**SQA Training Plan: Database SQL**



**ATEEB ADIL**

**Date: 11-Jan-22**

# Disclaimer

The information contained in these documents is confidential, privileged and only for the information of the intended recipient and may not be used, published or redistributed without the prior written consent of Agilosoft. The opinions expressed are in good faith and while every care has been taken in preparing these documents. Agilosoft makes no representations and gives no warranties of whatever nature in respect of these documents, including but not limited to the accuracy or completeness of any information, facts and/or opinions contained therein.

Table of Contents

[Disclaimer 2](#_Toc92801695)

[1 Introduction: 5](#_Toc92801696)

[2 Agilosoft SQL Training – Session 1 - Labs 5](#_Toc92801697)

[2.1 Lab2: Inner Join and Other Set Operations 5](#_Toc92801698)

[2.1.1 Run the below queries and observe the number of rows returned: 5](#_Toc92801699)

[2.1.2 Run the following queries and observe the number of rows returned: 5](#_Toc92801700)

[2.1.3 Run the following queries and observe the number of rows returned: 5](#_Toc92801701)

[2.1.4 Run the following queries and observe the rows returned: 5](#_Toc92801702)

[2.2 Lab 3: Where Clause and Operators 5](#_Toc92801703)

[2.2.1 List down professors that are at least 50 years old and are Professors: 5](#_Toc92801704)

[2.2.2 List down Students that have an ‘e’ in their names: 5](#_Toc92801705)

[2.2.3 List down names of all projects together with the names of their principal investigators 5](#_Toc92801706)

[2.2.4 List down names of all projects together with the names of their co-investigators: 5](#_Toc92801707)

[2.2.5 List down all those rows where the research assistant (student) is older than the principal investigator (professor): 5](#_Toc92801708)

[2.3 Lab 4: Partitioning and Aggregate Functions 5](#_Toc92801709)

[2.3.1 Run the below queries: 5](#_Toc92801710)

[2.3.2 Write the following queries: 6](#_Toc92801711)

[3 Agilosoft SQL Training – Session 2 - Labs 7](#_Toc92801712)

[3.1 Lab 5: Data Types & NULL 7](#_Toc92801713)

[3.1.1 Put data in some rows of Student Table in the Advisor column (put a valid Professor.SSN) 7](#_Toc92801714)

[3.1.2 Run the following queries: 8](#_Toc92801715)

[3.1.3 Identify the correct data type for the following: 8](#_Toc92801716)

[3.2 Lab 6: Insert, Update and Delete Operations 9](#_Toc92801717)

[3.2.1 First find out the last value of identity field: 9](#_Toc92801718)

[3.2.2 Insert a new row in the professor table 9](#_Toc92801719)

[3.2.3 Find out the SSN of newly inserted record: 9](#_Toc92801720)

[3.2.4 Insert a new row by specifying the column list 9](#_Toc92801721)

[3.2.5 Insert a new row by explicitly specifying a value for the identity column 9](#_Toc92801722)

[3.2.6 Delete From Professor Where SSN > 1000 9](#_Toc92801723)

[3.2.7 Delete From Registration 10](#_Toc92801724)

[3.2.8 Truncate Table Registration 10](#_Toc92801725)

[3.2.9 Update Professor Set ResearchSpeciality = ‘Computational Intelligence’ Where SSN = 1097 10](#_Toc92801726)

[4 Agilosoft SQL Training – Session 2b - Labs 10](#_Toc92801727)

[4.1 Lab 7: Subqueries 10](#_Toc92801728)

[4.1.1 List down those students who are not part of any research projects (who are not research assistants in any of the projects): 10](#_Toc92801729)

[4.1.2 Identify the oldest student 11](#_Toc92801730)

[4.1.3 Identify the professors who are older than the average age of all professors 11](#_Toc92801731)

[4.1.4 Identify professors who are involved in at least two projects (either as principle or co-investigator) and are less than 40 years old. 11](#_Toc92801732)

# Introduction:

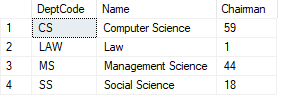
The Following document showcases SQL session labs tasks and practical hands-on with SQL queries.

# Agilosoft SQL Training – Session 1 - Labs

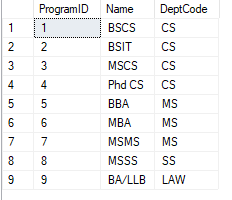
## Lab2: Inner Join and Other Set Operations

### Run the below queries and observe the number of rows returned:

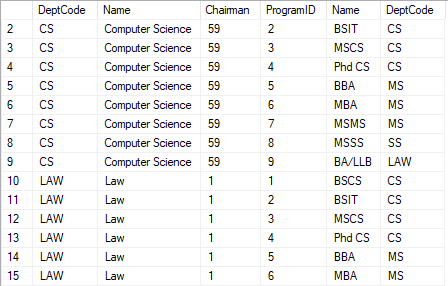
#### SELECT \* FROM DEPARTMENT



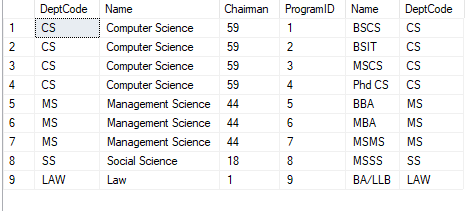
#### SELECT \* FROM PROGRAM



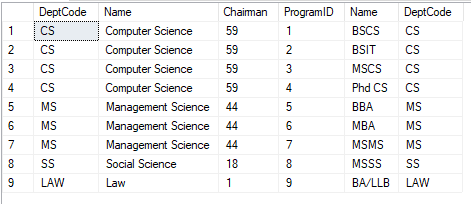
#### SELECT \* FROM DEPARTMENT, PROGRAM



#### Select \* From Department INNER JOIN Program ON Department.DeptCode = Program.DeptCode



#### Select \* From Department, Program Where Department.DeptCode = Program.DeptCode



### Run the following queries and observe the number of rows returned:

#### Select PrincipalInvestigator From Project

#### Select SSN From CoInvestigator

#### Select PrincipalInvestigator From Project Union Select SSN From CoInvestigator

#### Select PrincipalInvestigator From Project Union All Select SSN From CoInvestigator

### Run the following queries and observe the number of rows returned:

#### Select \* From Student

#### Select \* From ResearchAssistant

#### Select SSN From Student Except Select SSN From ResearchAssistant

#### Select SemesterCode From Semester Except Select SemesterCode From CourseInstance

### Run the following queries and observe the rows returned:

#### Select PrincipalInvestigator From Project Insersect Select SSN From Coinvestigator

#### Select DeptCode From Department INTERSECT Select DeptCode From CourseDepartment

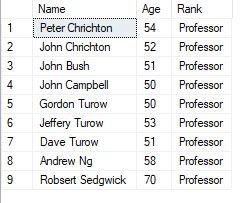
## Lab 3: Where Clause and Operators

### List down professors that are at least 50 years old and are Professors:

Select Professor.Name, Professor.Age, Professor.Rank

From Professor

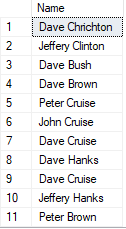
Where Professor.Age >= 50 AND Professor.Rank = 'Professor'



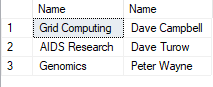
### List down Students that have an ‘e’ in their names:

Select Student.Name

From Student

WHERE Student.name LIKE '%e%'  
  


### List down names of all projects together with the names of their principal investigators

Select Project.Name, Professor.Name   
From Project Inner Join Professor On   
Project.PrincipalInvestigator = Professor.SSN

### List down names of all projects together with the names of their co-investigators:

Select Project.Name , Professor.Name

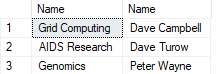
From Project Inner Join CoInvestigator On

Project.ProjectNumber = CoInvestigator.ProjectNumber

Inner Join Professor On

Project.PrincipalInvestigator = Professor.SSN

Group by Project.Name, Professor.Name



### List down all those rows where the research assistant (student) is older than the principal investigator (professor):

Select Student.Name, Student.Age

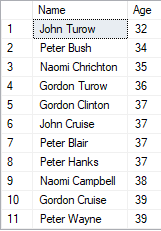
From Student Inner Join ResearchAssistant On

Student.SSN = ResearchAssistant.SSN

Inner Join Professor On

Student.Age > Professor.Age

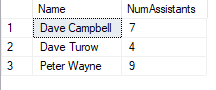
GROUP BY Student.Name, Student.Age



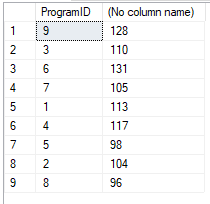
## Lab 4: Partitioning and Aggregate Functions

### Run the below queries:

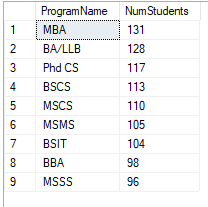
#### Select P.Name, Count(RA.SSN) As NumAssistants From Professor P Inner Join Project R On P.SSN = R.PrincipalInvestigator Inner Join ResearchAssistant RA On R.ProjectNumber = RA.ProjectNumber Group By P.Name



#### Select ProgramID, Count(1) From Student Group By ProgramID



#### Select P.Name As ProgramName, Count(1) As NumStudents From Program P, Student S Where P.ProgramID = S.ProgramID Group By P.Name Order By 2 Desc



### Write the following queries:

#### List down projects that have at least 3 research assistants

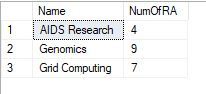
SELECT Project.Name , COUNT(Project.ProjectNumber) as NumOfRA

From Project , ResearchAssistant R

Where Project.ProjectNumber = R.ProjectNumber

Group by Project.Name

Having COUNT(Project.ProjectNumber) >= 3



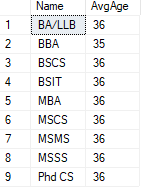
#### List down the average age of students of each program

SELECT Program.Name , AVG(Student.Age) as AvgAge

From Program, Student

WHERE Program.ProgramID = Student.ProgramID

GROUP BY Program.Name



#### List down names of professors who are either chairman of a department or are principle investigator of a project

Select Professor.Name

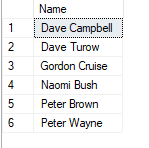
From Professor, Department, Project

WHERE Professor.SSN = Project.PrincipalInvestigator

OR

Professor.SSN = Department.Chairman

Group By Professor.Name



# Agilosoft SQL Training – Session 2 - Labs

## Lab 5: Data Types & NULL

### Put data in some rows of Student Table in the Advisor column (put a valid Professor.SSN)

INSERT INTO Student (Name, Age, ProgramID, Advisor)

VALUES ('AmberCUSTOM', '16', '2', '11');

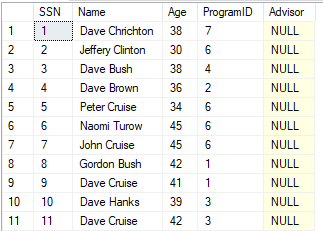


### Run the following queries:

#### Select \* From Student Where Advisor = NULL



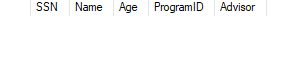
#### Select \* from Student Where Advisor Is Null



#### Select Count(\*) From Student Where Advisor Is Not Null



#### Select \* From Student Where Adviser < 5



### Identify the correct data type for the following:

#### A product code comprising of three alphabets or digits.

VARCHAR(3)

#### A product description which is max 255 characters in length.

TEXT or CHAR(255)

#### A column to store an image.

VARBINARY(MAX)

#### A column to store price which is at most 5 digits before decimal and three digits after decimal.

FLOAT(5,3)

#### A date field to store date of birth of current students.

DATE

#### A field to store date of discovery of museum artifacts.

tinyint, tinyint, tinyint, boolean = (DD,MM,YYYY, AD/BC)

#### A field to store whether the product is active or not.

BOOLEAN

## Lab 6: Insert, Update and Delete Operations

### First find out the last value of identity field:

Select Max(SSN) From Professor



### Insert a new row in the professor table

Insert Into Professor Values ('Andrew Ng', 58, 'Professor', NULL, '843-4738594')



### Find out the SSN of newly inserted record:

1. Select Max(SSN) From Professor  
   OR
2. Select Scope\_Identity()  
   OR
3. Select @@Identity  
   OR
4. Select Ident\_Current(‘Professor’)



### Insert a new row by specifying the column list

Insert Into Professor (Name, Age, PhoneNumber, Rank)  
Values ('Robsert Sedgwick', 70, '803-4848594', 'Professor')



### Insert a new row by explicitly specifying a value for the identity column

Set Identity\_Insert Professor On  
Insert Into Professor (SSN, Name, Age, Rank, ResearchSpeciality, PhoneNumber)  
Values (1097, 'Isaac Newton', 45, 'Associate Professor', 'Gravity', '384-5938593')  
Set Identity\_Insert Professor Off



### Delete From Professor Where SSN > 1000



### Delete From Registration



### Truncate Table Registration

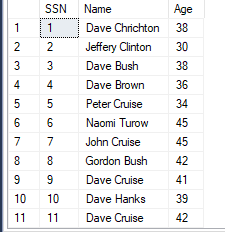


### Update Professor Set ResearchSpeciality = ‘Computational Intelligence’ Where SSN = 1097

# Agilosoft SQL Training – Session 2b - Labs

## Lab 7: Subqueries

### List down those students who are not part of any research projects (who are not research assistants in any of the projects):

Select SSN, Name, Age From Student  
Where SSN Not In (Select SSN From ResearchAssistant)  
  
Select SSN, Name, Age From Student  
Where SSN In (Select SSN From Student Except  
Select SSN From ResearchAssistant)  
  
  
Select SSN, Name, Age From Student Where SSN <> All  
(Select SSN From ResearchAssistant)

### Identify the oldest student

Select Top 1 SSN, Name, Age

From Student

Order By Age Desc

Select SSN, Name, Age

From Student

Where Age = (Select Max(Age) From Student)

Select SSN, Name, Age

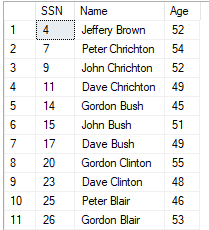
From Student

Where Age >=All (Select Age From Student)



### Identify the professors who are older than the average age of all professors

Select SSN, Name, Age  
From professor  
Where Age > (Select Avg(Age) From Professor)



### Identify professors who are involved in at least two projects (either as principle or co-investigator) and are less than 40 years old.

Select SSN, Name  
From Professor P  
Where ((Select Count(1) From Project Where PrincipalInvestigator = P.SSN)  
+ (Select Count(1) From CoInvestigator Where SSN = P.SSN)  
) >= 2

Select P.SSN, P.Name  
From Professor P Inner Join  
(Select PrincipalInvestigator As SSN From Project  
Union All  
Select SSN From CoInvestigator) Investigator On  
P.SSN = Investigator.SSN  
Group By P.SSN, P.Name  
Having Count(1) >= 2