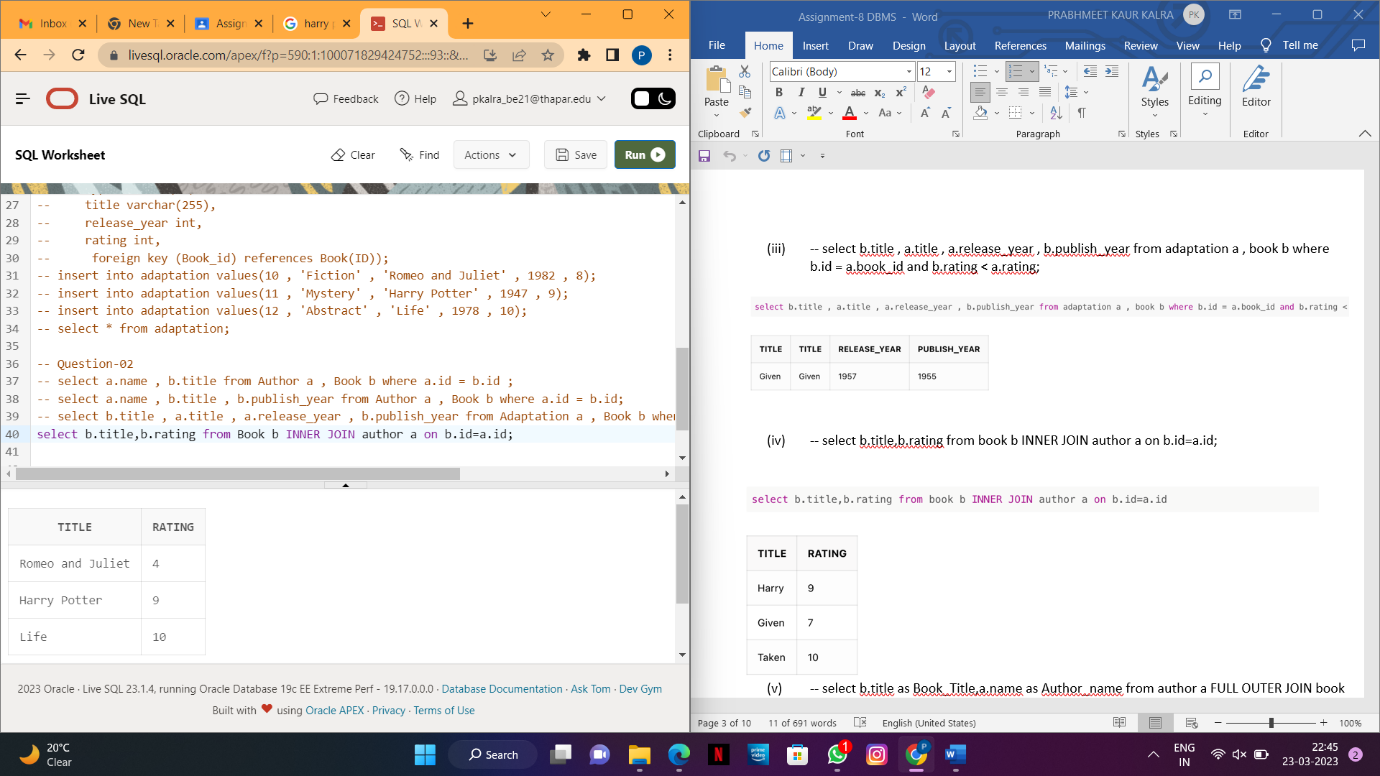
**iii) For each book, show its title, adaptation title, adaptation year, and publication year.**

-- select b.title , a.title , a.release\_year , b.publish\_year from Adaptation a , Book b where b.id = a.book\_id and b.rating < a.rating;



**iv) Display the title of each book together with its rating. Consider only those books that were published by authors who are still alive. (Use Inner Join).**

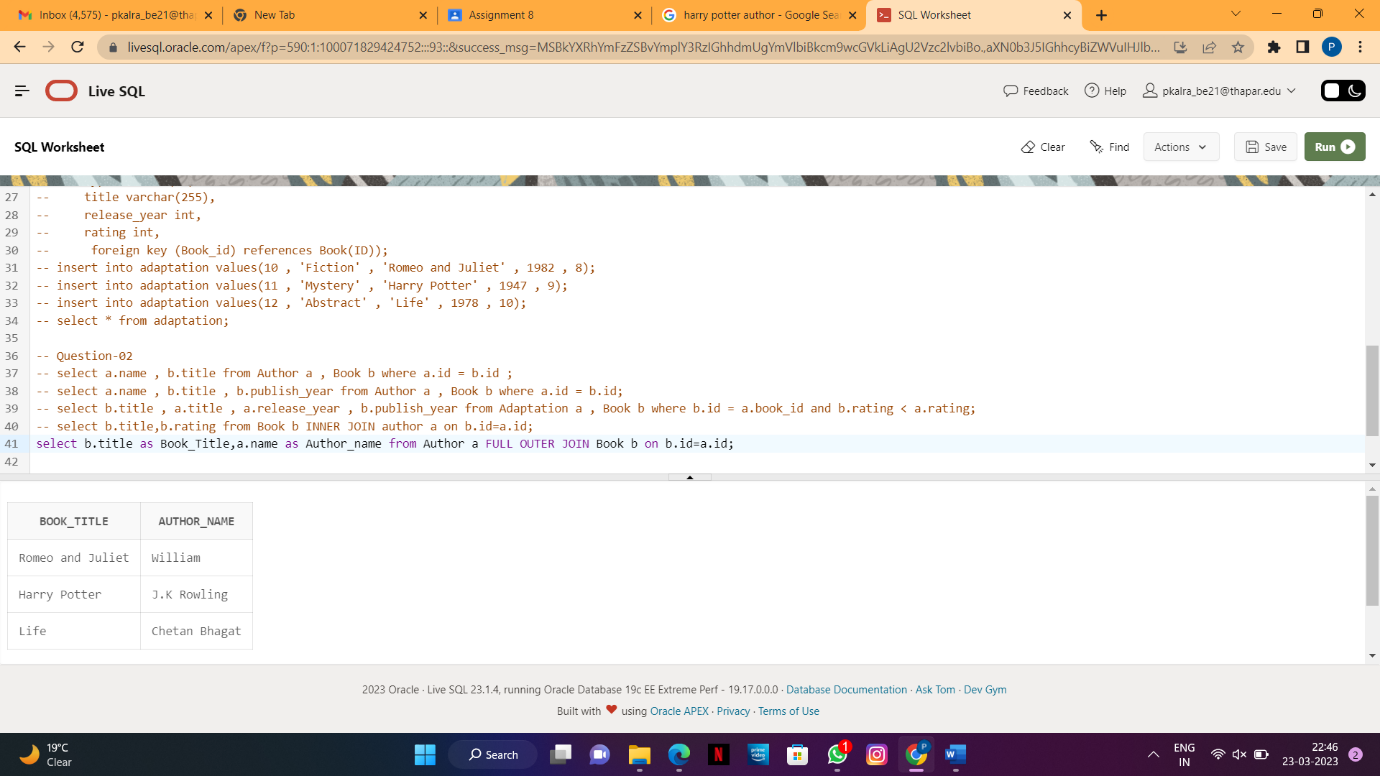
-- select b.title,b.rating from Book b INNER JOIN author a on b.id=a.id;



**v) Display the title of each book along with the name of its author. Show all books, even those without an author. Show all authors, even those who haven't published a book yet.**

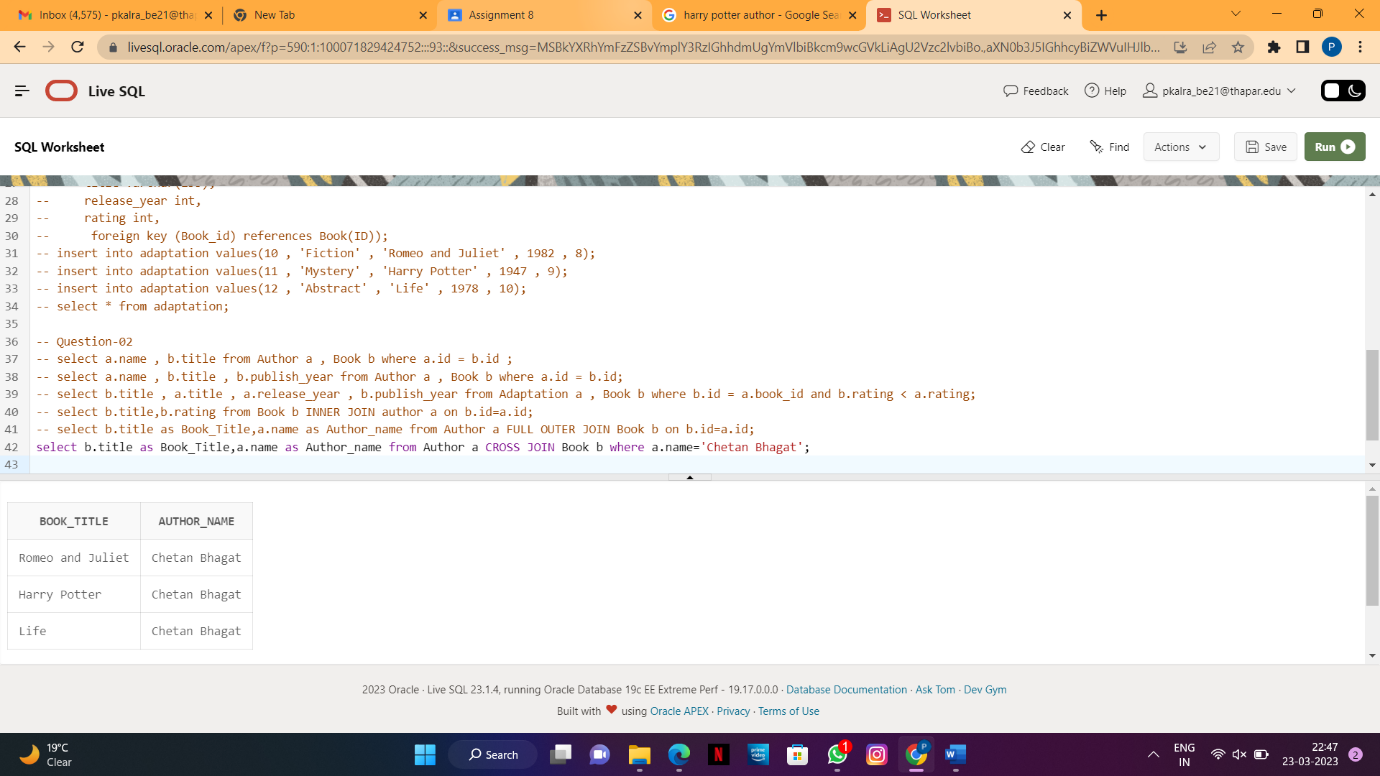
**(Use Full JOIN).**

-- select b.title as Book\_Title,a.name as Author\_name from Author a FULL OUTER JOIN Book b on b.id=a.id;



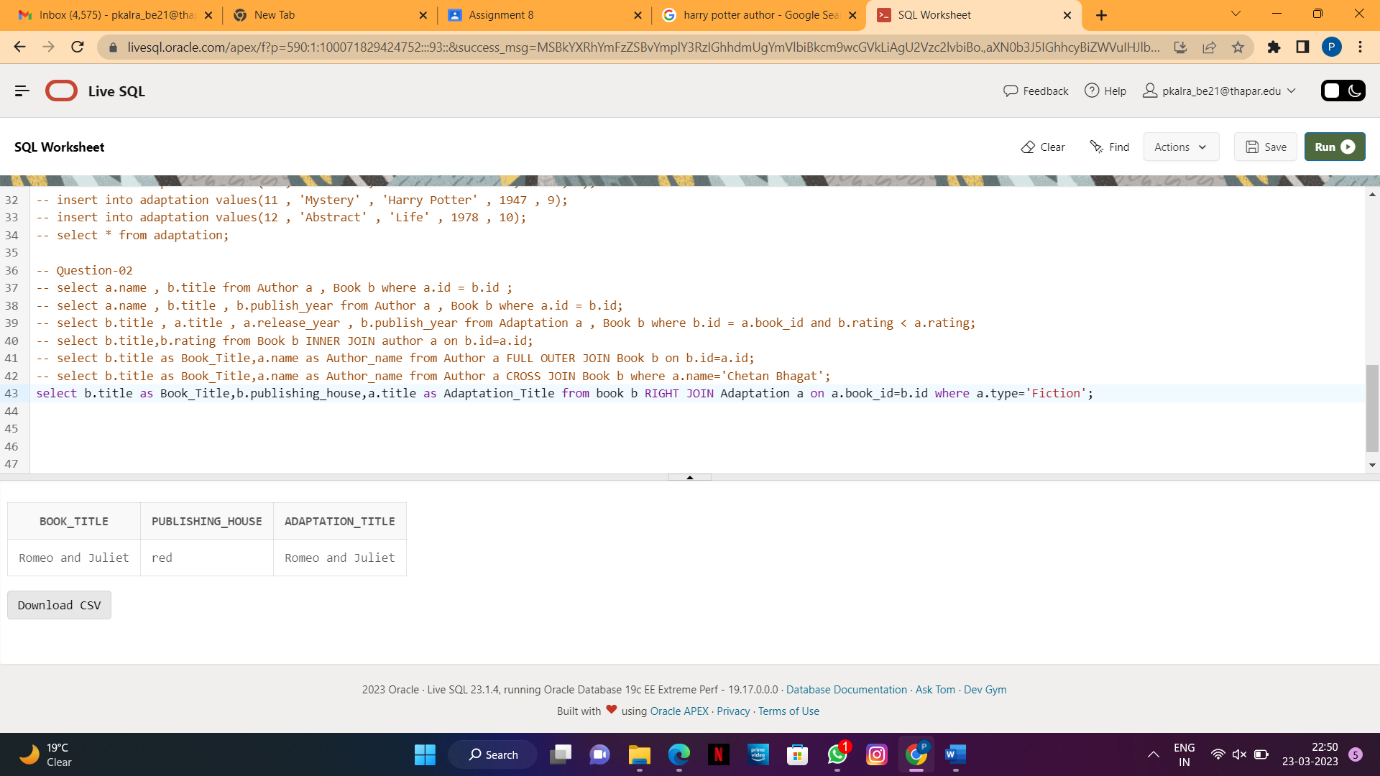
**vi) Generate all possible pairs of book titles and author names. Consider only books whose author's name is ‘Chetan Bhagat’ (Use Cross JOIN).**

-- select b.title as Book\_Title,a.name as Author\_name from Author a CROSS JOIN Book b where a.name='Chetan Bhagat';



**vii) Select each book's title, the name of its publishing house and the title of its adaptation on the type of the adaptation (‘Movie’). (Use Right JOIN).**

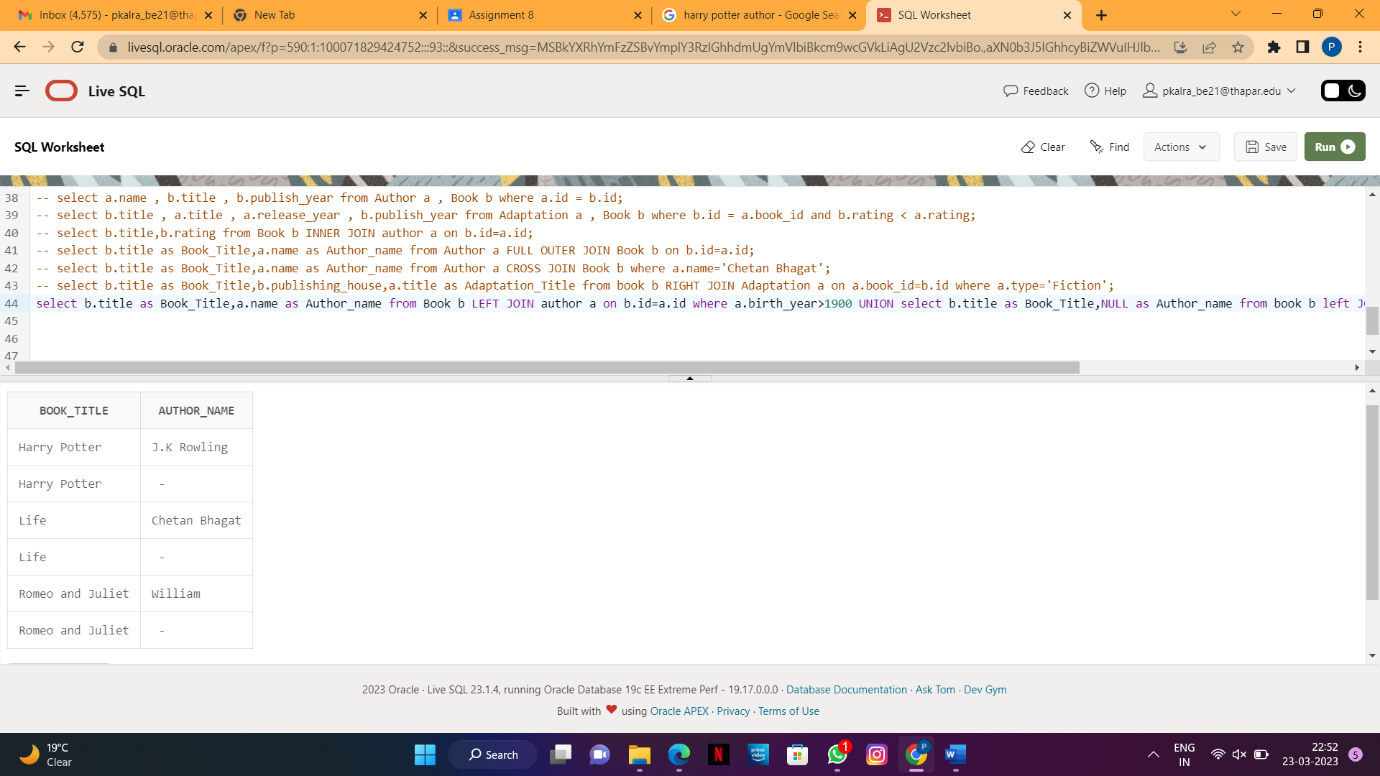
-- select b.title as Book\_Title,b.publishing\_house,a.title as Adaptation\_Title from book b RIGHT JOIN Adaptation a on a.book\_id=b.id where a.type='Fiction';



**viii) Show the title of each book and the name of its author — but only if the author was born**

**in the 20th century. Otherwise, the author's name field should be NULL (Use Left JOIN).**

-- select b.title as Book\_Title,a.name as Author\_name from Book b LEFT JOIN author a on b.id=a.id where a.birth\_year>1900 UNION select b.title as Book\_Title,NULL as Author\_name from book b left JOIN author a on b.id=a.id where a.birth\_year<1950;



3**. Consider the following relation and execute the given queries(Aggregate/Group By/Having):**

-- create table winners(year int,

--subject varchar(20),

--winner varchar(255) ,

--country varchar(25) ,

--category varchar(25));

-- insert into winners values(1970,'Physics','Hannes Alfven','Sweden','Scientist');

-- insert into winners values(1970,'Physics','Louis Neel','France','Scientist');

-- insert into winners values(1970,'Chemistry','Luis Federico Leloir','France','Scientist');

-- insert into winners values(1970,'Physiology','Ulf von Euler','Sweden','Scientist');

-- insert into winners values(1970,'Physiology','Bernard Katz','Germany','Scientist');

-- insert into winners values(1970,'Literature','Aleksandr Solzhenitsyn','Russia','Linguist');

-- insert into winners values(1970,'Economics','Paul Samuelson','USA','Economist');

-- insert into winners values(1970,'Physiology','Julius Axelrod','USA','Scientist');

-- insert into winners values(1971,'Physics','Dennis Gabor','Hungary','Scientist');

-- insert into winners values(1971,'Chemistry','Gerhard Herzberg','Germany','Scientist');

-- insert into winners values(1971,'Peace','Willy Brandt','Germany','Chancellor');

-- insert into winners values(1971,'Literature','Pablo Neruda','Chile','Linguist');

-- insert into winners values(1971,'Economics','Simon Kuznets','Russia','Economist');

-- insert into winners values(1978,'Peace','Anwar Al-Sadat','Egypt','President');

-- insert into winners values(1978,'Peace','Menachem Begin','Israel','Prime Minister');

-- insert into winners values(1987,'Chemistry','Donald J. Cram','USA','Scientist');

-- insert into winners values(1987,'Chemistry','Jean-Marie Lehn','France','Scientist');

-- insert into winners values(1987,'Physiology','Susumu Tonegawa','Japan','Scientist');

-- insert into winners values(1994,'Economics','Reinhard Selten','Germany','Economist');

-- insert into winners values(1994,'Peace','Yitzhak Rabin','Israel','Prime Minister');

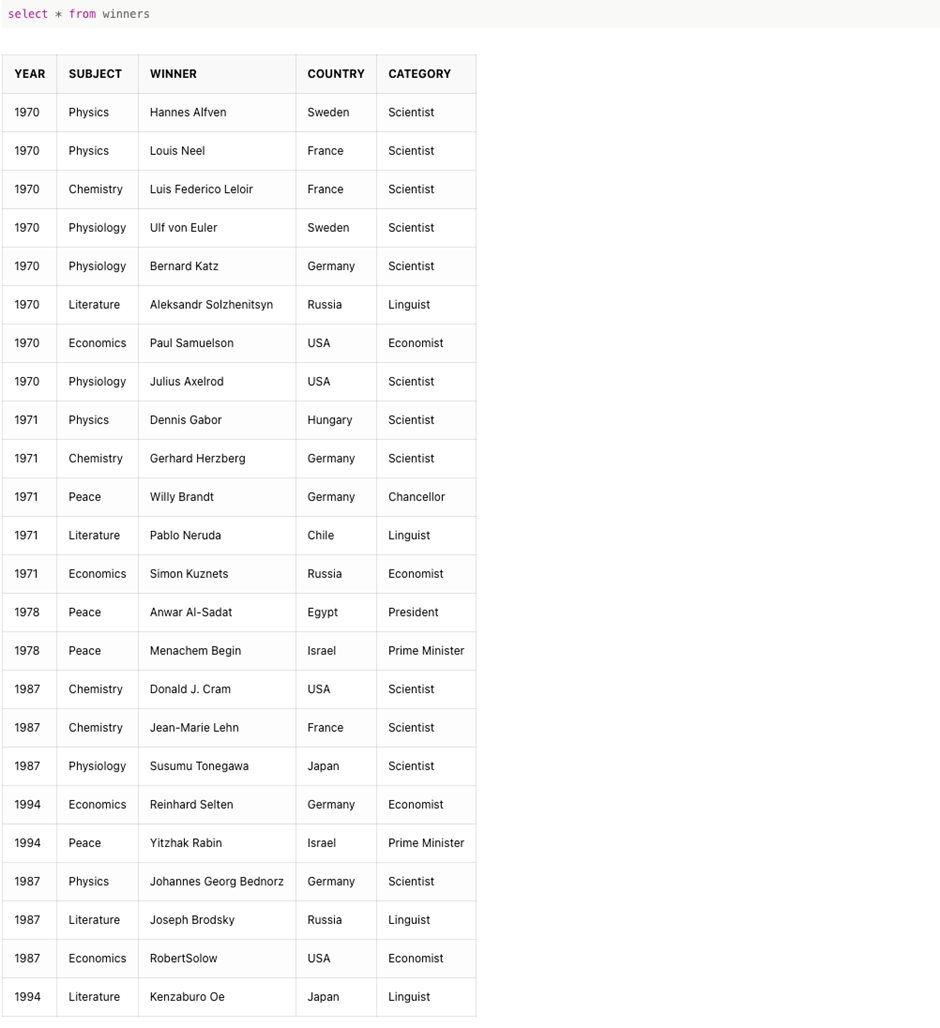
-- insert into winners values(1987,'Physics','Johannes Georg Bednorz','Germany','Scientist');

-- insert into winners values(1987,'Literature','Joseph Brodsky','Russia','Linguist');

-- insert into winners values(1987,'Economics','RobertSolow','USA','Economist');

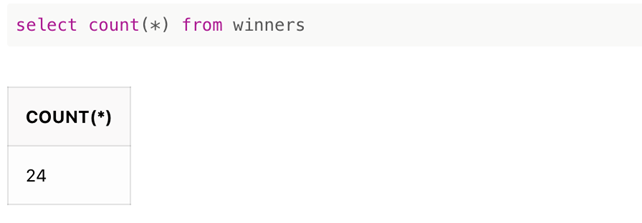
-- insert into winners values(1994,'Literature','Kenzaburo Oe','Japan','Linguist');

-- select \* from winners;



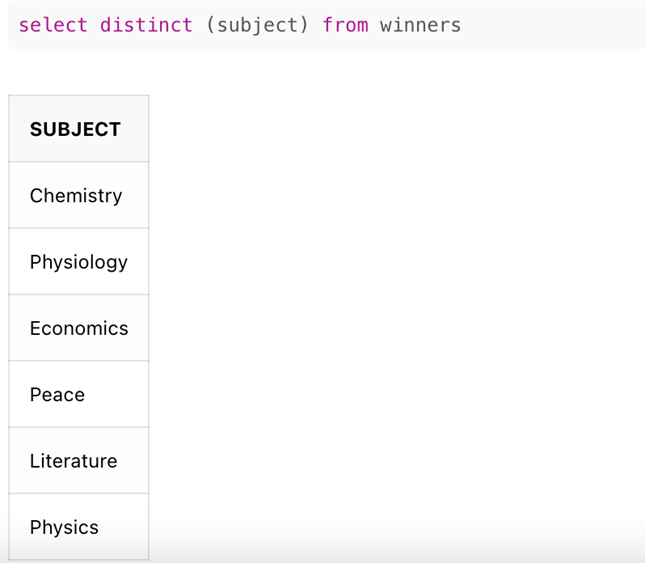
**i)Show the total number of prizes awarded.**

-- select count(\*) from winners;

****

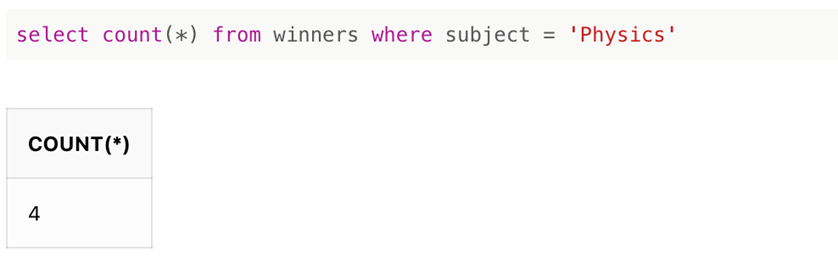
**ii) List each subject - just once**

-- select distinct (subject) from winners;



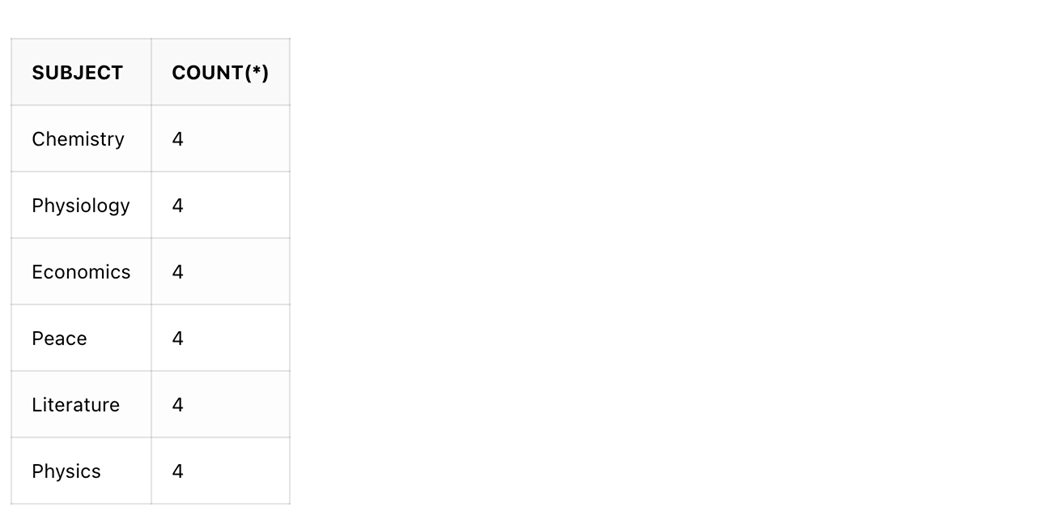
**iii) Show the total number of prizes awarded for Physics**

-- select count(\*) from winners where subject = 'Physics';



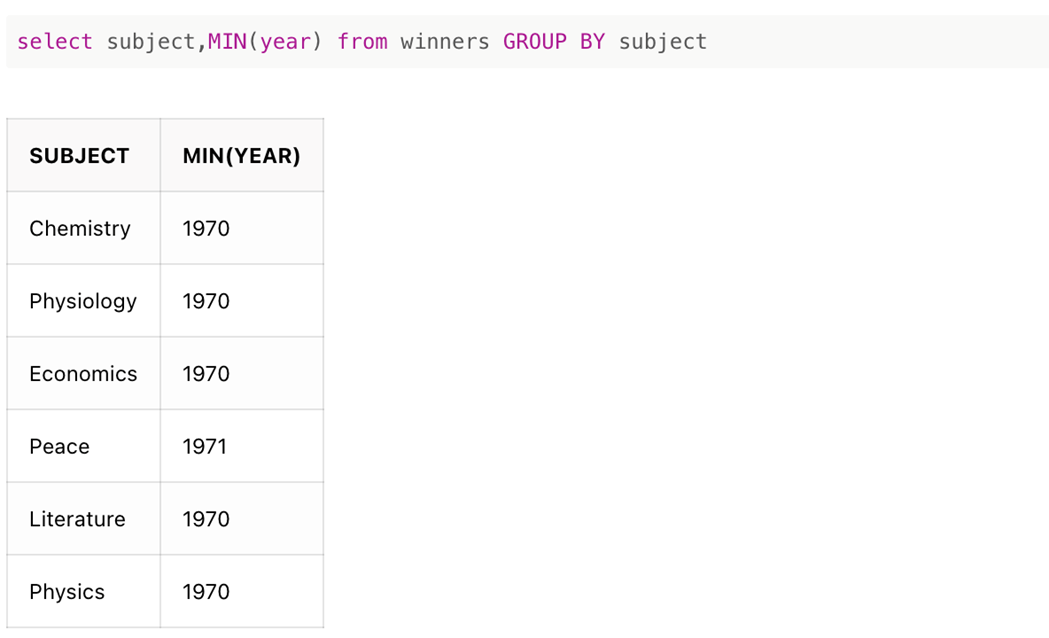
**iv) For each subject show the subject and the number of prizes.**

-- select subject,count(\*) from winners GROUP BY subject;

****

**v) For each subject show the first year that the prize was awarded.**

-- select subject,MIN(year) from winners GROUP BY subject;



**vi) For each subject show the number of prizes awarded in the year 2000.**

--select subject,count(\*) from winners where year = 2000 GROUP BY subject;

****

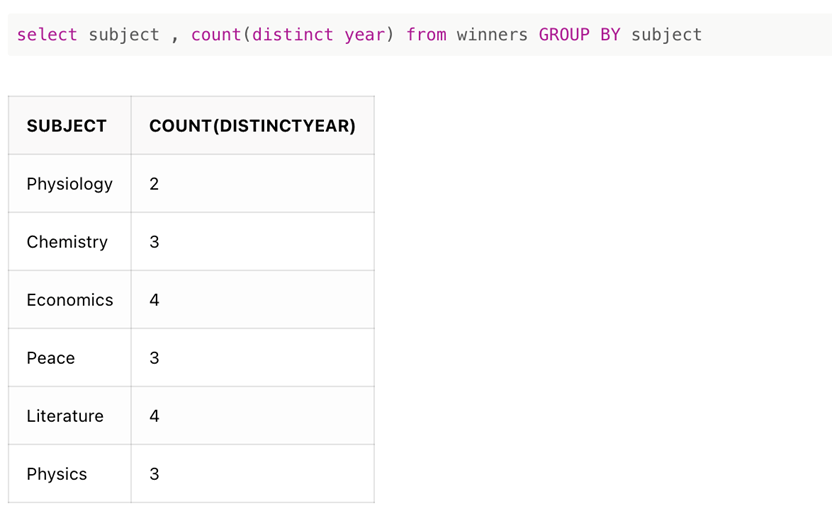
**vii) Show the number of different winners for each subject.**

--select distinct(subject) , count(distinct (winner)) from winners GROUP BY subject;



**viii) For each subject show how many years have had prizes awarded.**

--select subject , count(distinct year) from winners GROUP BY subject;



**ix) Show the years in which three prizes were given for Physics.**

--select year from winners where subject = 'Physics' GROUP BY year having count(year) = 3;

****

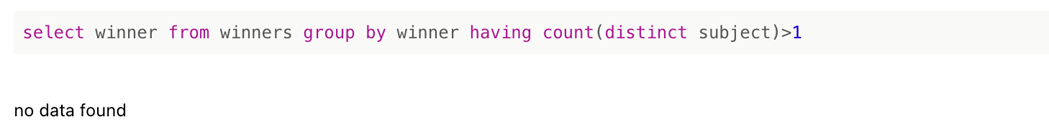
**x) Show winners who have won more than once.**

--select winner from winners GROUP BY winner having count(winner)>1;



**xi) Show winners who have won more than one subject.**

--select winner from winners group by winner having count(distinct subject)>1;



**xii) Show the year and subject where 3 prizes were given. Show only years 2000 onwards.**

--select year, subject from winners where year>=2000 group by year, subject having count(distinct winner)=3;

