



Sektorenübergreifende Qualitätssicherung im Gesundheitswesen nach § 137a SGB V

Technische Dokumentation zur Risikostatistik

Erfassungsjahr 2013

Stand: 30. Juni 2012

Impressum

Herausgeber:

AQUA – Institut für angewandte Qualitätsförderung und Forschung im Gesundheitswesen GmbH

Thema:

Risikostatistik 2013

Auftraggeber:

Gemeinsamer Bundesausschuss

Hinweis:

Aus Gründen der leichten Lesbarkeit wird im Folgenden auf eine geschlechtsspezifische Differenzierung verzichtet. Entsprechende Begriffe gelten im Sinne der Gleichbehandlung für beide Geschlechter.

Anschrift des Herausgebers:

AQUA – Institut für angewandte Qualitätsförderung und
Forschung im Gesundheitswesen GmbH
Maschmühlenweg 8-10 · 37073 Göttingen

Telefon: (+49) 0551 - 789 52 -0

Telefax: (+49) 0551 - 789 52-10

office@aqua-institut.de

www.aqua-institut.de

Inhaltsverzeichnis

| | |
|--|----|
| Tabellenverzeichnis | 4 |
| 1. Hintergrund..... | 5 |
| 1.1. Was ist die Risikostatistik? | 5 |
| 1.2. Zielsetzung der technischen Dokumentation | 5 |
| 1.3. Allgemeine Anmerkungen zur Struktur der Spezifikation | 5 |
| 1.4. Tabellenstruktur der Datenbank | 5 |
| 2. Grundlegende Tabellen | 7 |
| 2.1. Struktur der Datensatzdefinitionen | 7 |
| 2.1.1. Datensätze | 7 |
| 2.1.2. Teildatensätze | 7 |
| 2.1.3. Felder der Teildatensätze | 8 |
| 2.1.4. Felder | 8 |
| 2.1.5. Basistypen | 8 |
| 2.1.6. Schlüssel | 9 |
| 2.1.7. Schlüsselwerte | 9 |
| 2.2. ICD-Listen | 9 |
| 3. Der Risikostatistik-Eingangsdatensatz | 10 |
| 4. Risikostatistik | 11 |
| 4.1. Felder der Risikostatistik | 11 |
| 4.2. Algorithmus zur Berechnung der Risikostatistik | 12 |
| 4.2.1. Medizinische Teilbedingungen | 12 |
| 4.2.2. Administratives Einschlusskriterium | 14 |
| 4.3. Struktur und Syntax der Bedingungen | 15 |
| 4.3.1. Die Variablen der Bedingungen | 15 |
| 4.3.2. Diagnoselisten | 15 |
| 4.3.3. Die Operatoren der Bedingungen | 15 |
| 4.4. Fehlerprüfung | 16 |
| 4.5. Berechnung der Risikostatistik | 18 |
| 4.5.1. Datensatzdefinition der Risikostatistik | 18 |
| 4.5.2. Exportformat der Risikostatistik | 18 |
| 4.5.3. Plausibilitätsprüfungen der Risikostatistik | 19 |
| 4.5.4. E-Mail-Verfahren: Komprimierung und Verschlüsselung | 20 |
| Anhang | 22 |
| Erläuterungen zur Generierung der Tabelle RSDatensatz | 22 |

Tabellenverzeichnis

| | |
|---|----|
| Tabelle 1: Struktur der Tabelle Ds | 7 |
| Tabelle 2: Struktur der Tabelle Tds | 7 |
| Tabelle 3: Struktur der Tabelle BasisTyp | 8 |
| Tabelle 4: Struktur der Tabelle Schluessel | 9 |
| Tabelle 5: Struktur der Tabelle SchluesselWert | 9 |
| Tabelle 6: Felder des Risikostatistik-Eingangsdatensatzes | 10 |
| Tabelle 7: Felder der Risikostatistik (Exportdatei) | 11 |
| Tabelle 8: Struktur der Tabelle RSDatensatz | 14 |
| Tabelle 9: Struktur der Tabelle AdminKriterium | 14 |
| Tabelle 10: Basistypen der Variablen | 15 |
| Tabelle 11: Präzedenz und Assoziativität der Operatoren | 16 |
| Tabelle 12: Fehlercodes | 17 |
| Tabelle 13: Ländercodes der Landesgeschäftsstellen | 20 |

1. Hintergrund

1.1. Was ist die Risikostatistik?

Zur Verringerung des Dokumentationsaufwandes sollen im Leistungsbereich Dekubitus ab dem Erfassungsjahr 2013 händische Dokumentationen nur noch für Patienten mit Dekubitus durchgeführt werden. Ergänzend hierzu werden für eine sachgerechte Risikoadjustierung jedoch weiterhin aggregierte Basisinformationen für die gesamte Zielpopulation des Leistungsbereiches benötigt. Zukünftig werden diese Informationen automatisiert auf Basis vorhandener Routinedaten erstellt und in einer Risikostatistik zusammengefasst.

Die technische Umsetzung der Risikostatistik und das vorliegende Dokument sind weitgehend an die bekannten Strukturen und Vorgehensweisen im Rahmen der QS-Filter-Software angelehnt. Es handelt sich jedoch um einen separaten Prozess.

1.2. Zielsetzung der technischen Dokumentation

Die Spezifikation für die Risikostatistik wird als Datenbank zur Verfügung gestellt. Das vorliegende Dokument erläutert die Struktur der Datenbank und gibt Hilfestellung bei der Umsetzung der Spezifikation.

1.3. Allgemeine Anmerkungen zur Struktur der Spezifikation

Die Spezifikation ist in einer relationalen Datenbank abgelegt. Sie wird vorerst als Access-Datenbank (MS Access 2000) zur Verfügung gestellt. Der Name der Risikostatistik-Spezifikation richtet sich nach folgendem Schema:

`spez-Risikostatistik-<Version>.mdb`

`<Version>` bezeichnet die Versionsnummer (z.B. 2013).

1.4. Tabellenstruktur der Datenbank

Die Tabellen und deren Spalten (Attribute) unterliegen einem einheitlichen Namensschema. Erlaubte Zeichen sind die Buchstaben a-z, A-Z und die Ziffern 0-9. Umlaute und Sonderzeichen werden nicht verwendet. Das erste Zeichen eines Namens darf keine Ziffer sein.

Ein Tabellename beginnt immer mit einem Großbuchstaben und ein Attributname mit einem Kleinbuchstaben. Wenn ein Name aus mehreren Teilen (z.B. Substantiven) besteht, so beginnt jeder nachfolgende Namensteil mit einem Großbuchstaben.

Beispiel:

`BasisTyp` (Tabelle)

`idBasisTyp` (Spalte)

Für jede Tabelle ist in der Spezifikation ein Primärschlüssel definiert, der nach folgendem Schema aufgebaut ist:

`id<TabellenName>`

Der Ausdruck in den eckigen Klammern ist ein Platzhalter für den Namen der Tabelle. Die meisten Tabellen haben einen einfachen Primärschlüssel vom Typ `AUTOINCREMENT`. Zusätzlich enthalten derartige Tabellen

mindestens ein identifizierendes Attribut¹, welches durch Setzen eines weiteren, eindeutigen Index (bestehend aus einem oder mehreren Attributen) definiert ist.

Beispiele:

- Identifizierendes Attribut: Attribut `name` in Tabelle `BasisTyp`
- Identifizierende Attributkombination: Attribute `code` und `fkSchluessel` in Tabelle `SchluesselWert`

Es gibt auch Tabellen, deren einziger eindeutiger Schlüssel der Primärschlüssel ist. Ein Beispiel ist die Tabelle `MussKann` mit dem Primärschlüssel `idMussKann` vom Typ `TEXT(1)` (entspricht `VARCHAR(1)`). Diese Tabellen sind als einfache "Nachschlagtabellen" zu interpretieren. Im Fall der Tabelle `MussKann` soll im entsprechenden Fremdschlüsselfeld der verknüpften Detailtabelle durch das Datenbankschema gewährleistet werden, dass nur ein 'M' oder 'K' eingegeben werden darf.

Die Namen von Fremdschlüsseln sind analog zum Namen der Primärschlüssel aufgebaut:

`fk<FremdTabellenName>`

Die Namensgebung von Primär- und Fremdschlüsseln vereinfacht den Aufbau von komplexeren Abfragen, welche sich über mehrere Entitäten erstrecken (Inklusionsverknüpfungen, Joins).

Die Fremdschlüsselattribute (Namen beginnen mit `fk`) wurden in MS Access als Datenbankattribute zum Nachschlagen eingerichtet. Z.B. wird beim Fremdschlüsselattribut `fkDs` in der Tabelle `Tds` nicht mehr der Primärschlüssel des jeweiligen Datensatzes, sondern der Name des Datensatzes angezeigt. Diese Änderung betrifft nur die Anzeige, nicht jedoch die Struktur der Datenbank!!!

Sind zwei Tabellen mehrfach durch Schlüssel-Fremdschlüssel-Beziehungen miteinander verknüpft, so kann der Name eines Fremdschlüssels auch folgendermaßen aufgebaut sein:

`fk<FremdTabellenName><Rolle>`

`<Rolle>` ist der Platzhalter für eine zusätzliche Qualifizierung der Relation.

N-M-Beziehungen werden wie üblich über Verknüpfungstabellen realisiert. In der Spezifikation haben Verknüpfungstabellen gewöhnlich keinen Primärschlüssel², jedoch einen eindeutigen Schlüssel, der über die Fremdschlüsselfelder definiert ist.

Folgende Attribute treten in vielen Tabellen auf und seien hier kurz erläutert:

- `name` ist in der Regel als "technischer Name" zu verstehen.
- `bezeichnung` ist eine kurze Beschreibung.

¹ oder eine identifizierende Attributkombination, die einen eindeutigen Schlüssel definiert

² Hier: Primärschlüssel im Sinne der Access-Definition eines Primärschlüssels. Streng genommen wird über die beiden Fremdschlüssel ein neuer Primärschlüssel definiert.

2. Grundlegende Tabellen

2.1. Struktur der Datensatzdefinitionen

In den nachfolgenden Abschnitten dieses Unterkapitels wird die grundlegende Struktur der Tabellen der Datensatzdefinitionen beschrieben:

- Jeder Datensatz besteht aus Teildatensätzen, welche ausgehend von einem Basis-Teildatensatz hierarchisch angeordnet sind (Abschnitt 2.1.2).
- Die Definitionen der Datenfelder werden in den Abschnitten 2.1.3 - 2.1.7 erläutert.

2.1.1. Datensätze

In der QS-Spezifikation zur Risikostatistik sind zwei Datensätze definiert:

- Der Risikostatistik-Eingangsdatsatz (beschreibt die Datengrundlage zur Erstellung der Risikostatistik, vgl. Kapitel 3)
- Der Datensatz zur Risikostatistik (Kapitel 4).

Für jeden dieser Datensätze wird in der Tabelle D_s ein Eintrag angelegt.

Tabelle 1: Struktur der Tabelle D_s

| Feldname | Feldtyp | Bemerkung |
|-------------|---------|---|
| idDs | INTEGER | Primärschlüssel |
| name | TEXT | Technischer Name des Datensatzes (Identifizierendes Attribut) |
| bezeichnung | TEXT | Beschreibender Text |

2.1.2. Teildatensätze

Die Definition von Teildatensätzen findet sich in der Tabelle $T_d s$ der Datenbank. Jeder Teildatensatz ist eindeutig durch seinen Namen (z.B. FALL oder DIAG) charakterisiert.

Tabelle 2: Struktur der Tabelle $T_d s$

| Feldname | Feldtyp | Bemerkung |
|---------------|---------|---|
| idTds | INTEGER | Primärschlüssel |
| name | TEXT | Technischer Name des Teildatensatzes (Identifizierendes Attribut) |
| bezeichnung | TEXT | Beschreibender Text |
| fkTds | INTEGER | Optionalen Fremdschlüssel zu einem Mutter-Teildatensatz |
| fkDs | INTEGER | Bezug des Teildatensatzes zum übergeordneten Datensatz in der Tabelle D_s (Risikostatistik-Eingangsdatsatz oder Datensatz für Risikostatistik) |
| fkRelationTyp | TEXT(1) | Relationstyp, bezieht sich auf die Relation zum Mutterteildatensatz: * Eine beliebige Anzahl von Teildatensätzen darf angelegt werden! ? Höchstens ein Teildatensatz darf angelegt werden! + Mindestens ein Teildatensatz muss angelegt werden! 1 Genau ein Teildatensatz muss angelegt werden! |

In der Tabelle `Tds` ist eine Hierarchie der Teildatensätze definiert. Der Ausgangspunkt ("root") für die Teildatensatzhierarchie eines Datensatzes (z.B. Eingangsdatensatz) ist immer der Basis-Teildatensatz (= Teildatensatz³ `fkTds=NULL` in der Tabelle `Tds`). Über die in den restlichen Teildatensätzen definierten Bezüge zu den Mutter-Teildatensätzen und Relationstypen lässt sich ein "Hierarchiebaum" der Teildatensätze (=Datensatz) aufbauen.

Jeder Datensatz besteht aus

- genau einem Basis-Teildatensatz
- ggf. ein oder mehreren weiteren Teildatensätzen (= Kind-Teildatensätze)

Beispiele:

- Der Risikostatistik-Eingangsdatensatz besitzt den Basis-Teildatensatz `FALL` und den Kind-Teildatensatz `DIAG`.
- Der Datensatz Risikostatistik besitzt den Basis-Teildatensatz `RSBASIS` und den Kind-Teildatensatz `RISIKOSTATISTIK`.

2.1.3. Felder der Teildatensätze

Die Felder eines Teildatensatzes sind in der Tabelle `TdsField` definiert. Jedes Feld eines Teildatensatzes (kurz TDS-Feld) ist eindeutig charakterisiert durch die Zugehörigkeit zu einem Teildatensatz und zum referenzierten Feld.

Jedes Feld darf also nur einmal in einem Teildatensatz verwendet werden.

2.1.4. Felder

Ein Feld wird eindeutig über seinen technischen Namen definiert. Jedes Feld hat einen Basistyp (z.B. `SCHLUESSEL`, `NUMSCHLUESSEL`, `ZAHL`). Schlüsselfelder erfordern zusätzlich einen Schlüssel (siehe Kapitel 2.1.6).

2.1.5. Basistypen

Das Hauptmerkmal eines Basistyps ist der technische Typ eines Eingabefeldes (z.B. Zeichenkette, numerischer Typ, Datum usw.). Weiteres Charakteristikum ist die Beschreibung des Eingabeformats. Die Basistypen sind Voraussetzung für die Beschreibung einer formalen Regelsyntax.

Das identifizierende Merkmal eines Basistyps ist sein technischer Name (Attribut `name`).

Tabelle 3: Struktur der Tabelle `BasisTyp`

| Feldname | Feldtyp | Bemerkung |
|------------------------------|---------|---|
| <code>idBasisTyp</code> | INTEGER | Primärschlüssel |
| <code>name</code> | TEXT | Technischer Name (muss eindeutig sein) |
| <code>bezeichnung</code> | TEXT | |
| <code>formatAnweisung</code> | TEXT | Formatdefinition, z.B. TT.MM.JJJJ beim Basistyp Datum |

Anmerkungen

- In Zeichenketten (Basistyp `TEXT`) sind alle Zeichen des ASCII-Formats mit einem Kode > 32 erlaubt. Ausgenommen sind das Semikolon, die doppelten Anführungsstriche und Hochkommata.
- Es gibt zwei Arten von Schlüsseln: numerische und nichtnumerische.
- Das Komma trennt die Nachkommastellen, Vorzeichen + und – sind erlaubt.

³ Es darf nur eine Definition eines Basis-Teildatensatzes existieren

- Das Datumstrennzeichen ist der Punkt.

2.1.6. Schlüssel

Identifizierendes Merkmal eines Schlüssels ist sein technischer Name. Die meisten Schlüsselkodes sind in der Tabelle `SchluessselWert` (Kapitel 2.1.7) definiert. Externe Schlüsselkataloge (z.B. ICD10) sind von den entsprechenden Anbietern zu beziehen.

Tabelle 4: Struktur der Tabelle `Schluesssel`

| Feldname | Feldtyp | Bemerkung |
|----------------------------|---------|---|
| <code>idSchluesssel</code> | INTEGER | Primärschlüssel |
| <code>name</code> | TEXT | Technischer Name (muss eindeutig sein) |
| <code>bezeichnung</code> | TEXT | Beschreibender Text |
| <code>extern</code> | BOOL | zeigt an, ob der Schlüssel in der Tabelle <code>Schluesssel</code> oder in einer externen Tabelle gespeichert ist. |
| <code>externVerweis</code> | TEXT | Verweis auf externe Quelle des Schlüsselkataloges |
| <code>zahl</code> | BOOL | Wenn wahr, wird das Attribut <code>code</code> der zugehörigen Schlüsselwerte als ganze Zahl interpretiert, ansonsten als Zeichenkette. |

Schlüsselkodes können auf zwei Arten kodiert werden. Wenn das Attribut `zahl` gesetzt ist, so werden die Codes als ganze Zahl interpretiert. Ansonsten werden sie als Zeichenketten angesehen.

Externe Schlüsselkataloge

Externe Schlüsselkataloge sind über das Attribut `extern` deklariert. Hinweise zu den Bezugsquellen sind in der Spalte `externVerweis` zu finden (z.B. <http://www.dimdi.de>). Externe Schlüsselkataloge werden nicht vom AQUA-Institut bereitgestellt und somit auch nicht verantwortet.

Achtung:

Der Softwareanbieter hat dafür Sorge zu tragen, dass die aktuellen externen Schlüsselkataloge in der Software verwendet werden.

2.1.7. Schlüsselwerte

Identifizierendes Merkmal ist hier eine Kombination der Spalten `fkSchluesssel` und `code`. Das bedeutet, dass jeder Schlüsselcode innerhalb eines Schlüssels nur einmal vorkommen darf.

Tabelle 5: Struktur der Tabelle `SchluessselWert`

| Feldname | Feldtyp | Bemerkung |
|--------------------------------|---------|---|
| <code>idSchluessselWert</code> | INTEGER | Primärschlüssel |
| <code>fkSchluesssel</code> | INTEGER | Fremdschlüssel zur Tabelle <code>Schlüssel</code> |
| <code>code</code> | INTEGER | 0,1,2 .. |
| <code>bezeichnung</code> | TEXT | Textliche Definition des Schlüsselwertes |

2.2. ICD-Listen

Die Struktur der ICD-Listen (Tabellen `ICDListe` und `ICDWert`) entspricht den gleichnamigen Tabellen der QS-Spezifikationen für QS-Dokumentationssoftware und QS-Filter-Software. Daher sei hier auf die technische Dokumentation zur QS-Spezifikation für QS-Dokumentationssoftware verwiesen.

3. Der Risikostatistik-Eingangsdatensatz

Die Algorithmen zur Berechnung der Risikostatistik basieren auf administrativen und medizinischen Datenfeldern. Für die Erstellung der Risikostatistik sollen die im KIS befindlichen Routinedaten des Krankenhauses herangezogen werden. Die für die Risikostatistik benötigten Datenfelder finden sich im Aufnahmesatz und in der Entlassungsanzeige des §301-Datenformats. Fast alle benötigten Datenfelder können aus der Struktur des §301-Datensatzes abgeleitet werden. Einzige Ausnahme ist das Feld `Alter`.

Der Risikostatistik-Eingangsdatensatz besteht aus zwei Teildatensätzen:

- Behandlungsfall (Teildatensatz `FALL`)
- Diagnoseangaben des Behandlungsfalls (Teildatensatz `DIAG`, wiederholbar)

Tabelle 6: Felder des Risikostatistik-Eingangsdatensatzes

| Feld | Beschreibung | M/K | n Zeichen | Format |
|--|--|-----|-----------|-------------|
| Behandlungsfall: | | | | |
| <code>FALLNUMMER</code> | KH-interne Fallnummer Die Fallnummer dient der eindeutigen Identifizierung eines Krankenhausfalls. Sie wird vom Krankenhaus vergeben. Im Teildatensatz <code>FALL</code> ist sie ein Primärschlüssel, in den anderen Teildatensätzen ist sie als Fremdschlüssel einzufügen. | M | an..15 | |
| <code>AUFNDATUM</code> | Datum der Aufnahme ins Krankenhaus | M | an 10 | TT.MM.JJJJ |
| <code>ENTLDATEM</code> | Datum Entlassung aus Krankenhaus Es gelten die Daten, mit denen die Verweildauer der DRG berechnet wird. | M | an 10 | TT.MM.JJJJ |
| <code>PATALTER</code> | Alter in Jahren am Aufnahmetag | M | n..3 | GANZEZAHL |
| <code>DAUBEAT</code> | Beatmungstunden (vgl. Entlassungsanzeige §301) | K | n6 | GANZEZAHL |
| <code>GESCHLECHT</code> | Geschlecht | M | an 1 | Geschlecht |
| <code>AUFNGRUND</code> | Aufnahmegrund §301–Aufnahmesatz (1. und 2. Stelle): 01 - Krankenhausbehandlung, vollstationär 02 - Krankenhausbehandlung, vollstationär mit vorausgegangener vorstationärer Behandlung 03 - Krankenhausbehandlung, teilstationär 04 - vorstationäre Behandlung ohne anschließende vollstationäre Behandlung 05 - Stationäre Entbindung 06 - Geburt 07 - Wiederaufnahme wegen Komplikationen (Fallpauschale) nach KFPV 2003 08 - Stationäre Aufnahme zur Organentnahme 09 - frei | M | n..2 | AufnGrund |
| Diagnoseangaben (wiederholbar): | | | | |
| <code>FALLNUMMER</code> | KH-interne Fallnummer | M | an..15 | |
| <code>ICD</code> | Diagnoseschlüssel (ICD-Kode) | M | an..9 | ICD-Katalog |

4. Risikostatistik

4.1. Felder der Risikostatistik

Die Felder der Risikostatistik werden aus dem Risikostatistik-Eingangsdatensatz berechnet (Tabelle 6).

Für die Berechnung der Risikostatistik wird als Exportdatei eine Tabelle erzeugt, welche die in Tabelle 7 beschriebenen Felder beinhaltet. Dabei stellen die ersten Felder (DEK2 bis WEIBLICH) die zu prüfenden Risikofaktoren dar. Für einen Fall kann der jeweilige Risikofaktor erfüllt sein oder nicht. Daher sind die Spalten DEK2 bis WEIBLICH jeweils mit 1 und 0 (wahr oder falsch) kodiert. In der Risikostatistik sollen alle denkbaren Konstellationen in einer Matrix dargestellt werden. Das Feld „ANZAHLFAELLE“ berechnet sich anschließend für jede Konstellation an Risikofaktoren über die Anzahl an Fällen, die die beschriebenen Bedingungen erfüllen.

Tabelle 7: Felder der Risikostatistik (Exportdatei)

| Feld | Bezeichnung | Beschreibung | Format |
|------------|--------------------------|--|-------------------|
| DEK2 | Dek Grad 2 | DIAG EINSIN (L89.1) UND DIAG KEINSIN (L89.2;L89.3;L89.9) | wahr/falsch (1,0) |
| DEK3 | Dek Grad 3 | DIAG EINSIN (L89.2) UND DIAG KEINSIN (L89.3;L89.9) | wahr/falsch (1,0) |
| DEK4 | Dek Grad 4 | DIAG EINSIN (L89.3) UND DIAG KEINSIN (L89.9) | wahr/falsch (1,0) |
| DEKnnb | Dek Grad nnb | DIAG EINSIN (L89.9) | wahr/falsch (1,0) |
| ALTER3343 | Alter 33 bis 43 Jahre | ALTER >= 33 UND ALTER <= 43 | wahr/falsch (1,0) |
| ALTER4451 | Alter 44 bis 51 Jahre | ALTER >= 44 UND ALTER <= 51 | wahr/falsch (1,0) |
| ALTER5258 | Alter 52 bis 58 Jahre | ALTER >= 52 UND ALTER <= 58 | wahr/falsch (1,0) |
| ALTER45965 | Alter 59 bis 65 Jahre | ALTER >= 59 UND ALTER <= 65 | wahr/falsch (1,0) |
| ALTER6670 | Alter 66 bis 70 Jahre | ALTER >= 60 UND ALTER <= 70 | wahr/falsch (1,0) |
| ALTER7174 | Alter 71 bis 74 Jahre | ALTER >= 71 UND ALTER <= 74 | wahr/falsch (1,0) |
| ALTER7578 | Alter 75 bis 78 Jahre | ALTER >= 75 UND ALTER <= 78 | wahr/falsch (1,0) |
| ALTER7984 | Alter 79 bis 84 Jahre | ALTER >= 79 UND ALTER <= 84 | wahr/falsch (1,0) |
| ALTER85 | Alter ab 85 Jahre | ALTER >= 85 | wahr/falsch (1,0) |
| DMTYP12 | Diabetes Typ 1 und 2 | DIAG EINSIN (E10; E11) UND DIAG KEINSIN (E12; E13; E14; E16) | wahr/falsch (1,0) |
| DMTYPNNB | Diabetes Typ nnb | DIAG EINSIN (E12; E13; E14; E16) UND DIAG KEINSIN (E10; E11) | wahr/falsch (1,0) |
| DAUBEAT18 | Beatmung 1 bis 8 Stunden | DAUBEAT >= 1 UND DAUERBEAT <= 8 | wahr/falsch (1,0) |

| Feld | Bezeichnung | Beschreibung | Format |
|--------------|---------------------------------|--|-------------------|
| DAUBEAT924 | Beatmung 9 bis 24 Stunden | DAUBEAT >= 9 UND DAUERBEAT <= 24 | wahr/falsch (1,0) |
| DAUBEAT2572 | Beatmung 25 bis 72 Stunden | DAUBEAT >= 25 UND DAUERBEAT <= 72 | wahr/falsch (1,0) |
| DAUBEAT73240 | Beatmung 73 bis 240 Stunden | DAUBEAT >= 73 UND DAUERBEAT <= 240 | wahr/falsch (1,0) |
| DAUBEAT241 | Beatmung länger als 240 Stunden | DAUBEAT > 240 | wahr/falsch (1,0) |
| PARAPARESE | Paraparese | DIAG EINSIN (G82) | wahr/falsch (1,0) |
| WEIBLICH | Geschlecht (weiblich) | GESCHLECHT = w | wahr/falsch (1,0) |
| ANZAHLFAELLE | Anzahl | Anzahl an Fällen, für die die in der Exportdatei angegebene Kombination der oben beschriebenen Bedingungen zutrifft. | GANZEZAHL |

4.2. Algorithmus zur Berechnung der Risikostatistik

Für jeden Datensatz der Risikostatistik (RS-Datensatz) ist in der Tabelle `RSDatensatz` ein formal logischer Ausdruck definiert, welcher sich aus einer medizinischen Teilbedingung und einer administrativen Teilbedingung zusammensetzt:

`Zeile = medBedingung UND AdminBedingung`

Für jeden Fall soll geprüft werden, ob die in der jeweiligen Zeile beschriebenen Bedingungen zutreffen. In der Spalte `ANZAHLFAELLE` ist die Gesamtzahl an Fällen zu berechnen, die die jeweilige Konstellation an Bedingungen erfüllen.

4.2.1. Medizinische Teilbedingungen

Für jeden Datensatz der Risikostatistik (RS-Datensatz) ist in der Tabelle `RSDatensatz` eine auf die Risikofaktoren bezogene Bedingung in der Spalte `medBedingung` hinterlegt.

Beispiel:

Der RS-Datensatz mit `idRSDatensatz = 1` hat folgende `medBedingung`:

`DEK2 = 1 UND DEK3 = 0 UND DEK 4 = 0 UND DEKnnb = 0 UND ALTER3343 = 1 UND ALTER4451 = 0 UND ALTER5258 = 0 UND ALTER5965 = 0 UND ALTER6670 = 0 UND ALTER7174 = 0 UND ALTER7578 = 0 UND ALTER7984 = 0 UND ALTER85 = 0 UND DMTYP12 = 1 UND DMNNB = 0 UND DAUBEAT18 = 1 UND DAUBEAT924 = 0 UND DAUBEAT2572 = 0 UND DAUBEAT73240 = 0 UND DAUBEAT241 = 0 UND PARAPARESE = 1 UND WEIBLICH = 1`

Diese setzt sich aus folgenden Teilbedingungen zusammen:

- DEK2 = 1
- DEK3 = 0
- DEK4 = 0
- DEKnnb = 0
- ALTER3343 = 1
- ALTER4451 = 0
- ALTER5258 = 0
- ALTER5965 = 0
- ALTER6670 = 0
- ALTER7174 = 0
- ALTER7578 = 0
- ALTER7984 = 0
- ALTER85 = 0
- DMTYP12= 1
- DMNNB = 0
- DAUBEAT18 = 1
- DAUBEAT924 = 0
- DAUBEAT2572 = 0
- DAUBEAT73240 = 0
- DAUBEAT241 = 0
- PARAPARESE = 1
- WEIBLICH = 1

Zur Generierung der Risikostatistik ist für jeden Fall zu prüfen, ob der höchstgradige Dekubitalulkus dem Grad 2 entspricht, der Patient zwischen 33 und 43 Jahren alt ist, der Patient an einer Diabetes mellitus Typ 1 oder 2 erkrankt ist, zwischen einer und 18 Stunden beatmet wurde, eine G82-Diagnose aufweist und vom Geschlecht weiblich ist. Anschließend wird die Summe aller gezählten Fälle, auf die diese Bedingungen zutreffen, in der Spalte ANZAHLFAELLE ausgewiesen.

Beispiel:

Dieses Beispiel zeigt einen Ausschnitt der Exportdatei RISIKOSTATISTIK_2013.TXT.

```
DEK2;DEK3;DEK4;DEKnnb;ALTER3343;ALTER4451;ALTER5258;ALTER5965;ALTER6670;ALTER7174;ALTER7578;A
LTER7984;ALTER85;DIAB1U2;DIABNNB;DAUBEAT18;DAUBEAT924;DAUBEAT2572;DAUBEAT73240;DAUBEAT241;PAR
APARESE;WEIBLICH;ANZAHLFAELLE<CR><LF>
1;0;0;0;1;0;0;0;0;0;0;0;0;1;0;1;0;0;0;0;1;1<CR><LF>
1;0;0;0;1;0;0;0;0;0;0;0;0;1;0;1;0;0;0;0;1;0<CR><LF>
1;0;0;0;1;0;0;0;0;0;0;0;0;1;0;1;0;0;0;0;1<CR><LF>
1;0;0;0;1;0;0;0;0;0;0;0;0;1;0;1;0;0;0;0;0<CR><LF>
```

Nachfolgende Tabelle gibt einen Überblick über die Tabelle `RSDatensatz`.

Tabelle 8: Struktur der Tabelle `RSDatensatz`

| Feldname | Feldtyp | Bemerkung |
|-------------------------------|---------|---|
| <code>idRSDatensatz</code> | INTEGER | Primärschlüssel |
| <code>medBedingung</code> | MEMO | Medizinisch-inhaltliche Definition bzw. Beschreibung von Risikofaktoren |
| <code>fkAdminKriterium</code> | INTEGER | Bezug zum administrativen Abgrenzungskriterium |

4.2.2. Administratives Einschlusskriterium

Die medizinisch-inhaltlichen Teilbedingungen müssen für diejenigen Fälle geprüft werden, die das sogenannte administrative Einschlusskriterium erfüllen, also wenn

1. Der Patient stationär aufgenommen wird.
2. Der Behandlungsfall eine bestimmte zeitliche Klammer aufweist.

Zu 1)

Die Bedingung „stationär aufgenommen“ wird aus der Angabe im Datenfeld `AufnahmeGrund` abgeleitet:

```
AUFNGRUND NICHTIN (3;4) UND AUFNGRUND <> LEER
```

Diese Teilbedingung ist Teil des administrativen Kriteriums (siehe Attribut `bedingung` in Tabelle `AdminKriterium`).

Zu 2)⁴

In die Berechnung der Risikostatistik sollen nur Patienten einbezogen werden, welche im Jahre 2013 aufgenommen und bis zum 31. Dezember 2013 entlassen wurden:

```
AUFNDATUM >= '01.01.2013' UND AUFNDATUM <= '31.12.2013' UND ENTLDATUM <= '31.12.2013'
```

Nachfolgende Tabelle gibt einen Überblick über die Tabelle `AdminKriterium`.

Tabelle 9: Struktur der Tabelle `AdminKriterium`

| Feldname | Feldtyp | Bemerkung |
|-------------------------------|---------|---|
| <code>idAdminKriterium</code> | INTEGER | Primärschlüssel |
| <code>name</code> | TEXT | Technischer Name des administrativen Kriteriums |
| <code>bedingung</code> | MEMO | Definition der Einschlussbedingung |
| <code>info</code> | TEXT | Bezeichnungstext des administrativen Kriteriums |

⁴ verbindlich sind die in der Spezifikationsdatenbank hinterlegten Kriterien

4.3. Struktur und Syntax der Bedingungen

4.3.1. Die Variablen der Bedingungen

Die in den Bedingungen erlaubten Variablen sind in der Tabelle `SyntaxVariable` definiert.

Die Variablennamen (Attribut `SyntaxVariable.name`) bestehen aus maximal 32 Zeichen. Sie dürfen nur die Buchstaben A bis Z (Großbuchstaben) und die Ziffern 0 bis 9 enthalten. Ein Feldname muss immer mit einem Buchstaben beginnen. Umlaute und Sonderzeichen sind in Feldnamen nicht erlaubt. Ein Feldname darf auch nicht ein reserviertes Wort sein (z.B. Namen von Operatoren wie `EINSIN`).

Typen

Jede Variable hat einen Basistyp.

Jeder in der Tabelle `SyntaxVariable` definierten Variablen ist über den Wert des Attributes `SyntaxVariable.fkTdsFeld` ein Feld des Risikostatistik-Eingangsdatensatzes zugeordnet. Jedes dieser Felder besitzt einen Feldtyp und somit auch einen Basistyp.

Tabelle 10: Basistypen der Variablen

| Datentyp | Bezeichnung | Beispiele (Literele) |
|---------------|---|----------------------|
| BOOL | Boolsche Variable | WAHR, FALSCH |
| TEXT | Zeichenkette (String) | "Spezifikation" |
| GANZEZAHL | .. -2, -1, 0, 1, 2, 3, .. | 1 |
| ZAHL | Zahl (mit oder ohne Nachkommastellen) | 25,4 oder -100,8 |
| DATUM | Zehnstelliges Datum | 01.01.2012 |
| NUMSCHLUESSEL | Numerisch kodierter Schlüssel (wie <code>GANZEZAHL</code>) | 0 |
| SCHLUESSEL | Alphanumerischer Schlüssel | '19.1' |
| UHRZEIT | Uhrzeit | '10:15' |

Listen

Eine Variable wird als Liste interpretiert, wenn der Wert des Attributes `SyntaxVariable.istListe` entsprechend gesetzt ist.

Literele

Bis auf die numerischen Typen (`GANZEZAHL` und `ZAHL`) müssen Literale von einfachen Hochkommata eingeschlossen sein oder Zeichenketten von doppelten Anführungsstrichen.

4.3.2. Diagnoselisten

Die erlaubten Namen von Diagnoselisten sind in der Tabelle `ICDListe` (Attribut `name`) zu finden.

Die jeweiligen Codes (nach ICD-10-GM bzw. OPS) sind in der durch Fremdschlüsselbeziehungen verknüpften Tabelle `ICDWert` enthalten. Alle Codes sind terminal. Die Diagnosecodes enthalten nicht die Suffixe +, * oder !.

4.3.3. Die Operatoren der Bedingungen

Die nachfolgende Tabelle gibt einen Überblick über die in der Syntax zulässigen Operatoren. Der aktuelle Überblick über alle zulässigen Operationen (inkl. Operanden) ist in Tabelle `SyntaxOperator` der Spezifikation zu finden.

Tabelle 11: Präzedenz und Assoziativität der Operatoren⁵

| Präzedenz | Assoziativität | Operator | Erläuterung |
|-----------|----------------|----------|-------------------------------------|
| 0 | links | EINSIN | Mengenoperator |
| | links | KEINSIN | Mengenoperator |
| 3 | links | < | Vergleichsoperator „kleiner“ |
| | links | > | Vergleichsoperator „größer“ |
| | links | <= | Vergleichsoperator „kleiner gleich“ |
| | links | >= | Vergleichsoperator „größer gleich“ |
| 4 | links | = | Vergleichsoperator |
| | links | <> | Vergleichsoperator „ungleich“ |
| 5 | rechts | NICHT | logisches Nicht |
| 6 | links | UND | logisches Und |
| 7 | links | ODER | logisches Oder |

Anmerkungen:

Operatoren mit beidseitigen Listenfeldern als Operanden:

EINSIN: Wenn mindestens ein Element aus der linken Liste in der rechten Liste enthalten ist, so ist der Ausdruck wahr (nichtleere Schnittmenge).

KEINSIN: Wenn kein Element der linken Liste in der rechten Liste enthalten ist, so ist der Ausdruck wahr (Leere Schnittmenge)⁶.

Beispiele:

Folgende Regel prüft, ob ein Element der Listenvariable **DIAG** einen bestimmten Code besitzt:

```
DIAG EINSIN DEK4_ICD
```

String-Vergleich bei Diagnosen

Die Operation **EINSIN** basiert für Diagnosen auf einem Vergleich von Zeichenketten (String-Vergleich): Für jedes Element der linken Liste wird über einen String-Vergleich geprüft, ob es in der rechten Liste enthalten ist.

Eine Modifikation des String-Vergleiches ist beim String-Vergleich von Diagnosen nötig: Hier werden die Suffixe „+“, „*“ und „!“ ignoriert. Nicht ignoriert wird das Sonderzeichen „.“.

4.4. Fehlerprüfung

Vor Evaluation der in den vorhergehenden Abschnitten beschriebenen Zählbedingungen ist eine Fehlerprüfung durchzuführen. Die Fehlerprüfung bezieht sich auf die Daten des Risikostatistik-Eingangsdatensatzes.

⁵ In dieser Übersichtstafel hat jeder einzelne Operator eine Präzedenzstufe (höchste Präzedenzstufe ist 0). Operatoren, welche die gleiche Stufe haben, werden stattdessen nach den Regeln der Assoziativität aufgelöst.

⁶ Dieser Operator ist redundant, da er auch durch Negation des **EINSIN**-Operators abgedeckt ist.

Tabelle 12: Fehlercodes

| Beschreibung des Fehlers | Fehler-kode | Formale Fehlerbedingung | Fehlermeldung |
|---|-------------|---|--|
| Überprüfung des Formats | 1 | siehe Kapitel 3.2.1 Technische Dokumentation für QS-Dokumentationssoftware 2013 | Der Wert '<WERT>' des Datenfeldes <TdsFeld.name> ist kein gültiger <BasisTyp.name>-Wert (<BasisTyp.bezeichnung> <BasisTyp.format>). |
| Überprüfung der Feldlänge | 2 | siehe Kapitel 3.2.2 Technische Dokumentation für QS-Dokumentationssoftware 2013 | Der Wert '<WERT>' des Datenfeldes <TdsFeld.name> überschreitet die zulässige Feldlänge <Feld.laenge>. |
| Sind in den Datenfeldern mit internen und externen Schlüsseln (BasisTyp SCHLUESSEL oder NUMSCHLUESSEL) gültige Schlüsselcodes eingetragen? ⁷ | 3 | siehe Kapitel 3.2.3 Technische Dokumentation für QS-Dokumentationssoftware 2013 | Ungültiger Schlüsselcode <Wert> des Schlüssels <Schluessel.name> im Datenfeld <TdsFeld.name>! |
| Überprüfung numerischer Wertebereiche | 4 | siehe Kapitel 3.2.4. Technische Dokumentation für QS-Dokumentationssoftware 2013 | Der Wert '<WERT>' des Datenfeldes <TdsFeld.name> ist kleiner als '<Feld.min>' oder Der Wert '<WERT>' des Datenfeldes <TdsFeld.name> ist größer als '<Feld.max>' |
| Ist ein Muss-Feld ausgefüllt? | 5 | siehe Kapitel 3.2.5. Technische Dokumentation für QS-Dokumentationssoftware 2013 Kriterium: Attribut <TdsFeld.fk> | Das Datenfeld <TdsFeld.name> muss einen gültigen Wert enthalten. |
| Ist der Fall dem Erfassungsjahr der Spezifikation zugeordnet? | 6 | AUFNDATUM < '01.01.2013' ODER AUFNDATUM > '31.12.2013' | Der Fall ist im Jahr 2013 nicht aufgenommen: Aufnahmedatum = <Wert> |

Hinweise zu den Fehlermeldungen:

- In <Wert> ist der Wert des Datenfeldes der Falldaten einzusetzen, auf den sich die Fehlermeldung bezieht.
- Ansonsten sind in die < . . >-Ausdrücke die entsprechenden Einträge aus der Datenbank einzusetzen.

Im Fehlerfall sind entsprechende Einträge im Teildatensatz FEHLER zu generieren:

- Fehlercode in Attribut FKODE,
- Fehlermeldung in Attribut FMELDUNG.

Es wird beim Auftreten eines Fehlers nicht weiter geprüft, ob der Behandlungsfall in die Risikostatistik einbezogen wird.

⁷ Die Korrektheit (z.B. ausschließlich gültige terminale Codes oder Verwendung gültiger Katalogversionen) der extern definierten Codes (ICD) muss vor der Prüfung des Datensatzes sichergestellt sein.

4.5. Berechnung der Risikostatistik

Nach Abschluss eines Erfassungsjahres erstellt das Krankenhaus die Risikostatistik, welche eine Aufstellung der für das Krankenhaus geprüften Risikofaktoren darstellt.

Die Berechnung der Risikostatistik darf nur von einem berechtigten Mitarbeiter des Krankenhauses durchgeführt werden. Der Zeitpunkt der Erstellung wird in der Risikostatistik im Datenfeld `DOKABSCHLDATE` protokolliert.

4.5.1. Datensatzdefinition der Risikostatistik

Die Datensatzdefinition der Risikostatistik ist in der Spezifikationsdatenbank hinterlegt. In der Tabelle `DS` ist zusätzlich zum Risikostatistik-Eingangsdatensatz der Datensatz zur Risikostatistik definiert, welcher die Teildatensätze `RSBASIS` (Basisangaben zur Risikostatistik) und `RISIKOSTATISTIK` (berechnete Angaben zu Risikofaktoren) enthält. Zwischen beiden Teildatensätzen ist eine 1-n-Relation (Tabelle `TDS`) definiert: zu dem Eintrag in der Tabelle `RSBASIS` existieren n Einträge in der Tabelle `RISIKOSTATISTIK`.

In den Spalten `hinweis` und `formel` der Tabelle `FELD` sind Ausfüllhinweise und Berechnungsvorgaben zu den Datenfeldern der Risikostatistik hinterlegt.

4.5.2. Exportformat der Risikostatistik

Jede Exportdatei enthält eine Kopfzeile, in welcher die technischen Feldnamen des Datensatzes für die Risikostatistik aufgelistet sind.

Für jeden Teildatensatz der Risikostatistik wird analog zum Exportverfahren der QS-Spezifikation für QS-Dokumentationssoftware eine Exportdatei generiert, welche nach folgendem Schema benannt wird:

`<TDS.name>_<VJahr>.TXT`

Beispiel:

Für das Erfassungsjahr 2013 sind somit die Exportdateien `RSBASIS_2013.TXT` und `RISIKOSTATISTIK_2013.TXT` zu generieren.

Der Aufbau der Exportdateien folgt ebenso der QS-Spezifikation für QS Dokumentationssoftware⁸:

- ASCII-Format, erweiterter IBM-PC-Zeichensatz (8 Bit, DOS/OS/2)
- die erste Zeile enthält die Feldnamen, die nachfolgenden Zeilen enthalten die Werte der Exportfelder eines Teildatensatzes
- Zeilen werden durch `<CR><LF>` (ASCII 13, 10) abgeschlossen
- Feldnamen in der ersten und Werte der Exportfelder in den folgenden Zeilen werden durch Semikola getrennt und können variabel lang sein⁹

Beispiele:

Das folgende Beispiel zeigt die Exportdatei `RSBASIS_2013.txt`.

```
IKNRKH;BSNR;RSJAHR;KH_NAME;DOKABSCHLDATE;RS_SW_PRODUKT;RS_SW_VERSION;RS_SW_HERSTELLER;KH_VERAN
TWORTLICHER;FAELLE_GEPRUEFT<CR><LF>
123456789;1;2013;Krankenhaus XY;01.02.2014;RS TEST;01;Firma xy;Hans Meier;7912<CR><LF>
```

⁸ Die Anonymisierung entfällt hier, da die übermittelten Daten nicht personenbeziehbar sind.

⁹ Hinter dem letzten Feld einer Zeile darf kein Semikolon stehen.

Dieses Beispiel zeigt einen Auszug der Exportdatei RISIKOSTATISTIK_2013.TXT.

```
DEK2;DEK3;DEK4;DEKnnb;ALTER3343;ALTER4451;ALTER5258;ALTER5965;ALTER6670;ALTER7174;ALTER7578;A  
LTER7984;ALTER85;DMTYP12;DMTYPNNB;DAUBEAT18;DAUBEAT924;DAUBEAT2572;DAUBEAT73240;DAUBEAT241;PA  
RAPARESE;WEIBLICH;ANZAHLFAELLE<CR><LF>  
1;0;0;0;1;0;0;0;0;0;0;0;0;0;1;0;1;0;0;0;0;1;1;50<CR><LF>  
1;0;0;0;1;0;0;0;0;0;0;0;0;0;1;0;1;0;0;0;0;1;0;30<CR><LF>  
1;0;0;0;1;0;0;0;0;0;0;0;0;0;1;0;1;0;0;0;0;0;1;27<CR><LF>  
1;0;0;0;1;0;0;0;0;0;0;0;0;0;1;0;1;0;0;0;0;0;0;28<CR><LF>  
1;0;0;0;1;0;0;0;0;0;0;0;0;0;1;0;0;1;0;0;0;1;1;31<CR><LF>  
1;0;0;0;1;0;0;0;0;0;0;0;0;0;1;0;0;1;0;0;0;1;0;7<CR><LF>  
1;0;0;0;1;0;0;0;0;0;0;0;0;0;1;0;0;1;0;0;0;1;9<CR><LF>  
1;0;0;0;1;0;0;0;0;0;0;0;0;0;1;0;0;1;0;0;0;0;12<CR><LF>  
...
```

4.5.3. Plausibilitätsprüfungen der Risikostatistik

Die Risikostatistik muss vollständig und plausibel sein. Unplausible oder unvollständige Risikostatistiken werden von den Datenstellen als nicht geliefert angesehen. Für nicht akzeptierte Risikostatistiken wird ein Fehlerprotokoll erstellt und per E-Mail an das Krankenhaus übermittelt.

Empfehlungen:

- Die Funktionalität zur Übermittlung der Risikostatistik sollte den Krankenhäusern spätestens zum Ende eines Erfassungsjahres bereitgestellt werden.

Krankenhäuser sollten die Risikostatistik möglichst frühzeitig übermitteln, da dann noch genügend Zeit zur Fehlerkorrektur bis zum Stichtag besteht.

4.5.4. E-Mail-Verfahren: Komprimierung und Verschlüsselung

Die Risikostatistik ist als komprimierter und verschlüsselter E-Mail-Anhang an die zuständige Landesgeschäftsstelle Qualitätssicherung bis spätestens zum 28. Februar 2014 zu übermitteln.

Die beiden Exportdateien `RSBASIS_2013.TXT` und `RISIKOSTATISTIK_2013.TXT` werden mit Hilfe eines pkzip 2.04g-kompatiblen Archivierungsprogramms komprimiert und unter dem Namen `RS_2013_<IKNRKH>_<BSNR>.ZIP` gespeichert.

Die Verschlüsselung des ZIP-Archivs erfolgt mit der Verschlüsselungssoftware GnuPG¹⁰ Version 1.2.1 (GNU Privacy Guard). Das gleiche Public-Key-Verschlüsselungsverfahren wird bereits von den deutschen Krankenhäusern im Rahmen der Datenübermittlung nach §21 KHEntG an die DRG-Datenstelle (<http://www.g-drg.de>) eingesetzt.

Für die Verschlüsselung wird der öffentliche Schlüssel der jeweiligen Datenstelle verwendet, an welche die Risikostatistik übermittelt wird.

Das ZIP-Archiv `RS_2013_<IKNRKH>_<BSNR>.ZIP` ist mit dem öffentlichen Schlüssel der zuständigen Landesgeschäftsstelle zu verschlüsseln und unter folgendem Namen zu speichern:

`RS_2013_<IKNRKH>_<BSNR>_<Ländercode>.GPG`

Die einzutragenden Ländercodes werden in der folgenden Tabelle aufgeführt.

Tabelle 13: Ländercodes der Landesgeschäftsstellen

| Ländercode | Landesgeschäftsstelle |
|------------|------------------------|
| BA | Bayern |
| BB | Brandenburg |
| BE | Berlin |
| BW | Baden-Württemberg |
| HB | Bremen |
| HE | Hessen |
| HH | Hamburg |
| MV | Mecklenburg-Vorpommern |
| NI | Niedersachsen |
| NW | Nordrhein-Westfalen |
| RP | Rheinland-Pfalz |
| SH | Schleswig-Holstein |
| SL | Saarland |
| SN | Sachsen |
| ST | Sachsen-Anhalt |
| TH | Thüringen |

¹⁰ Diese frei verfügbare Open-Source-Software unterliegt der GNU General Public Licence.

Beispiel

Das Krankenhaus mit dem Institutionskennzeichen 123456789 und der Betriebsstätte 1 liege in Mecklenburg-Vorpommern.

Dann hat das ZIP-Archiv der Risikostatistik 2013 folgenden Namen:

RS_2013_123456789_1.ZIP

Die verschlüsselte Datei für die LQS hat den Namen:

RS_2013_123456789_1_MV.GPG

Datenstellen der LQS

Die E-Mail-Adressen und öffentlichen Schlüssel für die LQS-Datenübermittlung werden von den Landesgeschäftsstellen Qualitätssicherung bekannt gegeben.

Das absendende Krankenhaus hat dafür Sorge zu tragen, dass die Daten pünktlich gesendet werden und bei dem Empfänger(der LQS) übereinstimmende Inhalte aufweisen. Die fehlerfrei übermittelten Daten werden von der zuständigen Landesgeschäftsstelle an das AQUA-Institut weitergeleitet.

Anhang

Erläuterungen zur Generierung der Tabelle RSDatensatz

Die Tabelle RSDatensatz enthält die Bedingungen, die zur Berechnung der Risikostatistik-Zahlen benötigt werden. Die Bedingungen beziehen sich auf 22 Merkmale, die sich in 6 Gruppen zusammenfassen lassen: Dekubitus-Grad (DEK2, DEK3, DEK4, DEKNNB), Alter (ALTER3343, ALTER4451, ALTER5258, ALTER5965, ALTER6670, ALTER7174, ALTER7578, ALTER7984, ALTER85), Diabetes-Erkrankung (DMTYP12, DMTYPNNB), Beatmungsdauer (DAUBEAT18, DAUBEAT924, DAUBEAT2572, DAUBEAT73240, DAUBEAT241), Paraparese (PARAPARESE) und Geschlecht (WEIBLICH).

Achtung!

Ein Fall kann aus der jeweiligen Gruppe nur eines oder keines der Merkmale erfüllen.

Beispiel:

Ein Fall fällt immer nur in eine einzige Altersgruppe (z.B. 33 – 43 Jahre) oder in keine der zu prüfenden Altersgruppen (z.B. jünger als 33 Jahre).

Aus 6 Gruppen von Risikofaktoren (a, b, c, d, e, f) mit jeweils unterschiedlich vielen Merkmalen resultieren:

$(n(a) + 1) * (n(b) + 1) * (n(c) + 1) * (n(d) + 1) * (n(e) + 1) * (n(f) + 1)$ mögliche Konstellationen.

Bei $(4+1) * (9+1) * (2+1) * (5+1) * (1+1) * (1+1)$ ergeben sich somit 3600 zu prüfende Konstellationen. Diese entsprechen den im Feld `medBedingung` der Tabelle RSDatensatz beschriebenen Prüfbedingungen. Dabei kann der Inhalt der Spalte `medBedingung` automatisiert erzeugt werden. Hierzu wird der folgende Programm-Kode (Java) zur Verfügung gestellt.

```
Klasse Generator --> Generierung der Plausibilitätsregeln
```

```
import java.util.ArrayList;

import org.omg.CORBA.Environment;

public class Generator {

    public void generiereTabelle() {
        ArrayList<Datensatz> liste = new ArrayList<Datensatz>();

        for (int i = 0; i <= Datensatz.anzahlDekGrad; i++) {
            for (int j = 0; j <= Datensatz.anzahlAlter; j++) {
                for (int k = 0; k <= Datensatz.anzahlDiabetes; k++) {
                    for (int l = 0; l <= Datensatz.anzahlBeatmungsstunden; l++) {
                        for (int m = 0; m <= Datensatz.anzahlPARAPARESE; m++) {
                            for (int n = 0; n <= Datensatz.anzahlGeschlecht; n++) {

                                Datensatz aktuellerDatensatz = new Datensatz();
```

```
        if (i < Datensatz.anzahlDekGrad) {
            aktuellerDatensatz.DekGrad[i] = true;
        }

        if (j < Datensatz.anzahlAlter) {
            aktuellerDatensatz.Alter[j] = true;
        }
        if (k < Datensatz.anzahlDiabetes) {
            aktuellerDatensatz.Diabetes[k] = true;
        }
        if (l < Datensatz.anzahlBeatmungsstunden) {
            aktuellerDatensatz.Beatmungsstunden[l] = true;
        }
        if (m < Datensatz.anzahlPARAPARESE) {
            aktuellerDatensatz.PARAPARESE[m] = true;
        }
        if (n < Datensatz.anzahlGeschlecht) {
            aktuellerDatensatz.Geschlecht[n] = true;
        }

        liste.add(aktuellerDatensatz);
    }
}
}
}
}

String ausgabe = "";
for (int i = 0; i <= liste.size() - 1; i++) {
    Datensatz aktuellerDatensatz = liste.get(i);
    ausgabe = ausgabe + "DEK2 = "
        + wandleBoolNachInt(aktuellerDatensatz.DekGrad[0])
        + " & DEK3 = "
        + wandleBoolNachInt(aktuellerDatensatz.DekGrad[1])
        + " & DEK4 = "
        + wandleBoolNachInt(aktuellerDatensatz.DekGrad[2])
        + " & DEKnnb = "
        + wandleBoolNachInt(aktuellerDatensatz.DekGrad[3])
        + " & ALTER3343 = "
        + wandleBoolNachInt(aktuellerDatensatz.Alter[0])
        + " & ALTER4451 = "
        + wandleBoolNachInt(aktuellerDatensatz.Alter[1])
        + " & ALTER5258 = "
        + wandleBoolNachInt(aktuellerDatensatz.Alter[2])
        + " & ALTER5965 = "
```

```
+ wandleBoolNachInt(aktuellerDatensatz.Alter[3])
+ " & ALTER6670 = "
+ wandleBoolNachInt(aktuellerDatensatz.Alter[4])
+ " & ALTER7174 = "
+ wandleBoolNachInt(aktuellerDatensatz.Alter[5])
+ " & ALTER7578 = "
+ wandleBoolNachInt(aktuellerDatensatz.Alter[6])
+ " & ALTER7984 = "
+ wandleBoolNachInt(aktuellerDatensatz.Alter[7])
+ " & ALTER85 = "
+ wandleBoolNachInt(aktuellerDatensatz.Alter[8])
+ " & DMTYP12 = "
+ wandleBoolNachInt(aktuellerDatensatz.Diabetes[0])
+ " & DMTYPNNB = "
+ wandleBoolNachInt(aktuellerDatensatz.Diabetes[1])
+ " & DAUBEAT18 = "
+ wandleBoolNachInt(aktuellerDatensatz.Beatmungsstunden[0])
+ " & DAUBEAT924 = "
+ wandleBoolNachInt(aktuellerDatensatz.Beatmungsstunden[1])
+ " & DAUBEAT2572 = "
+ wandleBoolNachInt(aktuellerDatensatz.Beatmungsstunden[2])
+ " & DAUBEAT73240 = "
+ wandleBoolNachInt(aktuellerDatensatz.Beatmungsstunden[3])
+ " & DAUBEAT241 = "
+ wandleBoolNachInt(aktuellerDatensatz.Beatmungsstunden[4])
+ " & PARAPARESE = "
+ wandleBoolNachInt(aktuellerDatensatz.PARAPARESE[0])
+ " & WEIBLICH = "
+ wandleBoolNachInt(aktuellerDatensatz.Geschlecht[0]);

    ausgabe = ausgabe + "\n";
}

System.out.println(ausgabe);

}

int wandleBoolNachInt(boolean wert) {
    if (wert == true)
        return 1;
    else
        return 0;
}

}
```

Klasse Datensatz --> Länge der Felder definieren

```
public class Datensatz {  
    public static int anzahlDekGrad = 4;  
    public static int anzahlAlter = 9;  
    public static int anzahlDiabetes = 2;  
    public static int anzahlBeatmungsstunden = 5;  
    public static int anzahlPARAPARESE = 1;  
    public static int anzahlGeschlecht = 1;  
    public boolean DekGrad[] = new boolean[anzahlDekGrad];  
    public boolean Alter[] = new boolean[anzahlAlter];  
    public boolean Diabetes[] = new boolean[anzahlDiabetes];  
    public boolean Beatmungsstunden[] = new boolean[anzahlBeatmungsstunden];  
    public boolean PARAPARESE[] = new boolean[anzahlPARAPARESE];  
    public boolean Geschlecht[] = new boolean[anzahlGeschlecht];  
  
}
```

Klasse main --> Aufruf

```
public class Main {  
  
    /**  
     * @param args  
     */  
    public static void main(String[] args) {  
        Generator generator = new Generator();  
        generator.generiereTabelle();  
    }  
}
```

Das Programm erzeugt den im Attribut `medBedingung` der Tabelle `RSDatensatz` beschriebenen Inhalt (aufgrund der hohen Anzahl an unterschiedlichen Konstellationen werden hier nur die ersten 9 von 3600 Zeilen dargestellt).

Ausgabe

```
Dek2 = 1 & Dek3 = 0 & Dek4 = 0 & DEKnnb = 0 & Alter3343 = 1 & Alter4451 = 0 & ALTER5258 = 0 &
ALTER5965 = 0 & ALTER6670 = 0 & ALTER7174 = 0 & ALTER7578 = 0 & ALTER7984 = 0 & ALTER85 = 0 &
DMTYP12 = 1 & DMTYPNNB = 0 & DAUBEAT18 = 1 & DAUBEAT924 = 0 & DAUBEAT2572 = 0 & DAUBEAT73240
= 0 & DAUBEAT241 = 0 & PARAPARESE = 1 & WEIBLICH = 1
```

```
Dek2 = 1 & Dek3 = 0 & Dek4 = 0 & DEKnnb = 0 & Alter3343 = 1 & Alter4451 = 0 & ALTER5258 = 0 &
ALTER5965 = 0 & ALTER6670 = 0 & ALTER7174 = 0 & ALTER7578 = 0 & ALTER7984 = 0 & ALTER85 = 0 &
DMTYP12 = 1 & DMTYPNNB = 0 & DAUBEAT18 = 1 & DAUBEAT924 = 0 & DAUBEAT2572 = 0 & DAUBEAT73240
= 0 & DAUBEAT241 = 0 & PARAPARESE = 1 & WEIBLICH = 0
```

```
Dek2 = 1 & Dek3 = 0 & Dek4 = 0 & DEKnnb = 0 & Alter3343 = 1 & Alter4451 = 0 & ALTER5258 = 0 &
ALTER5965 = 0 & ALTER6670 = 0 & ALTER7174 = 0 & ALTER7578 = 0 & ALTER7984 = 0 & ALTER85 = 0 &
DMTYP12 = 1 & DMTYPNNB = 0 & DAUBEAT18 = 1 & DAUBEAT924 = 0 & DAUBEAT2572 = 0 & DAUBEAT73240
= 0 & DAUBEAT241 = 0 & PARAPARESE = 0 & WEIBLICH = 1
```

```
Dek2 = 1 & Dek3 = 0 & Dek4 = 0 & DEKnnb = 0 & Alter3343 = 1 & Alter4451 = 0 & ALTER5258 = 0 &
ALTER5965 = 0 & ALTER6670 = 0 & ALTER7174 = 0 & ALTER7578 = 0 & ALTER7984 = 0 & ALTER85 = 0 &
DMTYP12 = 1 & DMTYPNNB = 0 & DAUBEAT18 = 1 & DAUBEAT924 = 0 & DAUBEAT2572 = 0 & DAUBEAT73240
= 0 & DAUBEAT241 = 0 & PARAPARESE = 0 & WEIBLICH = 0
```

```
Dek2 = 1 & Dek3 = 0 & Dek4 = 0 & DEKnnb = 0 & Alter3343 = 1 & Alter4451 = 0 & ALTER5258 = 0 &
ALTER5965 = 0 & ALTER6670 = 0 & ALTER7174 = 0 & ALTER7578 = 0 & ALTER7984 = 0 & ALTER85 = 0 &
DMTYP12 = 1 & DMTYPNNB = 0 & DAUBEAT18 = 0 & DAUBEAT924 = 1 & DAUBEAT2572 = 0 & DAUBEAT73240
= 0 & DAUBEAT241 = 0 & PARAPARESE = 1 & WEIBLICH = 1
```

```
Dek2 = 1 & Dek3 = 0 & Dek4 = 0 & DEKnnb = 0 & Alter3343 = 1 & Alter4451 = 0 & ALTER5258 = 0 &
ALTER5965 = 0 & ALTER6670 = 0 & ALTER7174 = 0 & ALTER7578 = 0 & ALTER7984 = 0 & ALTER85 = 0 &
DMTYP12 = 1 & DMTYPNNB = 0 & DAUBEAT18 = 0 & DAUBEAT924 = 1 & DAUBEAT2572 = 0 & DAUBEAT73240
= 0 & DAUBEAT241 = 0 & PARAPARESE = 1 & WEIBLICH = 0
```

```
Dek2 = 1 & Dek3 = 0 & Dek4 = 0 & DEKnnb = 0 & Alter3343 = 1 & Alter4451 = 0 & ALTER5258 = 0 &
ALTER5965 = 0 & ALTER6670 = 0 & ALTER7174 = 0 & ALTER7578 = 0 & ALTER7984 = 0 & ALTER85 = 0 &
DMTYP12 = 1 & DMTYPNNB = 0 & DAUBEAT18 = 0 & DAUBEAT924 = 1 & DAUBEAT2572 = 0 & DAUBEAT73240
= 0 & DAUBEAT241 = 0 & PARAPARESE = 0 & WEIBLICH = 1
```

```
Dek2 = 1 & Dek3 = 0 & Dek4 = 0 & DEKnnb = 0 & Alter3343 = 1 & Alter4451 = 0 & ALTER5258 = 0 &
ALTER5965 = 0 & ALTER6670 = 0 & ALTER7174 = 0 & ALTER7578 = 0 & ALTER7984 = 0 & ALTER85 = 0 &
DMTYP12 = 1 & DMTYPNNB = 0 & DAUBEAT18 = 0 & DAUBEAT924 = 1 & DAUBEAT2572 = 0 & DAUBEAT73240
= 0 & DAUBEAT241 = 0 & PARAPARESE = 0 & WEIBLICH = 0
```

```
Dek2 = 1 & Dek3 = 0 & Dek4 = 0 & DEKnnb = 0 & Alter3343 = 1 & Alter4451 = 0 & ALTER5258 = 0 &
ALTER5965 = 0 & ALTER6670 = 0 & ALTER7174 = 0 & ALTER7578 = 0 & ALTER7984 = 0 & ALTER85 = 0 &
DMTYP12 = 1 & DMTYPNNB = 0 & DAUBEAT18 = 0 & DAUBEAT924 = 0 & DAUBEAT2572 = 1 & DAUBEAT73240
= 0 & DAUBEAT241 = 0 & PARAPARESE = 1 & WEIBLICH = 1
```

...

...

...
