



CURRICULUM

Data Warehousing Concepts

- ☞ What is a Data Warehouse ?
- ☞ What is a Data Mart ?
- ☞ What is the difference between Relational Databases and the Data in Data Warehouse (OLTP versus OLAP)
- ☞ Why do we need Data Warehouses when the Relational Source Database exists
- ☞ Multi Dimensional Analysis and Decision Support Reporting from Data Warehouse
- ☞ Data Warehouse Architecture (ETL Design)
- ☞ Normalized Relational Database Design (Entity Relationship Model)
- ☞ Dimensional Data Modeling
 1. Star Schema Design
 2. Snowflake Schema Design
- ☞ Slowly Changing Dimensions
- ☞ Why SAS BI ? Capabilities of SAS BI 2 Advantages of SAS BI Over Base & Advance SAS
 1. BI Architecture
 2. SAS BI Tools

BASE SAS

» Introduction

- ☞ An Overview of the SAS System
- ☞ SAS Tasks
- ☞ Output produced by the SAS System
- ☞ SAS Tools (SAS Program - Data step and Proc step)
- ☞ A sample SAS program
- ☞ Exploring SAS Windowing Environment Navigation

» Data Access & Data Management

- ☞ SAS Data Libraries
- ☞ Rules for Writing SAS Programs / Statements, Dataset & Variable Name
- ☞ Getting familiar with SAS Dataset
 1. Data portion of the SAS Dataset
 2. Rules for writing Dataset names / Variable names
 3. Attributes of a Variable (Numeric / Character)
 4. Options
 - a.) System Options (nodate, linesize, pagesize, pageno etc)
 - b.) Dataset Options (Drop, Keep, Rename, Where, Firstobs= Obs=)
- ☞ How SAS works (Flow of Data Step Processing - Compilation & Execution phase)
 1. Input Buffer
 2. Program data vector (PDV)
 3. Descriptor Information of a SAS Dataset

» Datalines or cards

» Data Transformations

- ☞ SAS Date Values
- ☞ Length Statement
- ☞ Creating multiple output SAS datasets from single input SAS dataset
- ☞ Conditionally writing observations to one or more SAS datasets
- ☞ Outputting Multiple Observations (Implicit Output)
- ☞ Selecting Variables and observations (DROP or KEEP statement and DROP= or KEEP = dataset options)
- ☞ Controlling which Observations are read (OBS= FIRSTOBS = Options)
- ☞ The Data Statement _Null_
- ☞ The _N_ Automatic Variable
- ☞ Creating Subset of observations
 1. Conditional Processing using IF-THEN and ELSE statement, IF----
THEN DO ; ----END;ELSE DO;----END;
 2. DO WHILE Statement
 3. DO UNTIL Statement
 4. Iterative DO loop Processing
 5. Where Statement OR Where Condition (dataset)
 6. Deciding whether to use a Where statement or Subsetting IF statement
- ☞ Accumulating Totals for a Group of Data (BY- Group Processing (First & Last))
- ☞ Multiple BY variables
- ☞ DATASETS Procedure (To modify the Variable name/lable/format/informat)
- ☞ Reading SAS datasets and Creating Variables
- ☞ Creating an Accumulating Variable (The RETAIN Statement)
- ☞ The DELETE Statement
- ☞ The SUM Statement
- ☞ The RENAME = Data Set option
 - ☞ Combining SAS Datasets
 1. Concatenating SAS Data Sets Using SET statement in DATA Step
 2. Inter Leaving SAS Data Sets
 3. Merging SAS Data Sets
 - a.) Match-Merge
 - b.) Using Merge Statement
 - c.) THE IN = Data Set option
 - d.) Additional Features of merging SAS Datasets
 - * One-to-Many Merging
 - * Many-to-Many Merging

» Reading Raw Data From External File (INFILE & INPUT Statement)

- ☞ Introduction to Raw Data
- ☞ Factors considered to examine the raw data
- ☞ Reading Unaligned Data (List Input)
- ☞ Reading Data Aligned to Columns (Column Input)
- ☞ Reading Data that requires Special Instructions (Formatted Input)
- ☞ Controlling the position of the Pointer in Formatted Input

1. Absolute - Column pointer control (@)
 2. Relative- Column pointer control (+)
- ☞ Mixed Style Input (Mixing List, Input. Formatted Input styles in one INPUT Statement)
 - ☞ Using colon (:) modifier to specify an informat in the INPUT Statement)
 - ☞ Recognize delimiter in the raw data file (Using DLM= option in INFILE Statement)
 - ☞ Missing data at the end of row (Using MISSOVER option in INFILE statement)
 - ☞ Missing values without placeholders (DSD option in INFILE statement)
 - ☞ Reading a raw data file with multiple records per observation (Column pointer controls)
 1. Method1: Using Multiple INPUT statement
 2. Method2: Using Line Pointer Control (/)
 3. Reading Variables from multiple records in any order (#n)
 - ☞ Line Hold Specifies in INPUT statement
 1. The Single Trailing @
 2. The Double Trailing @@ (Multiple Observations per Record)
 - ☞ Methods of Control in INFILE statement
 1. FLOWOVER
 2. STOPOVER
 3. MISSOVER
 4. TRUNCOVER
 - ☞ Writing to an External File (FILE & PUT Statement)
 - ☞ Reading Excel Spreadsheets (IMPORT Wizard / Import Procedure)

» SAS FUNCTIONS

- ☞ Manipulating Character Values (SUBSTRING / RIGHT / LEFT / SCAN/ CONCATENATION)
- ☞ TRIM / FIND / INDEX / UPCASE / LOWCASE / COMPRESS / LENGTH)
- ☞ Manipulation Numeric Values (ROUND / CEIL / FLOOR / INT / SUM / MEAN / MIN / MAX)
- ☞ Manipulating Numeric Values based on DATES (MDY / TODAY / INTCK / YRDIF)
- ☞ Converting Variable Type
 1. INPUT (character-to-numeric)
 2. PUT (numeric-to-character)
- ☞ Debugging SAS program (DEBUG Option)
- ☞ SAS VARIABLE Lists
- ☞ SAS Arrays
- ☞ Enhancing Report Output
 1. Defining Titles & Footnotes
 2. Formatting Data values (Date, Character & Numeric values)
 3. Creating User-Defined Formats (Proc Format)
 4. Formats & Informats

» Analysis & Presentation

- ☞ Descriptor portion of the SAS Data Set (Proc Contents)
- ☞ Producing List Reports (Proc Print)
- ☞ Sequencing and Grouping Observations (Proc Sort)
- ☞ Producing Summary Reports

1. PROC FREQ -(One Way & Two-Way Frequencies)
 2. PROC MEANS
 3. PROC REPORT
 4. PROC TABULATE
 5. PROC SUMMARY
 6. PROC PRINTO
 7. PROC APPEND
 8. PROC TRANSPOSE
 9. PROC COPY
 10. PROC COMPARE
 11. PROC DATASETS
 12. Regression Procedure
 13. Analysis-of-Variance Procedures
 14. Univariate / Multivariate Procedures
 15. Ranking Procedure
- Producing Bard and Pie Charts
 - Producing Plots
 - The Output Delivery System (SAS/ODS)
 1. Creating HTML Reports
 2. Creating Text Reports
 3. Creating PDF Reports
 4. Creating CSV Files

SAS Macro Language

Introduction to the Macro Facility

» Purpose of the Macro Facility

- Generate SAS code using Macros (%Macro & %Mend)
 1. Tips on Writing Macro-Based Programs
- Replacing Text Strings using Macros Variables (%Let)

» MACRO PROGRAMS

- Defining a Macro (%Macro & %Mend)
- Macro Compilation
- Monitoring Macro Compilation (MCOMPILENOTE OPTION)
- Calling a Macro (%Macro-Name)
- Macro Execution
- Monitoring Macro Execution (MLOGIC OPTION)
- Viewing the generate SAS Code in the Log from Macro Program (MPRINT OPTION)
- Macro Storage
- Macro Parameters
 1. Macro Parameters Lists
 2. Macros with Positional Parameters
 3. Macros with Keyword Parameters

- Arithmetic and logical Operations
- Conditional Processing
 1. % IF expression % THEN text ; %ELSE %TEXT;
 2. % IF expression % THEN %DO; %END; %ELSE; %DO;
- Stored Compiled Macros
- %INCLUDE Statement

» Macro Processing

- Tokens
- Macro Triggers
- How the Macroprocessor works

» Macro Variables Concepts

- Referencing a Macro Variable
- Displaying Macro Variable Value in the SAS log (SYMBOLGEN OPTION)
- Automatic Macro Variables
 1. System-Defined Macro Variables (_AUTOMATIC_)
 2. User-Defined Macro Variable (_USER_)
- %LET Statement
- Global Macro variables
- Local Macro Variables
- Deleting User-Defined Macro Variable (%SYMDEL)
- Macro Functions
 1. Character Strings
 2. Other SAS Functions
 - a.) %SYSFUNC
 - b.) %STR
- Combining Macro Variable References with Text
- Macro Variable Name Delimiter
- Quoting
- Creating Macro Variables in the Data Step (CALL SYMPUT ROUTINE)
- Obtaining Variable value during Macro Execution (SYMGET FUNCTION)
- Creating Macro Variables during PROC SQL Execution (INTO Clause)
 1. Creating a delimited list of Values.

SAS SQL PROCESSING

» Introduction to the SQL Procedure

- Terminology
- Features of PROC SQL
- PROC SQL Syntax (SELECT, FROM, WHERE, GROUP BY, HAVING, ORDER BY)
 1. VALIDATE Keyword
 2. NOEXEC Option
 3. Added PROC SQL Statements (ALTER, CREATE, DELETE, DESCRIBE, DROP)
 4. FEEDBACK OPTION
- PROC SQL and DATA Step Comparisons

- Queries
 1. Retrieving Data from a table
 2. Identify All Rows in a Table
 3. Remove Duplicate Rows
 4. Sub setting using WHERE clause
 5. Sub setting with Calculated Values
 6. Ordering Data
 7. Enhancing Query Output (LABEL, FORMAT)
 8. Grouping Data (Group By)
 9. Analyzing Groups of data (COUNT)
 10. Updating Data values (Update Statement)
 11. Using Table Alias
 12. Creating Views
 13. Creating Dropping Indexes
 14. Sub Queries
 - a.) Non-Correlated Sub Query
 - b.) Correlated Sub Query
- Combining Tables
 1. Joins
 - a.) Inner Joins
 - b.) Outer Joins
 - * Left Join
 - * Right Join
 - * Full Join
- Set Operators
 1. EXCEPT
 2. INTERSECT
 3. UNION
- Choosing between Data Step Merges and SQL Joins

UNIX

» Introduction to Unix

- Introduction to UNIX Architecture
- Understanding UNIX Commands
- Understanding ID/Groups/Permissions
- Introduction to Shell Scripting
- Writing UNIX Programs
- Understanding VI Editor
- Introduction to LSF
- Scheduling SAS Codes through UNIX

Partners :



Java



development | consultancy | training

E-mail: info@ducatindia.com
Visit us: www.ducatinidia.com
www.facebook.com/ducateducation

NOIDA

A-43 & A-52, Sector-16,
Noida - 201301, (U.P.) INDIA
☎ 70-70-90-50-90
☎/☎ +91 99-9999-3213

GURGAON

1808/2, 2nd floor old DLF,
Near Honda Showroom,
Sec.-14, Gurgaon (Haryana)
☎ 70-70-90-50-90

GHAZIABAD

1, Anand Industrial Estate,
Near ITS College, Mohan Nagar,
Ghaziabad (U.P.)
☎ 70-70-90-50-90

PITAMPURA (DELHI)

Plot No. 366, 2nd Floor,
Kohat Enclave, Pitampura,
(Near- Kohat Metro Station)
Above Allahabad Bank,
New Delhi- 110034.
☎ 70-70-90-50-90

SOUTH EXTENSION (DELHI)

D-27, South Extension-1
New Delhi-110049
☎ 70-70-90-50-90
☎ +91 98-1161-2707