



Generelle Tipps

Hier erfahren Sie, was wir beim Programmieren hinsichtlich der Lesbarkeit und Wartbarkeit für wichtig halten.

Einleitung

Die folgenden Tipps sollen Ihnen und Ihren Kollegen die Fehlersuche und das Verständnis für ein Programm leichter machen!

Die Hinweise habe ich nicht um ihrer selbst Willen gesammelt oder damit der Quellcode „schön“ aussieht, sondern weil dies Hilfen sind, die ich in jahrelanger Erfahrung gesammelt und genutzt habe.

Jeder hat einen eigenen Programmierstil und mit ein paar Konventionen machen Sie es sich und anderen, die Ihre Programme nach Ihnen lesen, einfacher.

Probieren Sie es aus...

Namensgebung von Variablen

Falls nötig, ist es sinnvoll, mit einem Präfix kenntlich zu machen, ob die Variable lokal in einer Routine definiert wird oder global zugänglich ist. Sie könnten z. B. L_ für **lokale Variablen** und G_ für **globale Variablen** verwenden.

Achten Sie darauf, dass Variablen einen Präfix haben, der auf den **Typ** schließen lässt. Benennen Sie also **einfache Variablen** mit V_, **Strukturen** mit X_ und **Tabellen** mit T_.

In der Kombination könnten Sie dann LT_ITAB für eine lokale Tabelle verwenden und GX_STRUC für eine globale Struktur.

Benennen Sie **PARAMETERS**-Eingabefelder mit P_

Benennen Sie **SELECT-OPTIONS**-Tabellen mit S_

Verwenden Sie den Präfix F_ für **Übergabeparameter in Formroutinen**, bzw. FI_ für Using-Parameter, FE_ oder FC_ für Changing-Parameter und FT_ für Tabellen.

Benennen Sie in **Funktionsbausteinschnittstellen** die Eingabeparameter mit I_ und die Ausgabewerte mit E_, bzw. Tabellen mit IT_ oder ET_.

Sie werden sehen, dass Programme, die nach einer solchen Konvention programmiert sind, wesentlich einfacher zu verstehen sind.

Unnötiger Quellcode

Wenn Sie ein Programm kopieren, prüfen Sie, welchen Quellcode Sie wirklich benötigen! Nicht benötigte Überbleibsel aus dem Originalprogramm verwirren bei nachträglichen Änderungen!

Wenn Sie den Returncode von Funktionsbausteinaufrufen nicht abfragen, bzw. nicht gesondert auswerten, dann können Sie die **EXCEPTIONS** auch auf OTHERS beschränken und die anderen Exceptions löschen!

Beim Pretty Printer werden FORM-Routinen automatisch **Kommentarzeilen** vorangestellt, in denen die Übergabeparameter beschrieben werden sollen. Wenn Sie das nicht tun, dann löschen Sie diese Zeilen! Sie sind unnötiger Ballast.

Wenn das Programm läuft, dann können auch Kommentarzeilen gelöscht werden, die Sie vorher noch stehengelassen haben, weil Sie sich nicht sicher waren, ob Sie das Coding evtl. noch benötigen. Wenn Sie sicher sind, dass das Coding nichts taugt, dann weg damit!

Dokumentation

Dokumentation ist ein leidiges Thema. Das liegt vor allen Dingen daran, dass Programmstrukturen schlecht verbal wiederzugeben sind.

Wenn Sie jedoch nach einer Vorgabe programmieren und wissen, was eine Funktion machen soll, dann schreiben Sie es kurz dazu. Nach drei Monaten weiss man nämlich nicht unbedingt mehr, warum das Feld TRGHU aus der Tabelle T123I nachgelesen wird!

Pretty Printer

Verwenden Sie den Pretty Printer! Er macht das Lesen des Codings wesentlich einfacher!

Erweiterte Programmprüfung

Über den Menüpunkt *Prüfen - Erw. Programmprüfung* können Sie eine umfangreichere Prüfung für das Programm durchführen lassen, als die normale Syntax-Prüfung.

Hier werden noch andere wichtige Schwachpunkte aufgedeckt, wie z. B.

- Nicht verwendete Variablen
- Fehlerhafte CALL-FUNCTION-Schnittstellen
- Zeichensatzkompatibilität

Nutzen Sie diese Hilfe!