



Forschungsdatenzentrum Gesundheit

Nebenbestimmungen - Anlage 2: Programmcoderegeln

Version 01-10-000

Inhalt

1	Allgemeine Programmcoderegeln	2
2	Spezielle Programmcoderegeln	4
2.1	Für SQL.....	4
2.2	Für den Datenbankzugriff (HANA).....	5
2.3	Für R.....	5
2.4	Für Python	5
3	Programmcoderegeln mit speziellem Fokus auf den Identifikationsschutz	6
4	Regeln für Variablen und Tabellen.....	7
4.1	Regeln zur Benennung von Variablen	7
4.2	Regeln zur Benennung von Tabellen	7
4.3	Regeln für erzeugte Tabellenstrukturen.....	7
4.3.1	Regeln zum Umfang: Maximale Tabellen- und Spaltenanzahl.....	7
4.3.2	Regeln zu erlaubten Datentypen und Feldlängen.....	8
4.3.3	Erforderliche Tabellen zur Beschreibung der Ergebnismenge	9
5	Hinweise.....	10
5.1	Regeln zur automatischen Ausführung von Skripten	10
5.2	Speicherung von Skripten und Ergebnissen	10
5.3	Ergänzende Hinweise zum Arbeiten mit R und Python.....	10
5.3.1	Vorgelagerte Datenaufbereitung per SQL-Skript	10
	Anhang	11

Bitte haben Sie Verständnis dafür, dass der Aufbau Ihres Programmcodes einigen formalen Vorgaben zu genügen hat. Sie erleichtern den Mitarbeiterinnen und Mitarbeitern des Forschungsdatenzentrums damit im Bedarfsfall die Einarbeitung in Ihren Programmcodes und können die zur Prüfung notwendige Zeitspanne erheblich verkürzen. Ihr Programmcodes muss es den Mitarbeiterinnen und Mitarbeitern ermöglichen, Ziel und Inhalt Ihrer Analysen schnell zu erfassen und zu beurteilen. Mit diesem Dokument geben wir Ihnen die notwendigen Konventionen an die Hand. Berücksichtigen Sie in Ihren Analysen, dass im FDZ-System sehr große Datenmengen vorgehalten und verarbeitet werden. Die Anzahl der Datensätze einer einzelnen Tabelle für ein Berichtsjahr umfasst, je nach Themengebiet, mehrere Millionen bis Milliarden Datenzeilen. Um derartig große Datenmengen effizient im Mehrbenutzerbetrieb zu verarbeiten, ist die Einhaltung folgender Vorgaben notwendig.

1 Allgemeine Programmcoderegeln

Machen Sie sich mit unserer Benennungskonvention (s. folgende Abschnitte) und Regeln vertraut und beachten Sie diese bei der Ausarbeitung Ihres Skripts.

- Halten Sie den Code der eingereichten Skripte möglichst kurz und einfach um den möglichen Prüfaufwand zu reduzieren.
- Verwenden Sie bitte durchgängig nur die Sprache Deutsch oder Englisch.
- Vergeben Sie sprechende, inhaltsbezogene Namen für Variablen und Tabellen.
- Machen Sie im Skript folgende Angaben im Programmkopf:
 - Nutzungsberechtigte Institution
 - Projekttitel
 - Name des Forschungsprojekts (Projektkürzel)
 - Aktenzeichen
 - Selektionskriterien zu Gruppen oder Untergruppen. Bitte im Programmkopf oder an geeigneter Stelle im Skript einfügen.
- Beschreiben Sie den Programmablauf möglichst systematisch. Erläutern Sie dabei ggf. auch Bezüge zu vorhergehenden und zukünftigen Auswertungen.
- Erläutern Sie die Bedeutung aller neu erzeugten Tabellen und inhaltlich relevante Variablen. Nutzen Sie hierfür die Kommentarfunktion und vergeben Sie sprechende Namen.
- Erläutern Sie auch den Bezug zu den Tabellen und Variablen der FDZ-Daten und Kataloge.
- Unterteilen Sie Ihr Programm entsprechend dem Programmablauf in verschiedene, nummerierte Abschnitte mit aussagekräftigen Titeln. Achten Sie dabei darauf den Programmcodes optisch klar zu strukturieren.
- Verwenden Sie übliche einheitliche Zeileneinrückungen für Codeblöcke.
- Erläutern Sie für jeden Abschnitt dessen Zweck im Kontext des von Ihnen beschriebenen Programmablaufs.
- Erläutern Sie ggf. auch einzelne Anweisungen und deren Komponenten.
- Achten Sie aber auch darauf, dass durch Kommentare und Erläuterungen kein erhöhtes Re-Identifikationsrisiko entsteht (z. B. ein Kommentar: „Nur 2 Personen über 90 Jahre, daher Vergrößerung“).
- Bitte verwenden Sie maximal 132 Zeichen pro Codezeile.
- Das Maximum an SQL-Statements liegt bei 10.000.

Forschungsdatenzentrum Gesundheit

- Rücken Sie Schleifen bitte einheitlich ein und fügen Sie Abstände zwischen den Programmblöcken ein.
- Die Groß- und Kleinschreibung sollte einheitlich verwendet werden.
- Bitte wechseln Sie nicht die Kennzeichnungen für im Code benutzte Objekte (Beispiel: Vergleichsoperatoren entweder einheitlich „<>“ oder „!=“).
- Bitte berücksichtigen Sie, dass Ausgaben (OUTPUTS), die im Analyseraum z. B. auf der Konsole erfolgen, nicht an Sie übermittelt werden.
- Bitte berücksichtigen Sie, dass in Ihrem Skript keine Ergebnisse aus FDZ-Daten in Form von Ausgaben (OUTPUTS) enthalten sein dürfen, insbesondere nicht in Form von Kommentaren.
- Nutzen Sie für Ihre Ergebnisse temporäre Tabellen (Präfix TT_) oder Ergebnistabellen (Präfix RT_).
- Erstellen Sie zwei Data Description Tabellen zur inhaltlichen Beschreibung Ihrer Ergebnistabellen mit dem Namen DD_RESULT_TABLES und DD_RESULT_COLS. Beispielskripte dazu finden Sie im Unterverzeichnis „Hilfen“ im GIT-Verzeichnis zu Ihrem Antrag.
- Erstellen Sie keine Kopien der Originaldatentabellen. Dies kostet unnötig Speicherplatz, reduziert die Ausführungsgeschwindigkeit Ihres Skriptes. Wo immer möglich, arbeiten Sie bitte mit frühzeitiger Datenreduktion über Filterbedingungen.
- Vermeiden Sie bei der Skripterstellung tiefe Schachtelung von Verzweigungen und / oder Schleifen sowie komplizierte Bedingungsabfragen.
- Vermeiden Sie die Verkettung mehrerer Unterabfragen. Erzeugen Sie stattdessen mehrere temporäre Zwischentabellen oder Vektoren, die Sie anschließend geeignet zusammenführen.
- Achten Sie darauf, dass Abfragen und Unterabfragen nicht wiederholt erstellt werden. Statt diese wiederholt zu erstellen, sollten Sie eine temporäre Tabelle mit dem Ergebnis dieser Abfrage erzeugen und diese dann, wo immer erforderlich, einbinden.
- Bitte setzen Sie Pfadreferenzen auf verwendete Dateien / Skripte an den Programmanfang. Verwenden Sie nur relative Pfadangaben.
- Achten Sie auf ein angemessenes Laufzeitverhalten und Fehlerfreiheit. Testen Sie dazu Ihren Code syntaktisch mit Hilfe der Testdaten.
- Die Verwendung von zusätzlichen als den durch das FDZ bereitgestellten Bibliotheken, insbesondere Bibliotheken von Drittanbietern, ist untersagt. Wenn Sie weitere als die angebotenen Pakete benötigen, wenden Sie sich an das FDZ um die Möglichkeit einer Bereitstellung zu klären. Pakete werden in periodischen Abständen nach Prüfung in das System aufgenommen.
- Das Filtern auf Pseudonyme innerhalb des Skripts im Analyseraum ist nicht erlaubt und auch nicht sinnvoll. Dies führt zu unerwünschten Fehlern, da die Pseudonyme auf den Testdaten im Analyseraum antragsspezifisch bereitgestellt werden und beim Lauf auf den Originaldaten anders benannt sind. Das Filtern von Subgruppen sollte daher immer über andere merkmalsbezogene Felder in einer Tabelle erfolgen und nicht über ein gezieltes Filtern von Pseudonymen.

2 Spezielle Programmcoderegeln

2.1 Für SQL

- Sorgen Sie dafür, dass Ihr Programm in einem Schritt in einer Datei ausführbar ist. Es ist nicht möglich weitere Skriptdateien aus dem Hauptprogramm in SQL aufzurufen. Sie können bis zu 10.000 Statements im SAP HANA Database Explorer ausführen.
- Verwenden Sie SAP HANA kompatible SQL-Syntax. Nutzen Sie dazu auch das [Referenzhandbuch](#), das von SAP veröffentlicht wird. Unterschiede sind unter anderem:
 - SAP HANA SQL behandelt Tabellen- und Spaltennamen standardmäßig als Großbuchstaben, sofern sie nicht in doppelte Anführungszeichen ("") gesetzt sind.
 - Ohne Anführungszeichen: automatisch in Großbuchstaben interpretiert
 - Mit Anführungszeichen: exakt so interpretiert, wie geschrieben (case-sensitive)
 - Empfehlung:
 - Keine Anführungszeichen bei Tabellen- und Spaltennamen verwenden
 - In Großbuchstaben schreiben, was dem HANA-Standard entspricht
 - Aliase ohne Anführungszeichen verwenden
 - Automatische Typkonvertierungen sind in SAP HANA SQL nicht oder nur eingeschränkt möglich. Bei Vergleichen zwischen NVARCHAR und numerischen Datentypen immer `CAST()` verwenden, um Laufzeitfehler oder falsche Ergebnisse zu vermeiden.
 - Nullverhalten und leere Strings
 - Ein leerer String („“) ist ungleich NULL; in CHAR-Feldern wird er als Leerzeichen gespeichert, in VARCHAR-Feldern als Zeichenkette der Länge 0
 - NULL steht für „kein Wert“, „unbekannt“ oder „missing“
 - IS NULL oder IS NOT NULL verwenden
- Setzen Sie das Schema am Anfang des Skripts mit dem Befehl „`SET SCHEMA <Aktenzeichen>;`“ (z.B. `SET SCHEMA P31851_123;`). Achten Sie auf den Unterstrich beim Aktenzeichen.
- Verzichten Sie auf „`SELECT *`“ Abfragen. Benennen Sie in einem `SELECT`-Statement die einzubeziehenden Spalten, damit diese erkennbar werden und nur die notwendigen Spalten selektiert werden.
- Verwenden Sie keine DML-Befehle wie `INSERT`, `UPDATE`, `DELETE` und `MERGE`, da diese auf großen Datenmengen lange Laufzeiten und einen hohen Ressourcenverbrauch verursachen. Benutzen Sie stattdessen „`CREATE TABLE AS (SELECT)`“ und „`UNION ALL`“.
- Vermeiden Sie die Erstellung von Tabellen per `CREATE TABLE`, die nachträglich über DML-Befehle (z. B. `DELETE FROM` oder `ALTER TABLE`) minimiert werden, da dies sehr ressourcenintensiv ist und unnötiges Change-Logging auslöst.
- Filtern Sie die Daten frühzeitig über `CREATE-TABLE` Statements mit `WHERE`-Bedingungen, da sich dies positiv auf die Ausführungsgeschwindigkeit Ihres Skriptes auswirkt.
- Vermeiden Sie wenn möglich den Einsatz von Cursor-Syntax, um die Performance nicht zu beeinträchtigen.
- Vermeiden Sie implizite Typumwandlungen bzw. verwenden Sie explizite Typumwandlungen, um Datenmaterialisation zu vermeiden. Datenmaterialisation führt zu längeren Ausführungszeiten, Speicherplatz und CPU-Verbrauch.
- Aus Sicherheitsgründen ist die Benutzung von einigen Befehlen bei der Ausführung auf den Volldatensatz nicht gestattet. Während der Skriptentwicklung im Analyseraum kann man diese Befehle nutzen, aber eine erfolgreiche Lauffähigkeitsprüfung ist nicht möglich. Auch bei auskommentierten Befehlen schlägt die Prüfung fehl. **Wichtig!** Dazu gehören auch die meisten `DROP`-Befehle, inklusive `DROP TABLE` und `DROP VIEW`. Eine Liste mit den nicht-ausführbaren SQL-Befehlen befindet sich im [Anhang](#) dieses Dokuments.

2.2 Für den Datenbankzugriff (HANA)

Beachten Sie, dass beim Arbeiten mit R und Python durch spezielle Funktionsaufrufe ebenfalls Zugriffe auf die Datenbank erfolgen.

- Benutzen Sie für das Ansprechen der von Ihnen erzeugten Tabellen keine Schema-Objekt Präfixe wie „mein_schema.meine_tabelle“. Setzen Sie das Schema am Anfang des Skripts mit dem Befehl „*SET SCHEMA <Aktenzeichen>*“ (z. B. *SET SCHEMA P31851_123*).
- Fügen Sie einzelne Ergebnistabellen nur dann zu einer Ergebnistabelle zusammen, wenn sich dadurch der Tabelleninhalt und der im Grunde gegebene Bezug zwischen den einzelnen Ergebnistabellen leichter erschließen lässt und ein inhaltlicher Zusammenhang besteht.
- Vermeiden Sie das Zusammenfügen von sehr großen FDZ-Rohdaten-Tabellen zu temporären Zwischentabellen. Verwenden Sie besser keine temporären Tabellen die mittels „CREATE LOCAL TEMPORARY TABLE“ erstellt werden. Bei ungefilterten und großen Tabellen führt dies zu Skriptabbrüchen und Fehlermeldungen. Falls es an einer Stelle nötig sein sollte eine große Zwischentabelle zu erstellen, verwenden sie CREATE VIEW. Besser ist es jedoch, direkt stark zu filtern und nur kleine Zwischentabellen per CREATE TABLE zu erstellen. Erzeugen Sie temporäre Tabellen als benannte Standardtabellen mit starker Filterung: Z.B. CREATE TABLE TT_02_temp_table AS SELECT * FROM TT_01_temp_table_already_filtered (...) WHERE (...).
- Verwenden Sie keine *ORDER BY* Anweisungen in *CREATE* Table Anweisungen. Diese werden bei der Erstellung einer Tabelle ignoriert. Verwenden Sie Sortierungen nur, wenn dies nötig ist. Insbesondere die Sortierung großer Datenmengen wirkt sich negativ auf die Laufzeiten aus.
- Verwenden Sie die ANSI-92 Syntax mit explizit formulierten JOIN-Bedingungen und Angaben zu den einzubeziehenden Spalten mit *ON*-Bedingungen des JOINS nicht die ältere unübersichtlichere ANSI-89 Syntax oder DB spezifische Kurzformen die vordefinierte Fremdschlüssel voraussetzen.
- Vermeiden Sie umfangreiche *CASE WHEN* Abfragen, zum Beispiel zur Altersgruppen-Bestimmung. Verwenden Sie in diesen Fällen Schlüsseltabellen, die Sie dann über einen JOIN einbinden.

2.3 Für R

- Beachten Sie den [Tidyverse Style Guide](#)
- Zeichencodierung: UTF-8 Textdateien
- Dateiendung: .R oder .Rmd
- Rufen Sie alle im Programmcode verwendeten Bibliotheken im ersten Teil Ihrer Syntax auf.
- Definieren Sie anschließend alle Funktionen, die Sie für Ihren Code benötigen und beschreiben deren Funktionalität.
- Beachten Sie die [Ergänzende Hinweise zum Arbeiten mit R und Python](#).

2.4 Für Python

- Beachten Sie den [PEP 8 – Style Guide for Python Code](#). Dies ist für Python auch syntaktisch erforderlich.
- Zeichencodierung: UTF-8 Textdateien
- Dateiendung: .py oder .ipynb
- Rufen Sie alle im Programmcode verwendeten Bibliotheken im ersten Teil Ihrer Syntax auf.
- Definieren Sie anschließend alle Funktionen, die Sie für Ihren Code benötigen und beschreiben deren Funktionalität.
- Beachten Sie die [Ergänzende Hinweise zum Arbeiten mit R und Python](#).

3 Programmcoderegeln mit speziellem Fokus auf den Identifikationsschutz

Ergebnistabellen, die außerhalb des integrierten Systems des Forschungsdatenzentrums zur Publikation exportiert werden sollen, müssen auf Wahrung der Anonymität überprüft werden. Neben den zuvor genannten Hinweisen zum Aufbau des Programmcodes müssen daher die folgenden Punkte berücksichtigt werden:

- **Wichtig! Beachten Sie die Vorgaben aus der Anlage für die Ergebnismengenprüfung!**
- **Datensparsamkeit:** Erstellen Sie so wenige Ergebnistabellen mit so wenig Spalten und Zeilen wie möglich. Vorgaben zum Umfang und der Struktur von Ergebnistabellen finden Sie in diesem Dokument.
- Für die Prüfung wird die Information benötigt, auf Basis welcher Fallzahlen die einzelnen Werte in einer Tabelle berechnet worden sind. Dies gilt beispielsweise für die Berechnung von Anteilswerten. Bei der Verwendung von Gewichtungsfaktoren müssen auch die ungewichteten Fallzahlen abgebildet werden.
- Tabellen sind mit mindestens einer Spalte für die Anzahl der distinkt gezählten Versicherten, die den aggregierten Angaben in den anderen Spalten zugrunde liegen, zu versehen. Als Spaltenname muss CNT_D_PSID verwendet werden. Diese Spalten werden für die Umsetzung der Mindestfallzahlregelung benötigt.

Beispiel: Ambulante ICD-Codes der Gruppe X im Jahr 2010

Tabelle: RT_2010_AMB_ICD_GRUPPE_X

ICD Code	CNT_D_PSID
E110	220
E111	444

- Bei der Berechnung statistischer Modelle ist die Anzahl der Freiheitsgrade in den Ergebnistabellen auszuweisen. Der Spaltenname lautet dann CNT_DF.
- Bedenken Sie die tabellenübergreifende Geheimhaltung. Vermeiden Sie z. B. bei der Verwendung von Filterbedingungen Restmengen, die sich als Differenz zu anderen dargestellten Mengen erschließen lassen. Beispiel: Sie erzeugen eine „Insgesamt“-Tabelle und eine Tabelle „Männlich“; die Differenz ist „Weiblich“. Erzeugen Sie in einem solchen Fall die Tabelle „Weiblich“ explizit mit, da diese bei der Prüfung der Wahrung der Anonymität berücksichtigt werden muss.
- Lösen Sie inhaltliche Bezüge/Abhängigkeiten zwischen und innerhalb von Tabellen auf. Sollte dies nicht möglich sein, minimieren Sie Abhängigkeiten, die durch verschiedenartige Gruppierungen auf den gleichen Daten entstehen. Beachten Sie dabei, dass die Ausweisung von Werten, die die Rückrechnung von modifizierten Werten ermöglichen und somit zur Deanonymisierung geeignet sind, in Ergebnistabellen nicht erlaubt sind. Nicht modifizierte Randsummen sind in jedem Fall unzulässig, wenn zugleich die den Randsummen zugrundeliegenden Werte ausgewiesen werden.
- Vermeiden Sie Gruppierungskaskaden und inhaltliche Wiederholungen in den Ergebnistabellen. Im Beispiel sollten Sie nur die Tabelle 3 ausgeben lassen, da dies die feinste Gruppierung ist und die anderen Gruppierungen sich aus dieser Tabelle errechnen lassen.
 - Tabelle 1 – Daten A gruppiert nach Alter
 - Tabelle 2 – Daten A gruppiert nach Geschlecht
 - Tabelle 3 – Daten A gruppiert nach Alter und Geschlecht
- Gibt es Tabellen, die nur der Plausibilisierung oder dem Identifikationsschutz dienen, dann trennen Sie diese Tabellen namentlich per Präfix „TT_“ von den eigentlichen Ergebnistabellen. Tabellen, die diesen Zwecken dienen, sind nicht Teil der zu exportierenden Ergebnismenge.
- Vermeiden Sie einen unnötigen Detaillierungsgrad. Wenn Sie kalkulierende SQL-Funktionen verwenden, runden Sie Ergebnisse vor der Beantragung der Ergebnismenge bitte auf eine sinnvolle Anzahl an Nachkommastellen.

4 Regeln für Variablen und Tabellen

4.1 Regeln zur Benennung von Variablen

- Vergeben Sie für jede Variable und selbstgeschriebene Funktion einen inhaltlich klaren, eindeutigen und sprechenden Namen.
- Halten Sie sich an die entsprechenden Style Guides der verwendeten Programmiersprache.
- Bitte wechseln Sie nicht zwischen Benennungskonventionen (zum Beispiel von snake_case zu CamelCase) oder Sprachen (von Deutsch zu Englisch).

4.2 Regeln zur Benennung von Tabellen

- Benennen Sie alle Tabellen nach einem einheitlichen Schema.
- Ihre gewählten Tabellennamen dürfen maximal 30 Zeichen lang sein.
- Tabellen müssen selbsterklärend sein, d.h. aus den Tabellenspalten und -zeilen muss der Tabelleninhalt eindeutig hervorgehen.
- Kürzel in der Benennung der Tabelle geben Hinweise zum Inhalt der Tabelle.

Tabelle 1: Übersicht Tabellen

Kürzel	Bedeutung	Kommentar	Länge Tabellenname
RT_	Result Table	RT-Tabellen stellen die zu exportierende Ergebnismenge dar. Da Zwischenergebnistabellen eine Vorschau für Ergebnistabellen sind, werden diese ebenfalls mit dem Präfix „RT_“ benannt.	Max. 30 Zeichen
TT_	Temporary Table	RT-Tabellen werden aus TT-Tabellen erstellt.	Max. 30 Zeichen
VT_	Value Table	Eine Tabelle mit Schlüsselwerten	Max. 30 Zeichen
DD_RESULT_TABLES	Data Description	Eine inhaltliche Beschreibung der Ergebnistabellen, die unverändert immer exakt mit dem Tabellennamen DD_RESULT_TABLES benannt ist.	16 Zeichen
DD_RESULT_COLS	Data Description	Eine inhaltliche Beschreibung der Spalten der Ergebnistabellen, die unverändert immer exakt mit dem Tabellennamen DD_RESULT_COLS benannt ist.	14 Zeichen

4.3 Regeln für erzeugte Tabellenstrukturen

4.3.1 Regeln zum Umfang: Maximale Tabellen- und Spaltenanzahl

Restriktionen für temporäre Tabellen in der Analysephase:

- Auf Basis der Testdaten können im integrierten FDZ-System nahezu beliebig viele temporäre Tabellen in einer Analyse erstellt werden. Hier besteht eine systemgegebene Begrenzung durch die System-Ressourcen, die mehrere zeitgleiche Analysen anderer Benutzer sicherstellen müssen.
- Das Tabellenpräfix für temporäre Tabellen lautet „TT_“.

Restriktionen für Zwischenergebnisse in der Analysephase:

- Zwischenergebnisse dürfen maximal 10 Spalten pro Tabelle enthalten. Liegen mehr als 10 Spalten vor, werden die überzähligen Spalten automatisch entfernt
- Wird in einer Tabelle die Zellenanzahl von 480 überschritten, werden überzählige Zellen automatisch entfernt.
- Zwischenergebnisse müssen aggregiert sein, d.h. sie dürfen keine Pseudonyme oder Daten auf Einzeldatensatzebene enthalten
- Es können bis zu fünfmal Zwischenergebnisse im Antragsportal des FDZ kostenpflichtig angefordert werden und pro Zwischenergebnislauf werden maximal 10 Zwischenergebnistabellen bereitgestellt. Nur die Zwischenergebnistabellen des jeweils aktuellen Zwischenergebnislaufs werden im Analyse-raum angezeigt, alte Zwischenergebnistabellen werden gelöscht.

Forschungsdatenzentrum Gesundheit

- Das Tabellenpräfix für Zwischenergebnistabellen lautet „RT_“ wie bei den Ergebnistabellen. Nur potenzielle Ergebnistabellen dürfen per Zwischenergebnis angefordert werden.

Restriktionen für die zu exportierende publizierbare Ergebnismenge:

- Das Tabellenpräfix für Ergebnistabellen lautet „RT_“.
- Ergebnistabellen müssen verständlich, prüfbar, exportierbar und publizierbar sein.
- Die Anzahl der publizierbaren Tabellen, die exportiert werden können, sind **pro Antrag auf 5 Tabellen begrenzt**. Dies dient der fristgemäßen Durchführbarkeit der Identifikationsschutzprüfung und sichert antragsübergreifend eine zügige Verfahrensabwicklung. Falls antragstellende Personen mehr als 5 Tabellen anfordern, werden nach Rücksprache maximal fünf Tabellen für den Export freigegeben.
- Im Antragsportal pro Tabelle können derzeit technisch bedingt **maximal 10 Spalten** beantragt werden. Falls mehr als 10 Spalten benötigt werden:
In diesem Fall bitte zunächst die Datenspalten und (falls vorhanden) Prüfungsspalten anlegen. Beschreibungsspalten bis zur maximalen Spaltenzahl anlegen. Falls zusätzliche Beschreibungsspalten benötigt werden, dann diese Beschreibungsspalten bitte in der Tabellenbeschreibung erläutern.
- Spalten in Ergebnismengen enthalten berechnete Zellen (Datenzellen) oder beschreibende Zellen (Beschreibungszellen). Für die Prüfung der Minimierung des Reidentifikationsrisikos können zusätzliche Zellen erforderlich sein (Prüfzellen).
- Spalten bestehen einheitlich aus einem der drei genannten Zelltypen.
 - Datenzellen:
z.B. Mittelwert, Fallzahl der Versicherten (CNT_D_PSID). Die Spalte CNT_D_PSID muss als Datenzelle(n) oder als Prüfungszelle(n) angelegt sein. Pro Ergebnismenge sind **höchstens 2.400 Datenzellen** zulässig. Antragstellende können 2.400 Datenzellen frei auf fünf Tabellen verteilen.
 - Beschreibungszellen:
z. B. Geschlecht, Alter oder Diagnosecodes oder Daten aus Schlüsselstabellen. Beschreibungsspalten sind der Anzahl nur durch technische Grenzen limitiert.
 - Prüfungszellen:
Für die Freigabeprüfung relevante Zellen (z.B. CNT_D_PSID, siehe Abschnitt 3). Diese sind vom Antragsteller verpflichtend in der Ergebnistabelle anzulegen, werden aber nicht in die maximale Zellenanzahl eingerechnet, da diese Spalten beim Export entfernt werden.
- Das Forschungsdatenzentrum Gesundheit behält sich vor, in Einzelfällen die Gesamtzahl der Zellen einer Ergebnismenge zu beschränken, wenn dies zur Sicherstellung der Prüfbarkeit erforderlich ist, um die Kapazitäten des FDZ nicht unverhältnismäßig zu binden.

Regeln für Kennzahltabellen

- Die Zieltabelle mit den maximal 5 Kennzahlen ist mit dem Präfix RT_KENNZ zu versehen. Diese Tabelle darf maximal 5 Zeilen und 2 Spalten haben (NAME=Name der Kennzahl, VALUE=Wert der Kennzahl).

4.3.2 Regeln zu erlaubten Datentypen und Feldlängen

- Für Zwischenergebnisse und Ergebnismengen sind nur Formate zugelassen die einen CSV-Export ermöglichen. Diese sind:
 - DATE, TIME, SECONDDATE, TIMESTAMP, TINYINT, SMALLINT, INTEGER, BIGINT, SMALLDECIMAL, DECIMAL, REAL, DOUBLE, BOOLEAN
 - VARCHAR, NVARCHAR, ALPHANUM; maximal 95 Zeichen (theoretisches Maximum in den Originaldaten)

4.3.3 Erforderliche Tabellen zur Beschreibung der Ergebnismenge

Um die Ergebnismengen eines Projekts klar zu dokumentieren, müssen im Analyseraum zwei Metadaten-Tabellen mit den Namen DD_RESULT_TABLES und DD_RESULT_COLS angelegt werden. Diese beiden Tabellen beschreiben, welche Ergebnistabellen und Spalten exportiert werden, und machen so die Struktur der Ergebnisdaten für eine Prüfung nachvollziehbar. Ein Beispielskript zur Erzeugung dieser Tabellen wird vom FDZ im „Hilfen“-Unterverzeichnis im Git bereitgestellt.

Tabelle DD_RESULT_TABLES: Diese Tabelle enthält eine tabellenweise Beschreibung der zu exportierenden Ergebnismenge. Jede Zeile beschreibt eine Ergebnistabelle

Erläuterung der Spalten:

- **Tabellenname:** Interner Bezeichner der Ergebnistabelle (z. B. RT_1)
- **Klassifikation_Versichertenbezug_Leistungsraegerbezug:** Kontextbezug der Tabelle
- **Klassifikation_statistisch:** Art der dargestellten Statistik (z. B. Mittelwert, Häufigkeitstabelle)
- **Beschreibung:** Klartextbeschreibung des Inhalts
- **Einschluss_Filterkriterien:** Angaben zu ggf. angewandten Einschränkungen auf die Grundgesamtheit

Tabelle 2: Beispiel für eine DD_RESULT_TABLES in vorgegebener Struktur

Tabellenname	Klassifikation_Versichertenbezug_Leistungsraegerbezug	Klassifikation_statistisch	Beschreibung	Einschluss_Filterkriterien
RT_1	Versichertenbezug	Mittelwert	Kennzahl: Mittelwert des Alters der betrachteten Versicherten	Nur Versicherte der Geburtsjahrgänge 1980-1989
RT_2	Versichertenbezug	Häufigkeitstabelle	Kreuztabelle aus Altersgruppe und Geschlecht	Alle beantragten Versicherten

Tabelle DD_RESULT_COLS: Diese Tabelle beschreibt die in den Ergebnistabellen enthaltenen Spalten auf Spaltenebene.

Erläuterung der Spalten:

- **Tabellenname:** Referenz auf die in DD_RESULT_TABLES definierte Ergebnistabelle
- **Klassifikation_statistisch:** Art der verwendeten statistischen Methode
- **Spaltenname:** Bezeichner der Spalte in der Ergebnistabelle
- **Beschreibung:** Klartextbeschreibung der dargestellten Information

Tabelle 3: Beispiel für eine DD_RESULT_COLS in vorgegebener Struktur

Tabellenname	Klassifikation_statistisch	Spaltenname	Beschreibung
RT_1	Mittelwert	MW_Alter	Mittelwert des Alters der betrachteten Versicherten
RT_1	Fallzahl	CNT_D_PSID	Fallzahl der Versicherten
RT_2	Häufigkeitstabelle	Altersgruppe	Altersgruppe der betrachteten Versicherten
RT_2	Häufigkeitstabelle	Geschlecht	Geschlecht der betrachteten Versicherten
RT_2	Fallzahl	CNT_D_PSID	Fallzahl der Versicherten

5 Hinweise

5.1 Regeln zur automatischen Ausführung von Skripten

Um Skripte auf den Originaldaten im isolierten System automatisiert auszuführen, beachten Sie bitte folgende Festlegungen:

- Legen Sie ihr Skript im Hauptverzeichnis der Versionsverwaltung GIT ab.
- Je nach im Antragsformular ausgewählter Programmiersprache müssen Sie Ihre Skriptdatei folgendermaßen benennen:
- Im SAP HANA Database Explorer: <Aktenzeichen>-main.sql, z. B. P31851-123-main.sql
- In Posit: main.*Dateiformat* (.R, .Rcmd oder .py, .ipynb)

5.2 Speicherung von Skripten und Ergebnissen

Beim Anfordern einer Lauffähigkeitsprüfung des Auswertungsskripts (Button im Antragsportal) wird die aktuelle R oder Python Session im Analyseraum im integrierten System beendet und alle laufenden Datenbankabfragen in allen Arten von Analyseräumen abgebrochen. Stellen Sie sicher, dass Sie Skripte im GIT und Tabellen in der Datenbank gespeichert haben, bevor Sie eine Lauffähigkeitsprüfung anfordern.

5.3 Ergänzende Hinweise zum Arbeiten mit R und Python

Zusätzlich zu SQL ist entweder R oder Python zu verwenden, nicht beide Programmiersprachen. Die Echtdaten werden in der Datenbank und nicht in den Statistikprogrammen vorgehalten. Alle zu erstellenden Tabellen sind explizit in der Datenbank zu speichern. Dies spart Speicherplatz und erhöht in der Regel die Ausführungsgeschwindigkeit Ihres Skriptes. Zusätzlich ist der Datentransfer zwischen der Datenbank und den Statistikprogrammen auf das Nötigste zu beschränken. Aufwändig erzeugte Tabellen, auf die im Verlauf erneut zugegriffen wird, sollten in der Datenbank gespeichert werden und nicht auf dem Dateisystem des Analyseraums.

5.3.1 Vorgelagerte Datenaufbereitung per SQL-Skript

Der erste Schritt einer Analyse ist die Auswahl der Versicherten, die Ihre Auswertungspopulation definieren. In der Regel führt dies zu einem gezielten Ausschluss von Versicherten und damit zu einer Verringerung der zu analysierenden Daten. In weiteren Schritten werden dann in der Regel auf Basis der Auswertungspopulation weitere Versichertengruppen gebildet, die nachfolgend detailliert analysiert werden. Auch diese Schritte zur Bildung der Untersuchungsgruppen führen oft zu einer Verringerung der zu analysierenden Daten. **Wichtig! Führen Sie alle datenreduzierenden Schritte so früh wie möglich und mit Hilfe eines ggf. auch mehrerer SQL-Skripte durch!** Dies ist bei den teilweise großen Datensätzen der Tabellen zwingend notwendig. Optimal wäre es, wenn das Python- oder R-Skript lediglich die Operationen ausführen würde, die nur in diesen Programmen mit vertretbarem Aufwand umsetzbar sind. Ein Vorteil dieser Herangehensweise ist eine performantere Analyse.

Anhang

Liste mit nicht ausführbaren SQL-Befehlen

SQL Statement	Quelle
ALTER AUDIT POLICY Statement (Access Control)	Link
ALTER CLIENTSIDE ENCRYPTION COLUMN KEY Statement (Encryption)	Link
ALTER CLIENTSIDE ENCRYPTION KEYPAIR Statement (Encryption)	Link
ALTER CREDENTIAL Statement (Access Control)	Link
ALTER DATABASE Statement (Tenant Database Management)	Link
ALTER JWT PROVIDER Statement (Access Control)	Link
ALTER LDAP PROVIDER Statement (Access Control)	Link
ALTER PSE Statement (System Management)	Link
ALTER REMOTE SOURCE Statement (Access Control)	Link
ALTER ROLE Statement (Access Control)	Link
ALTER SAML PROVIDER Statement (Access Control)	Link
ALTER STRUCTURED PRIVILEGE Statement (Access Control)	Link
ALTER SYSTEM ... (All ALTER SYSTEM Statements)	Link
ALTER SYSTEM {ADD ALTER REMOVE} STATEMENT HINT Statement (System Management)	Link
ALTER SYSTEM {ADD ACTIVATE UPDATE DROP} KEY MANAGEMENT CONFIGURATION Statement (System Management)	Link
ALTER SYSTEM {ADD REMOVE} ABSTRACT SQL PLAN FILTER (System Management)	Link
ALTER SYSTEM ALTER AUDIT LOG (System Management)	Link
ALTER SYSTEM ALTER CONFIGURATION Statement (System Management)	Link
ALTER SYSTEM ALTER DATAVOLUME ADD PARTITION Statement (System Management)	Link
ALTER SYSTEM ALTER DATAVOLUME DROP PARTITION Statement (System Management)	Link
ALTER SYSTEM ALTER SESSION SET Statement (System Management)	Link
ALTER SYSTEM ALTER SESSION UNSET Statement (System Management)	Link
ALTER SYSTEM ALTER TABLE PLACEMENT Statement (System Management)	Link
ALTER SYSTEM APPLICATION ENCRYPTION Statement (System Management)	Link
ALTER SYSTEM BACKUP ENCRYPTION Statement (System Management)	Link
ALTER SYSTEM CANCEL [WORK IN] SESSION Statement (System Management)	Link
ALTER SYSTEM CLEAR AUDIT LOG Statement (System Management)	Link
ALTER SYSTEM CLEAR CACHE Statement (System Management)	Link
ALTER SYSTEM CLEAR COLUMN JOIN DATA STATISTICS (System Management)	Link
ALTER SYSTEM CLEAR INIFILE CONTENT HISTORY Statement (System Management)	Link
ALTER SYSTEM CLEAR RESULT CACHE Statement (System Management)	Link
ALTER SYSTEM CLEAR SQL PLAN CACHE Statement (System Management)	Link
ALTER SYSTEM CLEAR TIMEZONE CACHE Statement (System Management)	Link

Forschungsdatenzentrum Gesundheit

ALTER SYSTEM CLEAR TRACES Statement (System Management)	Link
ALTER SYSTEM CLIENTPKI { UPDATE / DROP } (System Management)	Link
ALTER SYSTEM CREATE RUNTIMEDUMP Statement (System Management)	Link
ALTER SYSTEM CREATE WAITGRAPH Statement (System Management)	Link
ALTER SYSTEM DISCONNECT SESSION Statement (System Management)	Link
ALTER SYSTEM {ENABLE DISABLE REMOVE} ABSTRACT SQL PLAN (System Management)	Link
ALTER SYSTEM {ENABLE DISABLE} ALL [ASYNCHRONOUS SYNCHRONOUS] TABLE REPLICAS Statement (System Management)	Link
ALTER SYSTEM {ENABLE DISABLE} STATEMENT HINT Statement (System Management)	Link
ALTER SYSTEM {ENABLE DISABLE} SYSTEM REPLICATION Statement (System Management)	Link
ALTER SYSTEM ENCRYPTION CONFIGURATION Statement (System Management)	Link
ALTER SYSTEM LOAD PERFTRACE Statement (System Management)	Link
ALTER SYSTEM LOG ENCRYPTION Statement (System Management)	Link
ALTER SYSTEM LOGGING Statement (System Management)	Link
ALTER SYSTEM MIGRATE ABSTRACT SQL PLAN (System Management)	Link
ALTER SYSTEM PERSISTENCE ENCRYPTION Statement (System Management)	Link
ALTER SYSTEM {PIN UNPIN} SQL PLAN CACHE ENTRY Statement (System Management)	Link
ALTER SYSTEM RECLAIM DATA SPACE Statement (System Management)	Link
ALTER SYSTEM RECLAIM DATAVOLUME Statement (System Management)	Link
ALTER SYSTEM RECLAIM FLEXIBLE TABLES Statement (System Management)	Link
ALTER SYSTEM RECLAIM LOB SPACE Statement (System Management)	Link
ALTER SYSTEM RECLAIM LOG Statement (System Management)	Link
ALTER SYSTEM RECLAIM VERSION SPACE Statement (System Management)	Link
ALTER SYSTEM RECOMPILE SQL PLAN CACHE ENTRY Statement (System Management)	Link
ALTER SYSTEM RECONFIGURE SERVICE Statement (System Management)	Link
ALTER SYSTEM REFRESH RESULT CACHE Statement (System Management)	Link
ALTER SYSTEM REFRESH RESULT CACHE ENTRY Statement (System Management)	Link
ALTER SYSTEM {REGISTER UNREGISTER} SYSTEM REPLICATION SITE Statement (System Management)	Link
ALTER SYSTEM RELOAD FILE PSE Statement (System Management)	Link
ALTER SYSTEM REMOVE RESULT CACHE ENTRY Statement (System Management)	Link
ALTER SYSTEM REMOVE SQL PLAN CACHE ENTRY Statement (System Management)	Link
ALTER SYSTEM REMOVE TRACES Statement (System Management)	Link
ALTER SYSTEM RESET MONITORING VIEW Statement (System Management)	Link
ALTER SYSTEM SAVE PERFTRACE Statement (System Management)	Link
ALTER SYSTEM SAVEPOINT Statement (System Management)	Link
ALTER SYSTEM SET ENCRYPTION ROOT KEYS BACKUP PASSWORD Statement (System Management)	Link
ALTER SYSTEM {START STOP} APPLY ABSTRACT SQL PLAN (System Management)	Link

Forschungsdatenzentrum Gesundheit

ALTER SYSTEM {START STOP SAVE CLEAR} KERNEL PROFILER Statement (System Management)	Link
ALTER SYSTEM {START STOP} CAPTURE ABSTRACT SQL PLAN (System Management)	Link
ALTER SYSTEM {START STOP CLEAR} SQLSCRIPT PLAN PROFILER Statement (System Management)	Link
ALTER SYSTEM {START STOP} PERFTRACE Statement (System Management)	Link
ALTER SYSTEM {START STOP} SQLSCRIPT CODE COVERAGE Statement (System Management)	Link
ALTER SYSTEM START DATABASE Statement (Tenant Database Management)	Link
ALTER SYSTEM STOP DATABASE Statement (Tenant Database Management)	Link
ALTER SYSTEM STOP SERVICE Statement (System Management)	Link
ALTER SYSTEM UPDATE ABSTRACT SQL PLAN Statement (System Management)	Link
ALTER SYSTEM VALIDATE ENCRYPTION ROOT KEYS BACKUP PASSWORD Statement (System Management)	Link
ALTER USER Statement (Access Control)	Link
ALTER USERGROUP Statement (Access Control)	Link
ALTER WORKLOAD CLASS Statement (Workload Management)	Link
ALTER WORKLOAD MAPPING Statement (Workload Management)	Link
ALTER X509 PROVIDER (Access Control)	Link
BACKUP CANCEL Statement (Backup and Recovery)	Link
BACKUP CATALOG DELETE Statement (Backup and Recovery)	Link
BACKUP CATALOG RETAINED Statement (Backup and Recovery)	Link
BACKUP CHECK Statement (Backup and Recovery)	Link
BACKUP CHECK ACCESS Statement (Backup and Recovery)	Link
BACKUP DATA Statement (Backup and Recovery)	Link
BACKUP DATA CLOSE SNAPSHOT Statement (Backup and Recovery)	Link
BACKUP DATA CREATE SNAPSHOT Statement (Backup and Recovery)	Link
BACKUP DATA DROP SNAPSHOT Statement (Backup and Recovery)	Link
BACKUP ENCRYPTION ROOT KEYS Statement (Backup and Recovery)	Link
BACKUP LIST DATA Statement (Backup and Recovery)	Link
COMMENT ON Statement (Data Definition)	Link
CREATE AUDIT POLICY Statement (Access Control)	Link
CREATE CERTIFICATE Statement (System Management)	Link
CREATE CLIENTSIDE ENCRYPTION COLUMN KEY Statement (Encryption)	Link
CREATE CLIENTSIDE ENCRYPTION KEYPAIR Statement (Encryption)	Link
CREATE CREDENTIAL Statement (Access Control)	Link
CREATE DATABASE Statement (Tenant Database Management)	Link
CREATE JWT PROVIDER Statement (Access Control)	Link
CREATE LDAP PROVIDER Statement (Access Control)	Link
CREATE PSE Statement (System Management)	Link

Forschungsdatenzentrum Gesundheit

CREATE PUBLIC KEY Statement (System Management)	Link
CREATE REMOTE SOURCE Statement (Access Control)	Link
CREATE ROLE Statement (Access Control)	Link
CREATE SAML PROVIDER Statement (Access Control)	Link
CREATE SCHEDULER JOB Statement (Data Definition)	Link
CREATE STRUCTURED PRIVILEGE Statement (Access Control)	Link
CREATE USER Statement (Access Control)	Link
CREATE USERGROUP Statement (Access Control)	Link
CREATE WORKLOAD CLASS Statement (Workload Management)	Link
CREATE WORKLOAD MAPPING Statement (Workload Management)	Link
CREATE X509 PROVIDER (Access Control)	Link
DELETE Statement (Data Manipulation)	Link
DROP AUDIT POLICY Statement (Access Control)	Link
DROP CERTIFICATE Statement (System Management)	Link
DROP CLIENTSIDE ENCRYPTION COLUMN KEY Statement (Encryption)	Link
DROP CLIENTSIDE ENCRYPTION KEYPAIR Statement (Encryption)	Link
DROP COLLECTION Statement (JSON Document Store)	Link
DROP CREDENTIAL Statement (Access Control)	Link
DROP DATABASE Statement (Tenant Database Management)	Link
DROP FULLTEXT INDEX Statement (Data Definition)	Link
DROP GRAPH WORKSPACE Statement (Data Definition)	Link
DROP INDEX Statement (Data Definition)	Link
DROP JWT PROVIDER Statement (Access Control)	Link
DROP LDAP PROVIDER Statement (Access Control)	Link
DROP PSE Statement (System Management)	Link
DROP PUBLIC KEY Statement (System Management)	Link
DROP REMOTE SOURCE Statement (Access Control)	Link
DROP ROLE Statement (Access Control)	Link
DROP SAML PROVIDER Statement (Access Control)	Link
DROP SCHEDULER JOB Statement (Data Definition)	Link
DROP SCHEMA Statement (Data Definition)	Link
DROP SCHEMA SYNONYM Statement (Data Definition)	Link
DROP SEQUENCE Statement (Data Definition)	Link
DROP STATISTICS Statement (Data Definition)	Link
DROP STRUCTURED PRIVILEGE Statement (Access Control)	Link
DROP SYNONYM Statement (Data Definition)	Link
DROP TABLE Statement (Data Definition)	Link

Forschungsdatenzentrum Gesundheit

DROP TRIGGER Statement (Data Definition)	Link
DROP USER Statement (Access Control)	Link
DROP USERGROUP Statement (Access Control)	Link
DROP VIEW Statement (Data Definition)	Link
DROP WORKLOAD CLASS Statement (Workload Management)	Link
DROP WORKLOAD MAPPING Statement (Workload Management)	Link
DROP X509 PROVIDER (Access Control)	Link
EXPLAIN PLAN Statement (Data Manipulation)	Link
EXPORT Statement (Data Import Export)	Link
EXPORT INTO Statement (Data Import Export)	Link
GRANT Statement (Access Control)	Link
IMPORT Statement (Data Import Export)	Link
IMPORT FROM Statement (Data Import Export)	Link
IMPORT SCAN Statement (Data Import Export)	Link
RECOVER DATA Statement (Backup and Recovery)	Link
RECOVER DATABASE Statement (Backup and Recovery)	Link
RECOVER ENCRYPTION ROOT KEYS Statement (Backup and Recovery)	Link
RELEASE SAVEPOINT Statement (Transaction Management)	Link
RENAME DATABASE Statement (Tenant Database Management)	Link
RENAME SCHEMA Statement (Data Definition)	Link
REVOKE Statement (Access Control)	Link
SET PSE Statement (System Management)	Link
SET SYSTEM LICENSE Statement (System Management)	Link
UNSET PSE Statement (System Management)	Link
UNSET SYSTEM LICENSE ALL Statement (System Management)	Link
VALIDATE LDAP PROVIDER Statement (Access Control)	Link
VALIDATE USER Statement (Access Control)	Link