App Engine
Course Checklist

☑ Task
Day 1: Technology Overview and Development Approach

Technology Overview
☐ Database Driven, Batch Programs, Set-Based
☐ Structure of AE (Sections, Steps, Actions)

Working with AE Programs in App Designer
☐ Viewing program structure
☐ Adding your AE Program to your Project
☐ Migrating AE Programs
☐ Finding code in existing programs

AE Action Types
☐ Do Actions: While, When, Select, Until
☐ Understanding Action Execution Order
☐ SQL
☐ PeopleCode
☐ Log Message
☐ Call Section

☑ Example 1: Creating an AE Shell

State (AET) Records
☐ Work Record throughout execution
☐ Retrieving and Storing Values in State Records

Creating Process Run Pages
☐ Process Run Pages Overview
☐ Designing Run Control Records (OPRID, RUN_CNTL_ID)
☐ Creating Run Control Pages (PRCSRUNCNTL_SBP)
☐ Building the Component (PRCSRUNCNTL Search Record)
☐ Connecting a Process to a Component (Process Definition)

Day 2: Building Set-Based AE Programs

Temporary Tables and AE
☐ Understanding Temp Table Architecture: How TAO Table instances are Used for isolating set Processing
☐ Structure of TAO Tables (PROCESS_INSTANCE as key)

☐ Table Instances and Where They Are Set (Temp Table tab for each AE Program), online instance count in PeopleTools Options
☐ Online vs. Batch Instances
☐ Referring to table instances [ %Table() ]

☐ Temporary Table Usage (PeopleTools > Application Engine > Review Temp Table Usage)

Inactivating Steps
☐ How to mark steps inactive
☐ Finding all steps marked inactive in a program (SQL)

Writing SQL for AE (Using Meta-SQL)
☐ Used to interact with state variables
☐ Simplifies SQL Coding Substantially
App Engine
Course Checklist

☑️ Task

- Y ☐ %InsertSelect: To insert into one table by selecting from another
- Y ☐ %Select: To take results and place into state variables

Writing PeopleCode for AE
- Y ☐ Take care to avoid row-level PeopleCode
- Y ☐ How to access state values in PeopleCode
- Y ☐ Affecting AE Program Execution with PeopleCode (Exit(0))

Day 3: Advanced Topics and Troubleshooting

- ☐ Running an AE from the Command-Line
  Building Restartable AE Programs
  - Y ☐ Why Restartability is Necessary
  - Y ☐ How Restart Works (Commits and PS_AERUNCONTROL)
  - Y ☐ Forcing Programs to Start from Beginning
  Dynamic Section Calls
  - Y ☐ Using state variables to store the section name
  - Y ☐ Other ways of conditionally calling sections (Do When)

- Y ☐ Best when there are many potential logic paths (Sections to call)
  Using the App Engine Request Page
  - ☐ PeopleTools > Application Engine > Request AE
  - ☐ Specifying Bind Variables through Run Page
  Starting AE From PeopleCode

- Y ☐ Initiating App Engine Programs from PeopleCode (CallAppEngine)
  Using AE Libraries
  - Y ☐ Similar to the App Class or FUNCLIBs
  Dynamic SQL in PeopleCode
  - Y ☐ Pulling in text from a SQL object using %SQL(SQLID)

- Y ☐ Putting in a where condition via %BIND(statevar,NOQUOTES)
  TroubleShooting AE
  - Y ☐ Finding within all code of an AE
  - Y ☐ ToolsTable Overview
  - Y ☐ Finding other programs called by this AE (tools SQL)
  - Y ☐ Finding inactive steps
  - Y ☐ Techniques to Handling Abends
  - Y ☐ Reviewing Trace timings

  Tracing in AE Programs
  - Y ☐ Turning on Tracing (Process Definition)
  - Y ☐ Setting Trace Flags (-Trace, -ToolsTraceSQL, -ToolsTracePC)
  - Y ☐ Reviewing the Trace File
  - Y ☐ AET vs. Trace SQL files