Technische Handleiding

# 

# Introductie

Dit document is opgesteld voor de technische staf van () en mogelijke studenten die dit project verder gaan ontwikkelen. Het document heeft als doel om de werkwijze en methodieken toe te lichten, zodat een begrip van de structuur van het project kan worden gevormd. Het document is opgedeeld in verschillende secties waar in elke sectie een deel van de structuur en de onderliggende connecties worden uitgelicht. De secties staan in chronologische volgorde van beginpunt tot eindpunt.

# Inhoudsopgave

[**Introductie**](#_qi0du4af8eny) **2**

[**Inhoudsopgave**](#_vfoco98qsapz) **3**

[**Data bronnen**](#_hs5xtn4fzg37) **4**

[**Python**](#_ss348ob2wmlf) **5**

[Cleaning](#_h9bs9avu0i0) 6

[Kolomnamen](#_9qia2ai4qbtn) 6

[Spelersnamen](#_77u9au4eehve) 6

[Scores](#_sy82i8ek9hv4) 6

[Tekst naar nummers](#_pmxl6uv0nisk) 6

[Python to database](#_owp5qfvjihyc) 7

[**SQL**](#_i1larm43t552) **8**

[Database](#_80zetibxduzu) 8

[Hoofdtabel](#_755h8o58eiwv) 8

[Log Tabel](#_d0dtlbj1fkrr) 8

[Updatedata tabel](#_6ins69ddwefz) 8

[Procedures](#_movcumltsiv1) 8

[Update](#_7dp50d5l5ymw) 8

[Insert](#_dr00yyqata4c) 9

[Add](#_496b7ih66l90) 9

[Logging](#_ljsiq7k4qdzx) 9

[**Power BI**](#_ifnt9teirjvh) **10**

[Data](#_k1fz6cdg428x) 10

[Berekeningen](#_il4yaksvwkte) 10

[Grafieken](#_2ua5isr6whcm) 10

[Slicers](#_ssj0vmamxjva) 10

# 

# Data bronnen

Vanuit de KNLTB is een Excel (.xlsx) workbook aangeleverd. In dit workbook wordt data verzameld uit onder andere Dartfish. Vervolgens wordt in dit workbook statistische analyses gedaan aan de hand van verscheidene filters.

Tijdens dit project worden deze analyses uitgevoerd door PowerBI, dit zorgt ervoor dat alleen de sheet ‘DATA’ wordt gebruikt voor verdere verwerking.

De ‘DATA’ sheet wordt geëxporteerd naar een .csv bestand en vervolgens via Python ingeladen.

# 

# Python

Python is een programmeertaal die veel wordt gebruikt bij Data Science en Data Engineering. Er zijn vele libraries die het makkelijk maken om data te verzamelen, verwerken en op te slaan naar verschillende databases.

De libraries die zijn gebruikt in dit project zijn:

**Pandas**

*‘Pandas is a fast, powerful, flexible and easy to use open source data analysis and manipulation tool, built on top of the Python programming language.’*

Pandas wordt in dit project gebruikt voor het maken van zogenaamde DataFrames vanuit het .csv bestand. Deze DataFrames zorgen ervoor dat de data makkelijk kan worden aangepast voor gebruik.

Meer informatie kunt u vinden in de [Pandas documentatie](https://pandas.pydata.org/docs/)

**Numpy**

*‘The fundamental package for scientific computing with Python.’*

Numpy wordt in deze context vooral gebruikt om missende gegevens in een dataset te vinden.

Meer informatie kunt u vinden in de [Numpy documentatie](https://numpy.org/doc/)

**SQLAlchemy**

*‘SQLAlchemy is the Python SQL toolkit and Object Relational Mapper that gives application developers the full power and flexibility of SQL. It provides a full suite of well known enterprise-level persistence patterns, designed for efficient and high-performing database access, adapted into a simple and Pythonic domain language*.*’*

SQLAlchemy wordt in de context van het project gebruikt om een connectie te leggen met een database en hier onze data naar toe te sturen.

In het project wordt een MySQL Database gehost op HvA domein: oege.ie.hva.nl

Meer informatie kunt u vinden in de [SQLAlchemy documentatie](https://docs.sqlalchemy.org/en/14/)

## 

## Cleaning

Het ‘cleanen’ van de data gebeurd nadat de data vanuit CSV naar een pandas dataframe omgezet is, deze DataFrames zijn ervoor ontwikkeld om snel en efficiënt met grote hoeveelheden te werken.

### Kolomnamen

De kolom namen worden genormaliseerd voor de database, waarbij de hoofdletters vervangen worden door kleine letters, de spaties vervangen worden door lage streepjes en de lege ruimtes verwijderd worden.

### Spelersnamen

De namen van de spelers zijn soms verkeerd gespeld, missen één letter of hebber er één te veel, om dit op te lossen is er een “sequency matcher” gebruikt. Deze sequency matcher gebruikt een richtlijn waarbij een naam die voor 90% hetzelfde is als een andere naam omgezet wordt naar de andere naam.

### Scores

Om de scores volledig bruikbaar te maken in Power BI was het nodig om de scores op te splitsen, hierbij werden de voor en tegen standen in aparte kolommen opgeslagen met de passende naamgeving erbij. Om dit voor elkaar te krijgen moest er eerst consistentie komen in de notering van de scores, hierbij moesten er voor alle 0 waardes een 0:0 komen en voor de waardes waarbij scores opgeslagen werden met een ‘-’, werden ze vervangen met een ‘:’. Als deze waardes opgeschoont waren, werden ze gesplitst in voor en tegen score kolommen.

### Tekst naar nummers

Het excel bestand heeft veel nummers opgeslagen als teksten, ook wordt er regelmatig een extra icoon gebruikt om aan te geven dat het getal groter dan een specifiek aantal is. Om te zorgen dat er gerekend kan worden met de getallen worden de extra icoontjes verwijderd en alle teksten met getallen omgezet naar getallen.

## 

## Python to database

SQL Alchemy is een client die het makkelijk maakt om een connectie te maken met een database.

## 

# SQL

## Database

De database is onderverdeeld in drie losse tabellen; de hoofdtabel waar alle data in wordt opgeslagen van de tennissers, de log tabel die bijhoudt wanneer de hoofdtabel wordt geüpdate en de hoeveelheid die waardes er in de hoofdtabel zitten en de update tabel waar de nieuwe data naar wordt gestuurd voordat het verwerkt wordt.

### Hoofdtabel

De hoofdtabel bestaat uit alle berekeningen en waardes die in het data tabblad van excel zitten, deze waardes worden opgeslagen geïdentificeerd aan de hand van de naam en positie van de rijen. Daarnaast wordt er aan de hand van drie sleutel waardes geïdentificeerd of de rijen actieve waardes zijn, wanneer ze geüpload en wanneer ze gecreëerd zijn, hierdoor worden oude records die geupdate worden niet verwijderd en kan de rest van de applicatie zien welke niet waardes niet gebruikt moeten worden.

### Log Tabel

In de log tabel wordt er alleen bijgehouden wanneer de hoofdtabel geüpdate is en hoeveel waardes de hoofdtabel heeft na de update, hierdoor is er bij te houden of er ergens iets fout gegaan is en wat er ongeveer gebeurd is.

### Updatedata tabel

De update tabel is een lege tabel die alleen bestaat voor het vasthouden van de nieuwe data die de hoofdtabel in moet, deze tabel wordt voor elke update geleegd en opnieuw gevuld.

## Procedures

In de MySQL database staan een aantal gemaakte procedures die nodig zijn om de data volledig in te kunnen voegen in de hoofdtabel van de database, deze procedures kunnen van buitenaf aangeroepen worden als de nieuwe data in update tabel zit.

### Update

De update procedure wordt gebruikt om in de update tabel voor elke rij te kijken of het al in de hoofdtabel zit, als het er al in zit worden alle waardes van beide rijen met elkaar vergeleken. Indien één van de waardes anders is wordt de insert procedure hieronder aangeroepen en worden de actief en update kolommen gewijzigd, als het hetzelfde is wordt de rij in de update tabel overgeslagen en wordt het proces opnieuw gedaan totdat de gehele update tabel bekeken is. Als dit proces compleet is worden de add en logging procedures aangeroepen, waarna de procedure compleet is.

### Insert

De insert procedure wordt gebruikt om een nieuwe rij toe te voegen in de hoofdtabel, bij deze procedure wordt een nieuwe rij toegevoegd in de hoofdtabel voor een rij die gewijzigd is.

### Add

De add procedure voegt alle rijen toe uit de update tabel die nog niet bestaan in de hoofdtabel, hierbij wordt er voor elke rij toegevoegd dat het actief is en wanneer het gecreëerd is.

### Logging

Bij de logging procedure wordt in de log tabel een nieuwe rij aangemaakt elke keer dat de update procedure wordt aangeroepen, het voegt in de log tabel een rij toe met de totale hoeveelheid rijen die in de hoofdtabel zitten en de datum en tijd dat het is aangepast.

# 

# Power BI

### Data

Binnen Power BI is allereerst een connectie gemaakt met een SQL server met daarop de relevante data. Deze data is geïmporteerd de data die wordt gebruikt kan binnen Power BI gezien worden door in de linkerbalk “data” te selecteren.

### Berekeningen

De grafieken die worden getoond zijn opgebouwd aan de hand van enkele berekeningen deze berekeningen zijn terug te vinden in de rechterbalk. Om inzicht te krijgen in hoe een berekeningen is opgebouwd kan er gedubbelklikt worden op de berekeningen. Daar wordt door het gebruik van namen getoond wat de berekening doet.

### Grafieken

De grafieken zijn opgebouwd vanuit de eerder genoemde berekeningen in samenwerking met filters. Waar mogelijk zijn berekeningen vermeden om de complexiteit zo laag mogelijk te houden. Door op een grafiek te drukken kan gezien worden welke waardes per grafiek worden gefilterd om tot de gewenste waardes te komen. Deze filters zijn specifiek per grafiek.

### Slicers

Aan de onderkant van iedere pagina is ook een set met slicers gemaakt (voor gemak voor de eindgebruiker noemen we deze op de front end filters). Deze slicers kunnen snel en gemakkelijk een specifiek deel van de data filteren. Deze slicers bevatten geen berekende waardes en zijn aanwezig om de gebruiker te helpen om de precieze data waar hij/zij op zoek naar is te vinden.