



­

Hands-on Exercise:

Joins

Version: Understanding ANSI SQL

Table of Contents

[Session 7: Joins 3](#_Toc356566046)

[Exercise 1 3](#_Toc356566047)

[Exercise 2 3](#_Toc356566048)

[Exercise 3 3](#_Toc356566049)

[Exercise 4 4](#_Toc356566050)

[Exercise 5 4](#_Toc356566051)

Session 7: Joins

# Exercise 1

|  |  |
| --- | --- |
|  |  |
|  |  |
| Hands-on Exercise Objective |
| After completing the hands-on exercises, you will be able to:  Use different types of joins on tables. |
|  |  |

Problem Statement:

Write a query to display trainer\_id and batch\_id details in such a way that we get all possible combinations of trainer IDs and batch IDs.

**Hint:** Use Trainer\_Info and Batch\_Info tables.

Deliverables Expected:

All combinations of trainer\_id and batch\_id are displayed.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **select Trainer\_Id, Batch\_Id from Trainer\_Info**  **cross join Batch\_Info;** Exercise 2 | |  | | |
|  | |  |

Problem Statement:

Write a query to display the records from all columns of table’s associate\_status and batch\_info, wherever the batch\_id in the two tables matches.

Deliverables Expected:

Records are displayed based on the condition specified.

**select \* from Associate\_Status**

**inner join Batch\_Info on Associate\_Status.Batch\_Id = batch\_info.Batch\_Id;**

# 

# Exercise 3

Problem Statement:

Write a query to display the associate IDs of the associates tagged to trainers and all the trainer IDs irrespective of whether there are any associates tagged to them or not.

Hint: Use associate\_status and trainer\_info tables

Note: Use Right Outer Join

Deliverables Expected:

Records are displayed based on the condition specified.

**select Associate\_Status.Associate\_ID , Trainer\_Info.Trainer\_Id FROM Associate\_Status**

**Right OUTER JOIN Trainer\_Info on Trainer\_Info.Trainer\_Id = Associate\_Status.Trainer\_Id;**

# Exercise 4

Problem Statement:

Write a query to display the associate IDs of the associates tagged to trainers and all the trainer IDs irrespective of whether there are any associates tagged to them or not. Hint: Use associate\_status and trainer\_info tables.

Note: Use Left Outer Join

Deliverables Expected:

Records are displayed based on the condition specified.

**select Associate\_Status.Associate\_ID , Trainer\_Info.Trainer\_Id FROM Trainer\_Info**

**LEFT OUTER JOIN Associate\_Status on Trainer\_Info.Trainer\_Id = Associate\_Status.Trainer\_Id;**

# Exercise 5

|  |  |
| --- | --- |
|  |  |

Problem Statement:

Write a query to display the associate IDs of all the associates and trainer IDs of all trainers irrespective of whether any associate is mapped to a trainer ID and vice versa.

**Hint:** Use associate\_status and trainer\_info tables.

(NOTE: ANSI syntax for Full Outer Join is supported by SQL Server, Not Supported by MySQL, Oracle, and Sybase)

**Prerequisite:** Add an associate\_id which is not mapped to any trainer ID. Take care of the alteration in constraints to the other tables in order to achieve this.

Deliverables Expected:

Records are displayed based on the condition specified.

**SELECT a.associate\_id, t.trainer\_id from associate\_status a**

**left outer join trainer\_info t on a.trainer\_id=t.trainer\_id**

**union**

**SELECT a.associate\_id, t.trainer\_id from associate\_status a**

**right outer join trainer\_info t on a.trainer\_id=t.trainer\_id;**