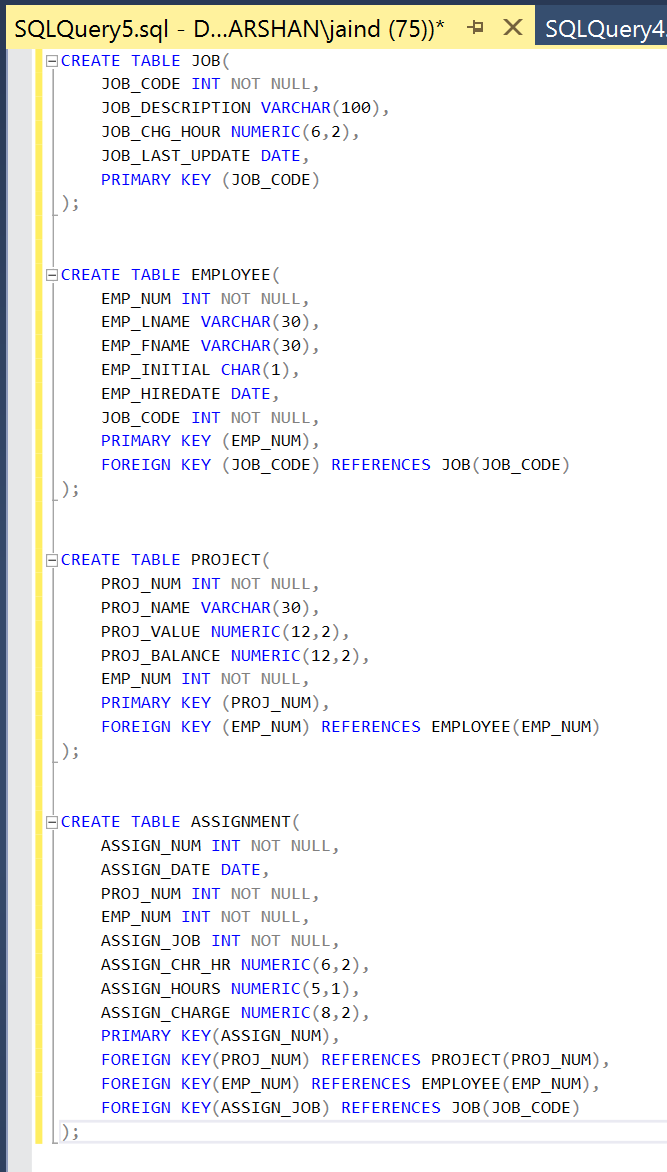
**Task 1:**



**Task 2:**

**Entity Integrity:**

* Entity integrity ensures that each table has a unique identifier as primary key.
* Each table has a unique identifier such as:

JOB\_CODE for JOB Table.

EMP\_NUM for EMPLOYEE Table.

PROJ\_NUM for PROJECT Table.

ASSIGN\_NUM for ASSIGNMENT Table.

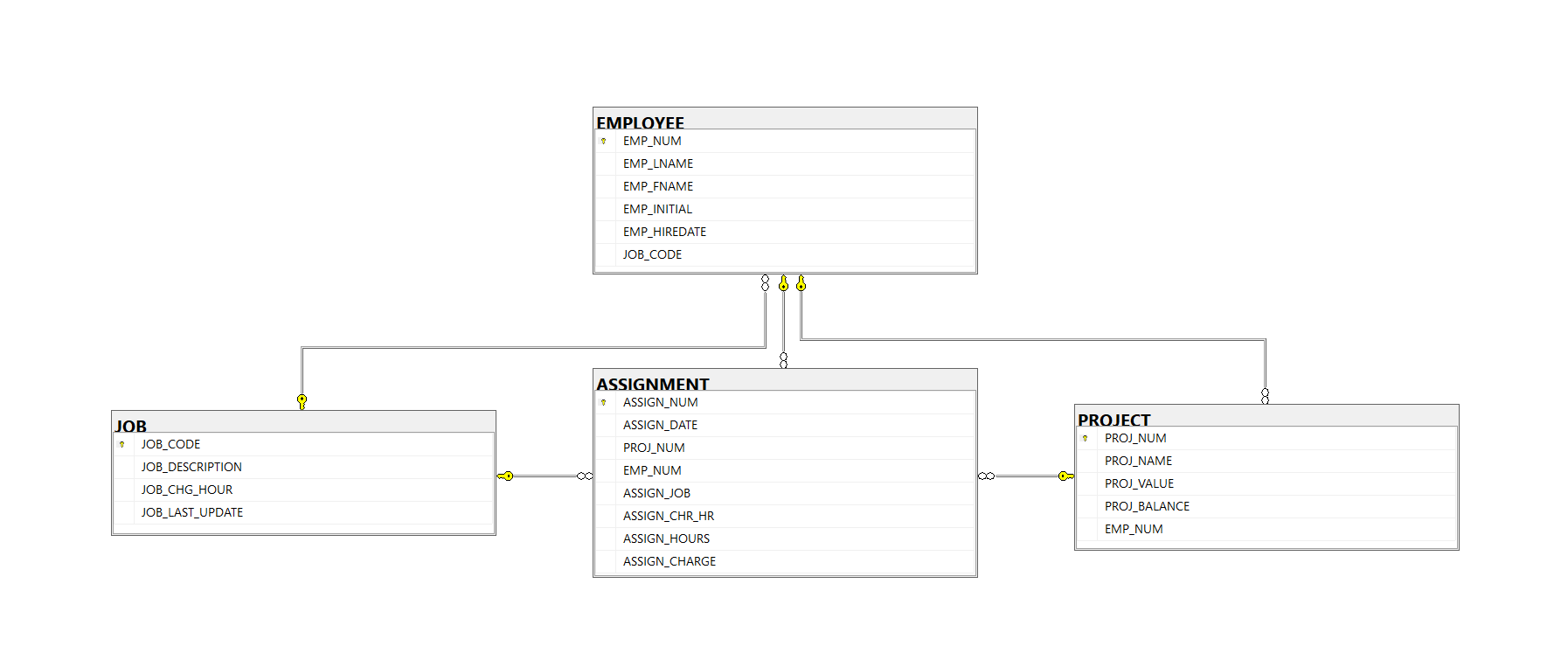
**Reference Integrity:**

* Reference integrity ensures valid relationships between tables by connecting them with Foreign Keys.
* In the EMPLOYEE Table: JOB\_CODE is a foreign key referencing JOB.JOB\_CODE. Ensures valid job assignment for each employee.
* In the PROJECT Table: EMP\_NUM is a foreign key referencing EMPLOYEE.EMP\_NUM. Ensures validity of the employee managing the project.
* In the ASSIGNMENT Table: PROJ\_NUM is a foreign key referencing PROJECT.PROJ\_NUM, EMP\_NUM is a foreign key referencing EMPLOYEE.EMP\_NUM and ASSIGN\_JOB is a foreign key referencing JOB.JOB\_CODE. Ensures validity of existing projects, employees and job roles.
* This ensures that if any data needs to be removed from a table, the data from the table referencing it with a foreign key should be removed first.

**Domain Integrity:**

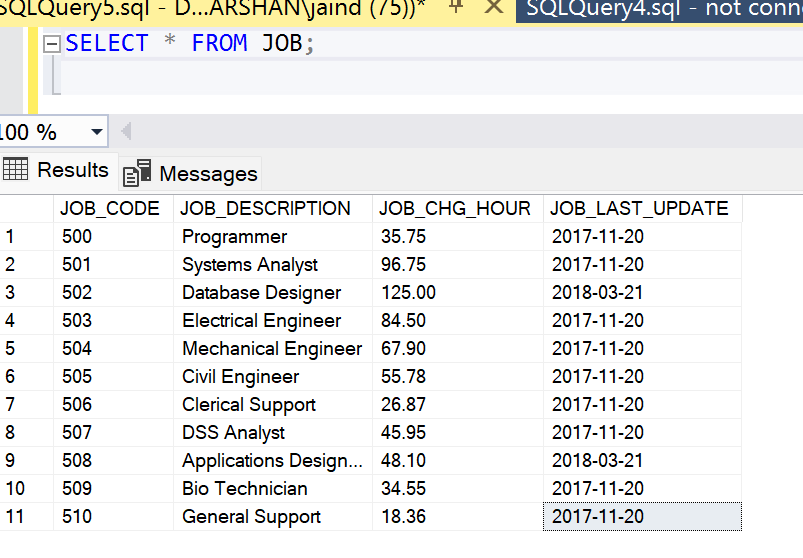
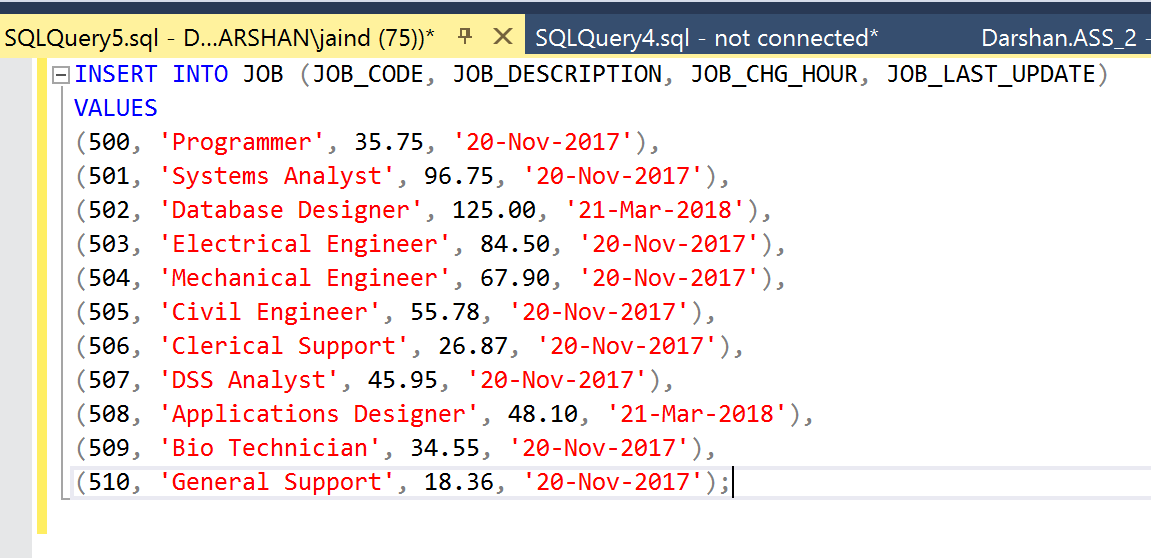
* Domain integrity ensures that columns have appropriate data types and constraints to maintain data validity.
* There is a DATE constraint on ASSIGN\_DATE, EMP\_HIREDATE AND JOB\_LAST\_UPDATE. This ensures only valid dates are entered.
* There is a NUMERIC constraint on JOB\_CHG\_HOUR(6,2) that ensures the maximum numbers should be 6 with 2 digits after the decimal, ASSIGN\_HOURS(5,1) that ensures that the maximum numbers should be 5 with just 1 digit after decimal and ASSIGN\_CHARGE(8,2) that ensures that the maximum numbers should be 8 with 2 digits after decimal.
* NUMERIC(5,1) CHECK (ASSIGN\_HOURS > 0) prevents negative values.

**DATABASE DIAGRAM:**

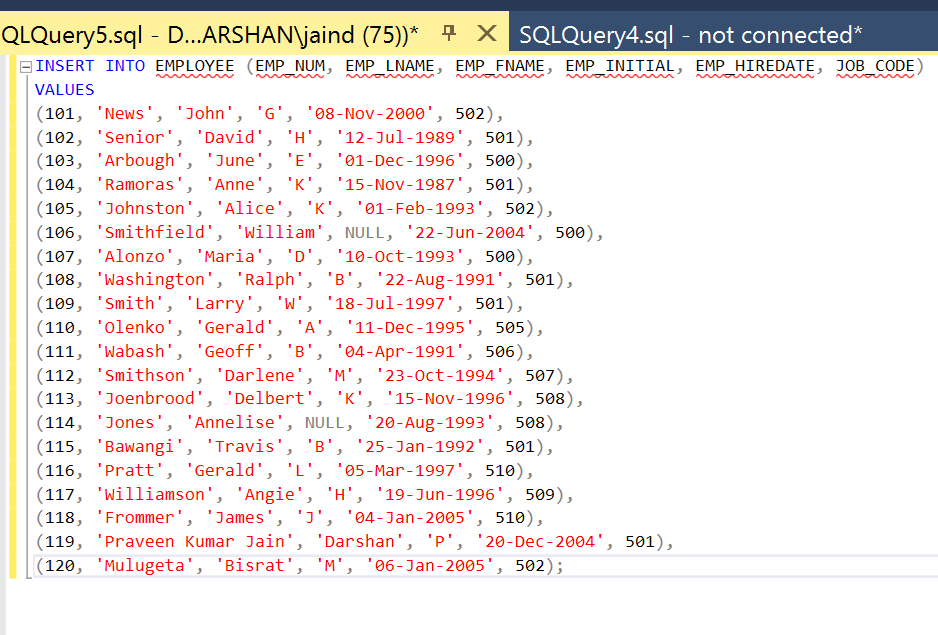
****

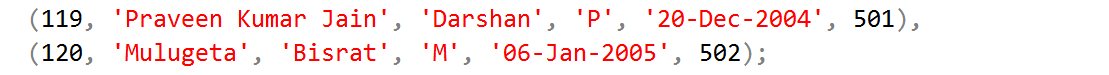
**Task 3:**

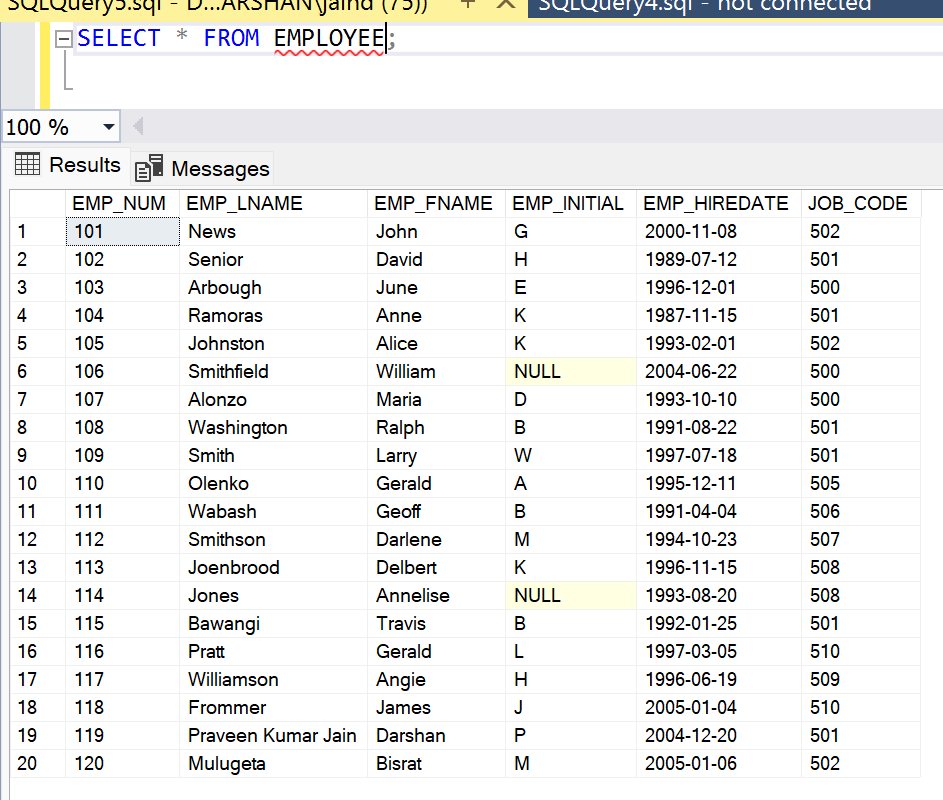
**Job Table:**

****

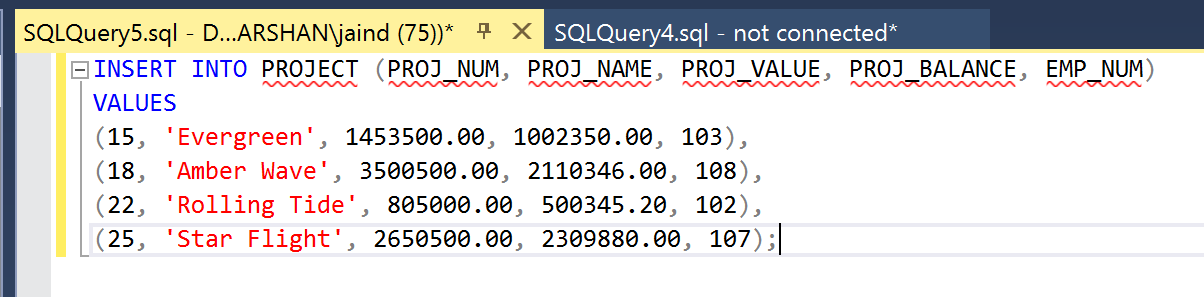
**Employee Table:**

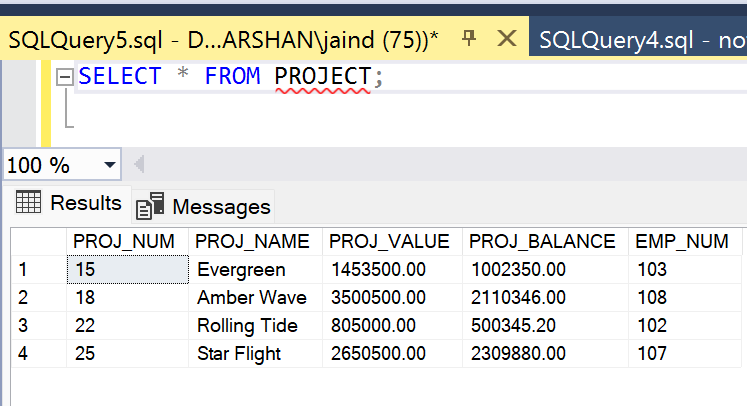
****

The last two rows of Employee table are added with student names as shown below:  


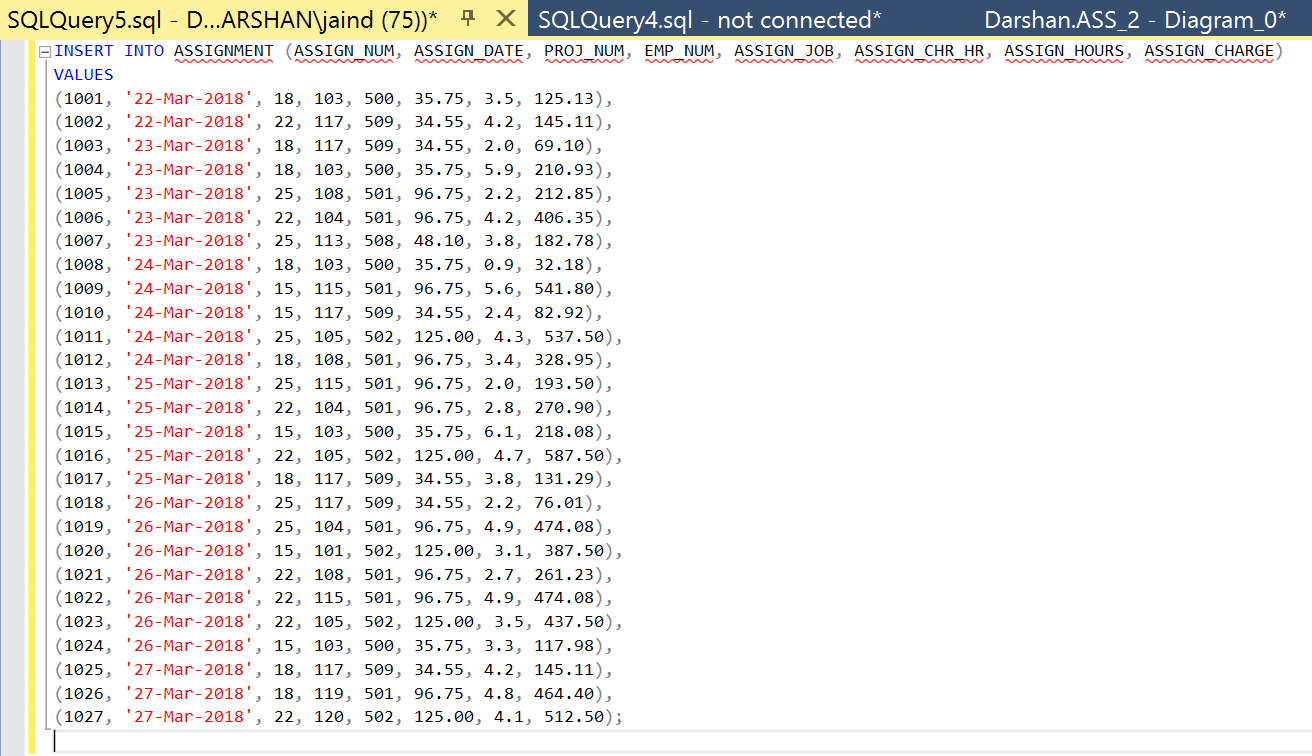


**Project Table:**

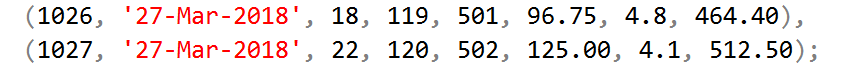


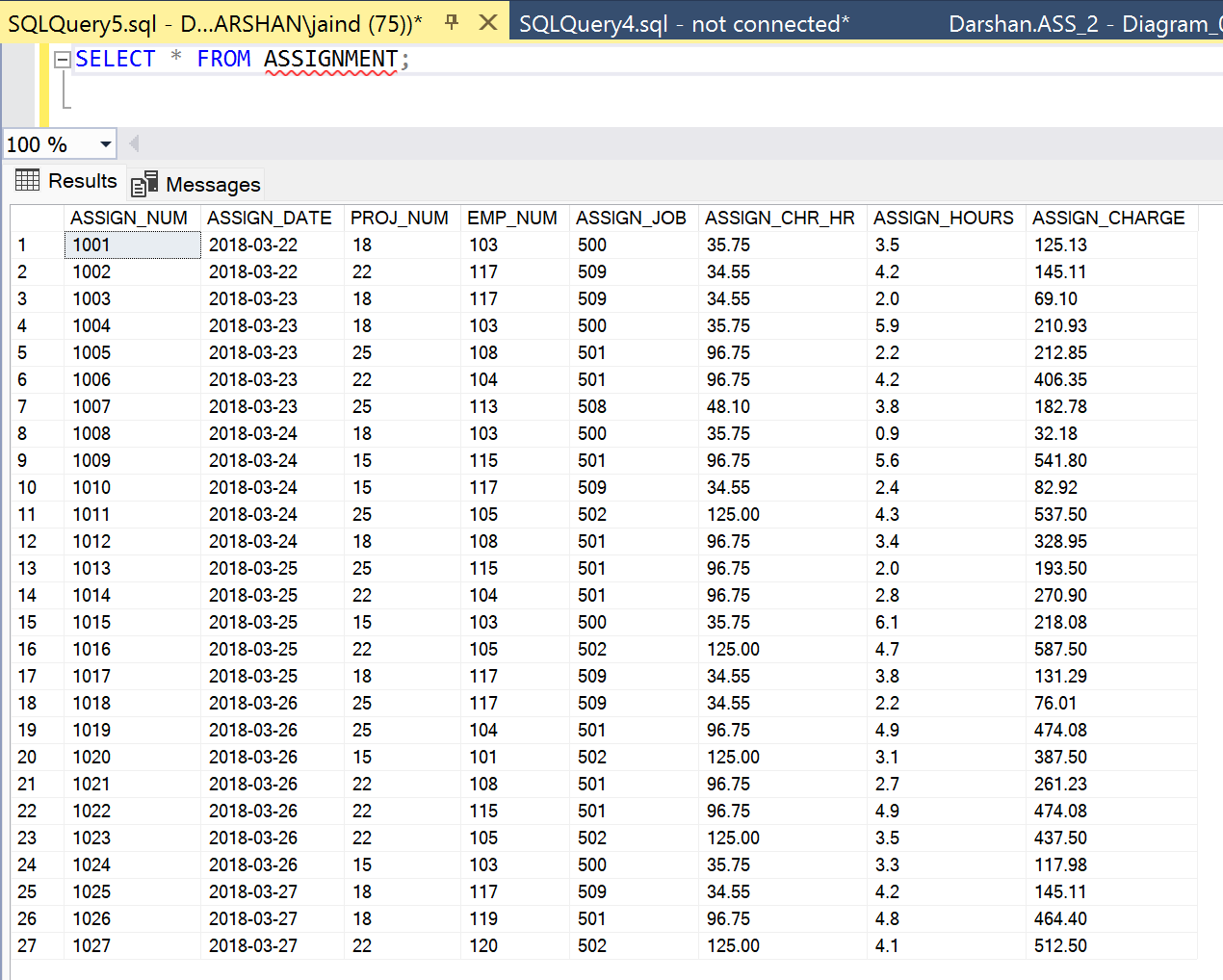


**Assignment Table:**

****

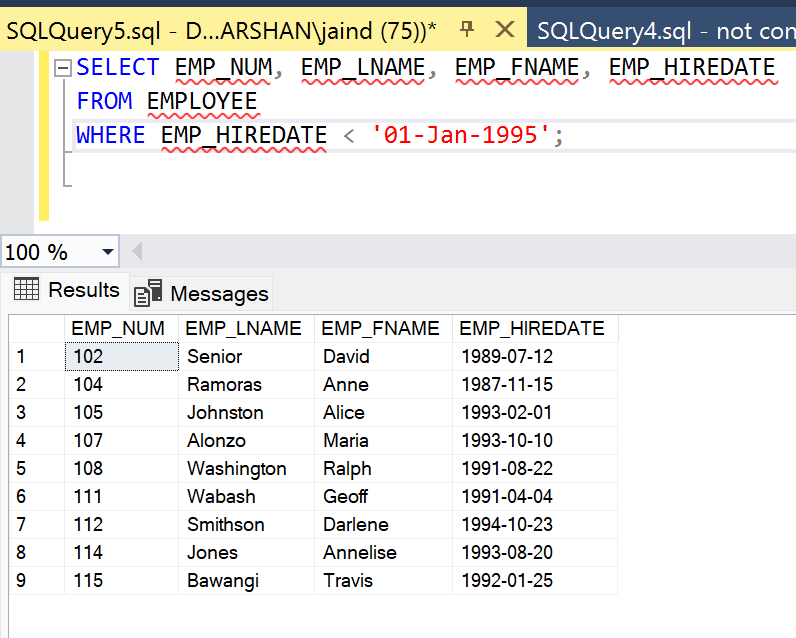
The last two rows of Assignment table are added with student names as shown below:



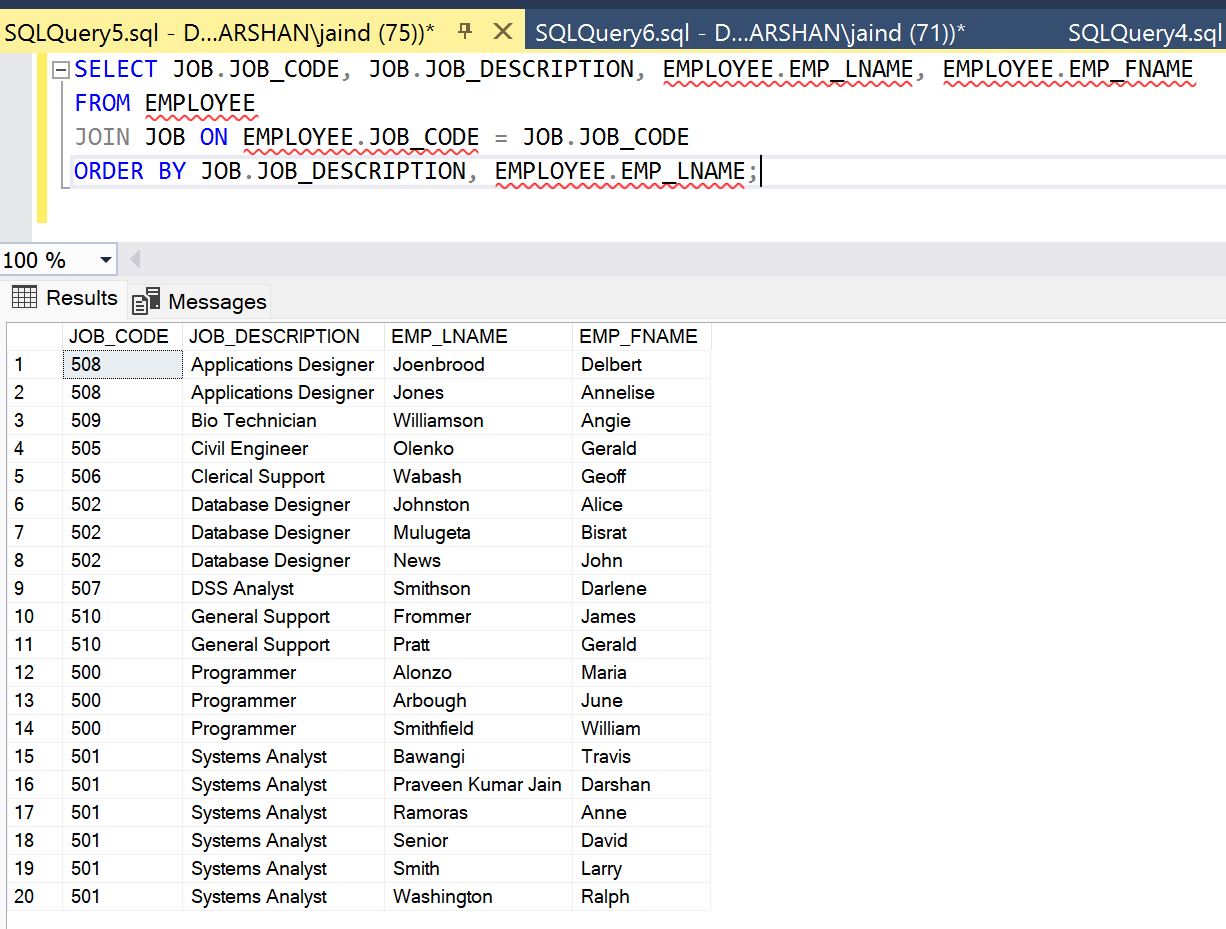


**Task 4:**

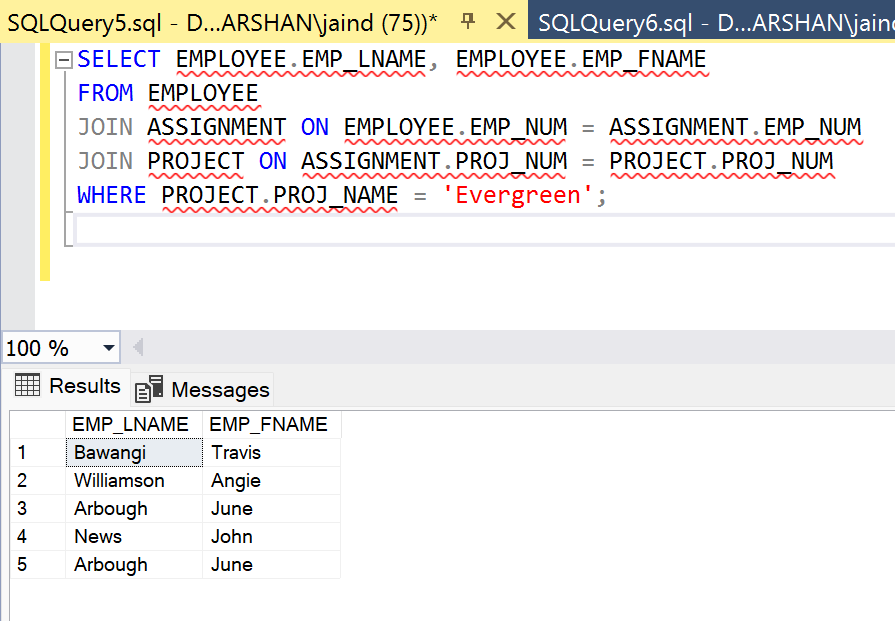
* Create and run a query to list employees who were hired before the year 1995.

****

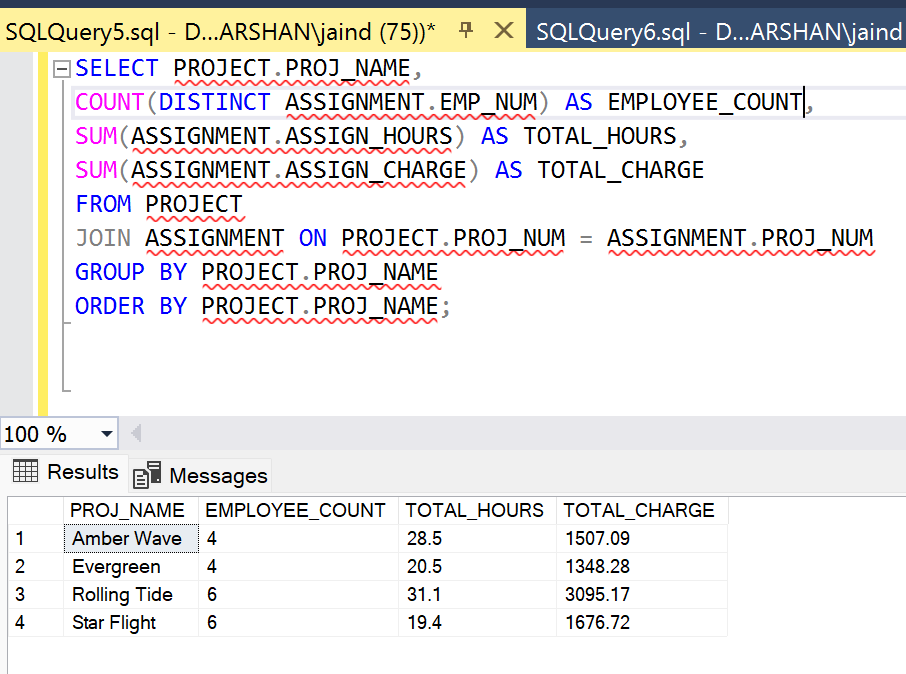
* Create and run a query to list job code, job description, and employee last name, first name sorted by job description and employee last name.

****

* Create and run a query to list the name of employees who were assigned to the Evergreen project.

****

* Create and run a query to list the project name, the number of employees worked for the project, and the total hours and total charges assigned to each project sorted by project name.

****

**Task 5:**

In task 1, we wrote commands to create the table by specifying the necessary primary keys, NOT NULL values and foreign keys. We also made sure to follow the same date type for foreign key reference to maintain data integrity.

In task 2, we have shown entity integrity, reference integrity and domain integrity in detail by specifying all the primary and foreign keys.

In task 3, we inserted sample data in all the tables with also specific rows of personalised data with our student names and executed the program. Two extra insert statements have been added to the EMPLOYEE table with student names to distinguish the table from other students and similarly two extra insert statements have been added to the ASSIGNMENT table referring to the same data added in the employee table.

In task 4, we have designed queries based on the given questions to achieve the required output.