Coding Standards

We can copy 10 style guides with their examples from this two links

[C# Style Guides](https://docs.microsoft.com/en-us/dotnet/csharp/fundamentals/coding-style/coding-conventions) and [Sql Style Guides](https://about.gitlab.com/handbook/business-technology/data-team/platform/sql-style-guide/v)

Team Members

Samuel Gerend

Abdiansir Muhumed

Bentley Epple

Ryan Schauer

C# Standards

1. Naming Classes and Structs: Using Pascal Case

Example: public class DataModel {}

1. Naming interfaces: using Pascal Case and putting I in-front of it.

Example: public interface IDataModel {}

1. Naming public members of types, such as fields, properties, events, methods,

and local functions: Using Pascal Case

Example: public bool IsCorrect; public void StartGame(){};

1. Naming private and internal fields: using camelCase with \_ in-front

Example: private int \_countNumber;

1. Naming method parameters: use camelCase

Example: public int Sum(int firstNumber, int secondNumber){}

1. Comments: start with an uppercase letter and finish with a period

Example: // This comment is very important.

1. Line comments: placed on new line and not on the end of a line of code

Example: public string myComment

     // Hello, this is my comment.

1. Loops: use explicit typing in foreach statements

Example: foreach (string word in phrase) {}

1. Implicit typed variables: use implicit typing when it is clear the variable type

Example - good: var myVar = “Software Engineering rules”;

Example - bad: var myVar = TeamFour.GetResults();

1. Spacing: place space after if / for / while / foreach, etc.

Example: if (teamFourIsCool) {}

SQL Standards

1. Naming Procedures must contain a verb

Example: GetEmployeeData

1. Include comments where it is necessary, using the syntax - - for single line comments

and /\* \*/ for multi-line comments

Example: - - Updating the view due to change in date

Example: /\* Selecting all of the records from

the Users table \*/

1. Always have keywords be all uppercase

Example: SELECT, FROM, WHERE

1. Names of tables, views, or procedures must begin with a capitalized letter

Example: Cities, Banks, Schools

1. Tables must have at least one key value

Example: Social Security Number (ssn) or ID (id) with a table of employees

1. Names of fields should begin with a lowercase letter and when longer than one word

should be connected with an underscore

Example – good: id, first\_name, title  
Example – bad: Id, firstName, Title

1. When adding in arguments, never put multiple arguments on one line.

Example – good:

SELECT \*

FROM Users

WHERE age > 20;

Example – bad:

SELECT

first\_name, phone\_number, email\_address

FROM Users;

1. Root keywords should always be on their own line, except if there is only one argument.

Example – good:

SELECT

first\_name,

last\_name

FROM Users;

Example – bad:

SELECT first\_name, last\_name

FROM Users;

1. Never add unnecessary square brackets around object names.

Example – good:

SELECT

id,

first\_name,

last\_name

FROM Users;

Example – bad:

SELECT

[id],

[first\_name],

[last\_name]

FROM [Users];

1. Never name columns the same as the table name to avoid confusion.

Example – good:

Table name: Orders

Column name: num\_orders

Example – bad:

Table name: Orders

Column name: orders