**Collin College**

**SAS® Classes Spring 2018**

**New for Spring 2018:**

* Students will have a choice of taking Continuing Education classes or a credit class
* Continuing Ed. students will have a choice of on-campus or web-delivered classes

**Instructor:**  [Marty Wertheim](http://www.linkedin.com/pub/marty-wertheim/99/71b/308)

[marty.the.sas.instructor@gmail.com](mailto:marty.the.sas.instructor@gmail.com)

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Course # | Title | Credit/CE | Start | End | Meeting times | Location |
| ITSW2370 | SAS Programming | Credit (3 hrs) | Jan 20 | May 12 | Sat 12-4 | SCC |
| 73504 | SAS Fundamentals 1 | CE | Jan 17 | Jan 31 | MW 6:30-9:30 | CYC |
| 75330 | SAS Fundamentals 1 | CE | Mar 7 | Mar 21 | MW 1:00-4:00 | web |
| 73560 | SAS Fundamentals 2 | CE | Feb 7 | Feb 21 | MW 6:30-9:30 | CYC |
| 75331 | SAS Fundamentals 2 | CE | Mar 28 | Apr 11 | MW 1:00-4:00 | web |
| 75332 | SAS PROC SQL | CE | Apr 18 | May 2 | MW 1:00-4:00 | web |

Note: SCC = Spring Creek Campus - 2800 E Spring Creek Pkwy, Plano, TX 75074

CYC = Courtyard Campus - 4800 Preston Park Blvd, Plano, TX 75093

**Registration / cost:**

* Continuing Ed - Call 972-985-3711. Each course is $239 regardless of residency
* Credit - <http://www.collin.edu/gettingstarted/admissions/>

<http://www.collin.edu/gettingstarted/bursar/tuition.aspx>

**SAS Programming Fundamentals 1 (Continuing Ed)**

This course is for anyone who wants to learn how to write SAS programs. Along with *SAS Programming Fundamentals 2,* it is the entry point to learning SAS programming, and can help you prepare for the[*SAS Base Programmer Exam for SAS 9*](http://support.sas.com/certify/creds/bp.html) certification exam.

Upon completion, you will be able to

* Understand , create, and use SAS datasets and libraries
* Read existing SAS datasets and Excel spreadsheets
* Create new variables, and use numeric and character functions
* Use IF and WHERE statements to implement sub-setting and conditional processing
* Sort data and produce detail and summary reports (SORT, PRINT, MEANS, FREQ)
* Produce charts and graphs
* Export data into CSV and Excel files

**SAS Programming Fundamentals 2 (Continuing Ed)**

This course is a continuation of *SAS Programming Fundamentals 1*.

Upon completion, you will be able to

* Write moderately complex SAS programs
* Read raw data and CSV files into SAS
* Work with date and time values
* Use DO loops and arrays
* Combine SAS datasets using SET and MERGE
* Use additional functions not covered previously
* Create and use user formats
* Explain compilation vs. execution and the significance of the Program Data Vector

**SAS Programming (credit)**

Covers all the topics in Fundamentals 1 and 2 plus these additional topics

* Using SQL in SAS (PROC SQL)
* Data validation and cleaning – principles and techniques
* Creating tabular reports (PROC TABULATE)
* Term project using public data

**SAS Intermediate Programming – PROC SQL (Continuing Ed)**

This course is for SAS programmers who want to expand their programming skills with the use of the SQL procedure (PROC SQL).

Upon completion, you will be able to

* query and subset data from SAS data sets and DBMS relational tables
* summarize and present data
* use SAS extensions to the base SQL language
* use joins, set operators, inline views and sub-queries
* create and modify tables and views
* replace multiple DATA and PROC steps with one SQL query

**SAS Intermediate Programming – the Macro Language (continuing Ed)**

**Not offered Spring 2018. Will return Summer 2018.**

This course is for SAS programmers who want to expand their programming skills with the use of the macro facility.

Upon completion, you will be able to

* Define and use user macro variables
* Define and call a macro program with parameters
* Describe what the macro processor does and how it interfaces with the compiler
* Write macro programs that use conditional and iterative processing
* Create macro variables in a data step or PROC SQL
* Use macro language functions
* Understand global vs. local variables
* Use %INCLUDE and AUTOCALL facilities
* Use indirect referencing (double ampersand)