

2015

Rückwärtssalto

Kodras, Tiryaki



Inhalt

Aufgabenstellung.....	2
Requirements-Analyse	3
Designüberlegung.....	3
Aufwand	4
Implementierung/Durchführung.....	5
Testbericht.....	5
GitHub-Link.....	5

Aufgabenstellung

JDBC: RÜCKWÄRTSSALTO

Erstelle ein Java-Programm, dass Connection-Parameter und einen Datenbanknamen auf der Kommandozeile entgegennimmt und die Struktur der Datenbank als EER-Diagramm und Relationenmodell ausgibt (in Dateien geeigneten Formats, also z.B. PNG für das EER und TXT für das RM)

Verwende dazu u.A. das ResultSetMetaData-Interface, das Methoden zur Bestimmung von Metadaten zur Verfügung stellt.

Zum Zeichnen des EER-Diagramms kann eine beliebige Technik eingesetzt werden für die Java-Bibliotheken zur Verfügung stehen: Swing, HTML5, eine WebAPI, Externe Programme dürfen nur soweit verwendet werden, als sich diese plattformunabhängig auf gleiche Weise ohne Aufwand (sowohl technisch als auch lizenzrechtlich!) einfach nutzen lassen. (also z.B. ein Visio-File generieren ist nicht ok, SVG ist ok, da für alle Plattformen geeignete Werkzeuge zur Verfügung stehen)

Recherchiere dafür im Internet nach geeigneten Werkzeugen.

Die Extraktion der Metadaten aus der DB muss mit Java und JDBC erfolgen.

Im EER müssen zumindest vorhanden sein:

- korrekte Syntax nach Chen, MinMax oder IDEFIX
- alle Tabellen der Datenbank als Entitäten
- alle Datenfelder der Tabellen als Attribute
- Primärschlüssel der Datenbanken entsprechend gekennzeichnet
- Beziehungen zwischen den Tabellen inklusive Kardinalitäten soweit durch Fremdschlüssel nachvollziehbar. Sind mehrere Interpretationen möglich, so ist nur ein (beliebiger) Fall umzusetzen: 1:n, 1:n schwach, 1:1
- Kardinalitäten

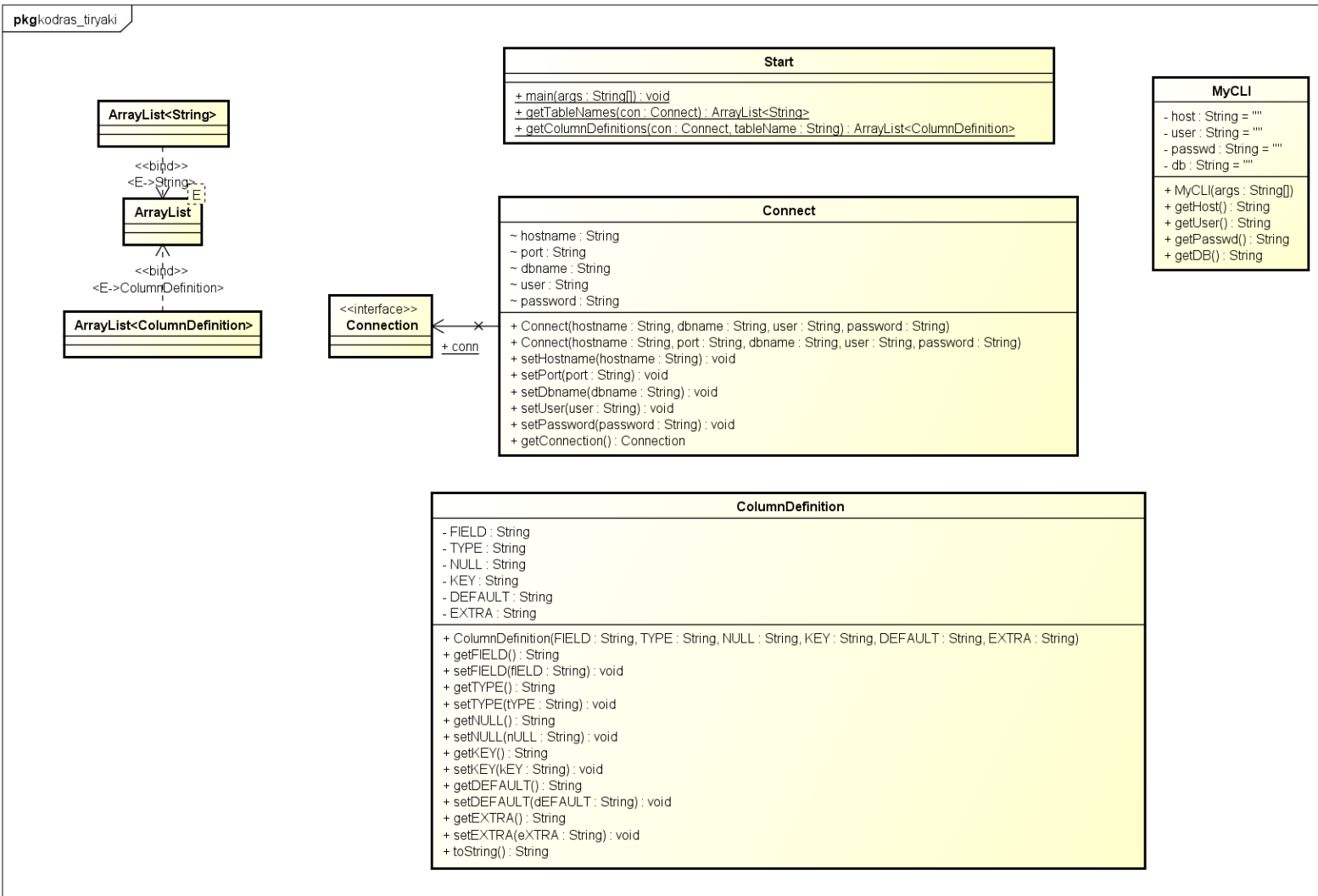
Fortgeschritten (auch einzelne Punkte davon für Bonuspunkte umsetzbar)

- Zusatzattribute wie UNIQUE oder NOT NULL werden beim Attributnamen dazugeschrieben, sofern diese nicht schon durch eine andere Darstellung ableitbar sind (1:1 resultiert ja in einem UNIQUE)
- optimierte Beziehungen z.B. zwei schwache Beziehungen zu einer m:n zusammenfassen (ev. mit Attributen)
- Erkennung von Sub/Supertyp-Beziehungen

Requirements-Analyse

- Ermitteln der Datenbanktypen die ausgelesen werden sollen
- Wählen der richtigen Treiber für die jeweilige Datenbank
- Wählen eines guten (Dokumentation + Erlernbarkeit + Einbindung) Tools zum Ausgeben eines ER-Diagramms
- Korrekte CLI-Übersetzung der notwendigen Parameter
-

Designüberlegung



Aufwand

Schätzung & Aufteilung

Geschätzter Aufwand: 6h * 2 Personen = 12 Stunden

Aufgabe	Name
Designüberlegung, UML-Klassendiagramm	Kodras, Tiryaki
Connection	Kodras
Testen	Kodras, Tiryaki
Protokoll fertigstellen	Kodras, Tiryaki

Reeller Aufwand

Aufgabe	Name	Zeit (min)

Aufwand Kodras:

Aufwand Tiryaki:

Gesamtaufwand:

Implementierung/Durchführung

Testbericht

GitHub-Link

<https://github.com/dkodras01-tgm/Rueckwaertssalto.git>