

Entwurf: Configurations etc.

Adrian

12. Juni 2018

# 1 Klasse: Configuration

Variablen:

- name: String
- delimiter: Character
- headerlines: Integer
- staticTimezone: TimeZone
- variableTimezone: Integer
- dateTime: List<StringColumn>
- streamData: List<StreamObservation>

Methoden:

- Konstruktor: Configuration(name)
- Einfache Getter und Setter für:
  - name
  - delimiter
  - headerlines
  - dateTime
  - streamData
- Angepasste getter und setter:
  - getStaticTimezone(): TimeZone
  - getVariableTimezone(): Integer
  - setStaticTimezone(TimeZone zone)
  - setVariableTimezone(Integer column)

Beschreibung Allgemein:

Diese Klasse stellt die Konfigurationen dar. Sie speichert alle notwendigen Daten und bietet getter und setter für einen vereinfachten Zugriff auf Daten bereit.

Beschreibung Variablen:

- `name`:  
Hier wird der Name, eine eindeutige Kennung der Konfiguration, gespeichert.
- `delimiter`:  
Falls eine CSV-Datei importiert werden soll, muss ein Trennsymbol für die Spalten gewählt werden.
- `headerlines`:  
In einem CSV-Dokument gibt es Kopfzeilen, welche für den Import irrelevant sind. Die Anzahl dieser Zeilen wird in dieser Variable gespeichert.
- `staticTimezone`:  
Falls die Zeitzone für alle Observations gleich ist, kann hier eine Zeitzone gewählt werden.
- `variableTimezone`:  
Hier wird die Spalte angegeben in der sich die Zeitzone befindet, falls die Zeitzone sich ändern kann
- `dateTime`:  
Hier wird eine Liste mit Datumsformaten mit einer zugehörigen Spalte gespeichert
- `streamData`  
In dieser Liste werden die Datastreams zusammen mit den Spalten gespeichert, aus welchen die Daten extrahiert werden.

#### Beschreibung Methoden:

- `getStaticTimezone(): TimeZone`  
Gibt die gewählte Zeitzone oder NULL, falls die `variableTimezone` oder keine `Timezone` gewählt wurde
- `getVariableTimezone(): Integer`  
Gibt die Spalte zurück, in der sich die Zeitzone befinden soll. Gibt -1 zurück, falls keine Spalte oder eine `StaticTimezone` gewählt wurde
- `setStaticTimezone(TimeZone zone)`  
Setzt die `StaticTimezone` auf `zone` und die `VariableTimezone` auf -1
- `setVariableTimezone(Integer column)`  
Setzt die `variableTimezone` auf `column` und `staticTimezone` auf NULL

## 2 Klasse: StringColumn

Variablen:

- string: String
- column: Integer

## 3 Klasse: StreamObservation

Variablen:

- dsID: String
- observation: List<Integer>

## 4 ConfigManagement

Variablen:

- Keine Variablen auf die der Nutzer Einfluss haben könnte.  
Die Speicherpfade sind final und nicht einsehbar.

Methoden:

- loadConfig(String id): Config (JSON)
- saveConfig(Config (JSON))
- static: ConvertToJava(Config (JSON)): Configuration
- listAll(): List<String>
- getServer(): URL

Beschreibung Allgemein:

Diese Klasse ist zuständig für die Verwaltung von Konfigurationen.

Beschreibung Methoden:

- `loadConfig(String id): Config (JSON)`  
Diese Methode bekommt den Namen einer Configuration übergeben, lädt die Configuration und gibt sie als JSON Objekt zurück.
- `saveConfig(Config (JSON))`  
Diese Methode bekommt als Eingabe eine Configuration als JSON Objekt und speichert dieses auf der Festplatte des Servers
- `ConvertToJava(Config (JSON)): Configuration` Um aus einer JSON Configuration eine Java Configuration zu erhalten übergibt man dieser Funktion ein JSON-Objekt und erhält als Ausgabe die gewünschte Java Configuration.
- `listAll()`  
Diese Funktion gibt eine Liste aller Konfigurationen, dargestellt als ihre Namen(String), zurück.
- `getServer(): URL`  
Diese Methode gibt die URL des Zielservers zurück