**De appels**

**Zaalvoetbal project**

**Database onderzoek**



Groepsleden: Jeroen Stamkot, Marco Havermans, Teun Aarts en Damian Leijten

RIO4-APO1A

Inhoud

[Inleiding 3](#_Toc388277931)

[Column/Field 3](#_Toc388277932)

[Row/Record 4](#_Toc388277933)

[Database Management System 4](#_Toc388277934)

[Welke zijn er? 4](#_Toc388277935)

[Welke gebruikt Visual Studio? 4](#_Toc388277936)

[DataType 5](#_Toc388277937)

[Primary Key 5](#_Toc388277938)

[Foreign Key 6](#_Toc388277939)

[Naamgevingsconventie tabellen 6](#_Toc388277940)

[Naamgevingsconventie rijen 6](#_Toc388277941)

[Welke manieren van aanleveren van data zijn er? 7](#_Toc388277942)

[Welke database gaat onze voorkeur naar uit? 8](#_Toc388277943)

[Datadictionary 8](#_Toc388277944)

[Koppeling applicatie 🡪 database 9](#_Toc388277945)

[Handtekeningen groepsleden: 10](#_Toc388277946)

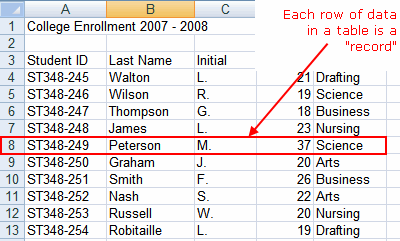
# Inleiding

Dit document is gemaakt voor Marco Havermans, Teun Aarts, Damian Leijten en Jeroen Stamkot. Het is gemaakt het project “Fifa”.

# Column/Field



# Row/Record



# Database Management System

## Welke zijn er?



## Wat gebruikt Visual Studio?

.SDF ADO.NET EF 5.x DbContext Generator Service-based Database

.MDF Dataset LINQ to SQL Classes Local Database

XML File XML Schema XSLT File

# DataType

Een datatype, ook wel gegevenstype genoemd, is in de [informatica](http://nl.wikipedia.org/wiki/Informatica) een specifiek soort gegevens, zoals [integers](http://nl.wikipedia.org/wiki/Integer_(informatica)), [booleans](http://nl.wikipedia.org/wiki/Boolean), [reals](http://nl.wikipedia.org/wiki/Zwevendekommagetal), [karakters](http://nl.wikipedia.org/wiki/Karakter_(informatica)), [strings](http://nl.wikipedia.org/wiki/String_(informatica)). In een [programmeertaal](http://nl.wikipedia.org/wiki/Programmeertaal) wordt met iedere [variabele](http://nl.wikipedia.org/wiki/Variabele_(informatica)), en meer in het algemeen met iedere [expressie](http://nl.wikipedia.org/wiki/Expressie_(programmeren)), een datatype geassocieerd. Dit datatype bepaalt welke waarden de variabele kan aannemen, hoe deze waarden in het geheugen worden opgeslagen en welke bewerkingen op de variabele of de expressie uitgevoerd kunnen worden.

# Primary Key

De primary key, ofwel primaire sleutel, identificeert ieder record in een tabel. De primary key zorgt ervoor dat ieder veld dat ingevoerd wordt niet leeg mag zijn en dat het een unieke waarde heeft. Per tabel kan er maximaal één primary key gedefinieerd worden.

CREATE TABLE *tabelnaam*   
(  
    *kolomnaam1* datatype,  
    *kolomnaam2* datatype,  
    *CONSTRAINT* Constraintnaam *PRIMARY KEY* (kolomnaam) )

Voorbeeld

In onderstaand voorbeeld wordt een tabel aangemaakt, waarin persoonsgegevens worden opgeslagen. In deze tabel wordt van de persoon een uniek id-nummer opgeslagen, zijn voor- en achternaam, woonplaats en het geslacht. Het id-nummer is in dit geval de primary key, omdat ieder veld hiermee geidentificeerd kan worden.

CREATE TABLE *persoon*   
(  
    *id* int,  
    *Voornaam* varchar(255) NOT NULL,  
    *Achternaam* varchar(255) NOT NULL,  
    *Woonplaats* varchar(255),  
    *Geslacht* varchar(1),  
    *CONSTRAINT* PK\_persoon *PRIMARY KEY* (id)  
)

# Foreign Key

Foreign Keys (oftewel verwijzende sleutels), ook wel afgekort tot FK's, duiden relaties aan tussen twee tabellen in een database. Een waarde in een tabel zal verbonden worden met een andere tabel door twee waardes altijd overeen te laten komen.

# Naamgevingsconventie tabellen

1. Zet als eerste “tbl”. (kleine letters)
2. Begin elk apart woord beginnen met een hoofdletter
3. Zo kort mogelijk maar wel krachtig
4. Naam van tabellen in het Engels.

# Naamgevingsconventie rijen

1. Begin elk apart woord met een hoofdletter
2. Zo kort mogelijk maar wel krachtig
3. Naam van de rijen in het Engels.

# Welke manieren van aanleveren van data zijn er?

|  |  |  |
| --- | --- | --- |
| **Naam:** | **Voordelen:** | **Nadelen:** |
| Csv | Het voordeel van een csv-encoding bestand is dat het eenvoudig te lezen / bewerken is met bijvoorbeeld de gratis ‘csvedit‘ software. De csv-encoding is ontwikkeld, omdat niet in alle gevallen de rijkdom van het gml-formaat noodzakelijk is. Een nadeel van de csv-encoding is dat het noodzakelijk is meerdere bestanden uit te wisselen.  Het csv encoding voor UM Aquo Metingen kent hiervoor de beschrijving van een drietal verschillende csv bestanden:  Uitwisselen van Meetpunten  Uitwisselen van Monsters  Uitwisselen van Tijdwaarden. | Nadeel van deze systematiek is dat numerieke waarden altijd omgezet moeten worden naar alfanumerieke waarden, wat uiteraard tot nauwkeurigheidsverlies of zelfs afrondingsproblemen kan leiden. Een ander probleem is dat alle computersystemen op één na een bepaald teken gebruiken als regeleinde, terwijl dat ene afwijkende systeem (MsWindows uiteraard) twee tekens voor een regeleinde gebruikt. Dat kan tot conversieverschillen leiden. Moderne applicaties kunnen deze conversie vaak automatisch doen, maar dat kan niet a priori gegarandeerd worden. |
| Json | Het voordeel is dat het gestructureerder is dan bij XML, kleiner en dus sneller te lezen is. Een ander voordeel is dat JSON door de gestructureerde opbouw OOP is. Dit betekent dat het Object georiënteerd is. Hierdoor sluit JSON naadloos aan bij talen als Java en C#. | Een nadeel van JSON is dat er (nog) geen complexe document validatie mogelijk is zoals bij XML, Hierdoor is het niet eenvoudig om te controleren of een JSON document correct is. |
| SQL | Snelheid en flexibiliteit. Deze taal maakt het onder andere mogelijk om zeer geavanceerde gegevensselecties te maken uit de database. | Kosten, beperkte compatibiliteit, Limited Functionaliteit. |
| XML | Eenvoudige syntax, XML is een universeel mechanisme om informatie te coderen en te manipuleren, waarbij vorm, inhoud en structuur van elkaar gescheiden kunnen blijven. Zowel bij multimedia als bij data. Dankzij dergelijke scheiding zijn uitwisseling en hergebruik van informatie veel goedkoper en beter te automatiseren. Het traditionele onderscheid tussen gestructureerde data en ongestructureerde tekst of multimedia documenten vervaagt grotendeels. XML biedt nieuwe en krachtige mogelijkheden tot significante verbeteringen in dienstverlening en bedrijfsvoering. | Een nadeel van een Xml database is dat alle Xml documenten geladen worden voordat het getoont kan worden, een deel inchecken is dus ook niet mogelijk. Xml heeft geen ondersteuning voor gebruikers beveiliging zoals in relationele databases. Het kan geen limitaties van toevoegen verwijderen of veranderen aan Xml documenten toepassen. |
| YAML | YAML is ontwikkeld vanuit oogpunt van menselijke leesbaarheid en uitbreidbaarheid. YAML is dan ook goed voor mensen leesbaar, mede door de verplichte witruimtes in regels om structuur aan te geven. YAML ondersteund default object referenties en relationele structuren. Dit maakt het mogelijk om in YAML net als in XML eenvoudig cyclische datastructuren met diepe hiërarchie vast te leggen. | Als het een grotere database word dan word het heel snel onoverzichtelijk, ingewikkelde indeling die je moet leren. |

## Welke database gaat onze voorkeur naar uit?

We kiezen voor SQL want SQL is sneller en flexibeler dan alle andere DataBase’s. Ook kiezen wij voor SQL omdat Media-development (waarvan wij de data aangeleverd krijgen) MySQL gebruiken. Zij zetten het voor ons om naar SQL zodat wij het heel makkelijk kunnen importeren n ons programma.

# Datadictionary

De datadictionary is een overzicht van alle tabellen en bijbehorende velden van een database.

Dus een datadictionary is een de plek waar alle velden worden klaar gemaakt zodat er data ingezet kan worden. Daar kun je dus de naam van de “Row” ingeven en de “Data Type”

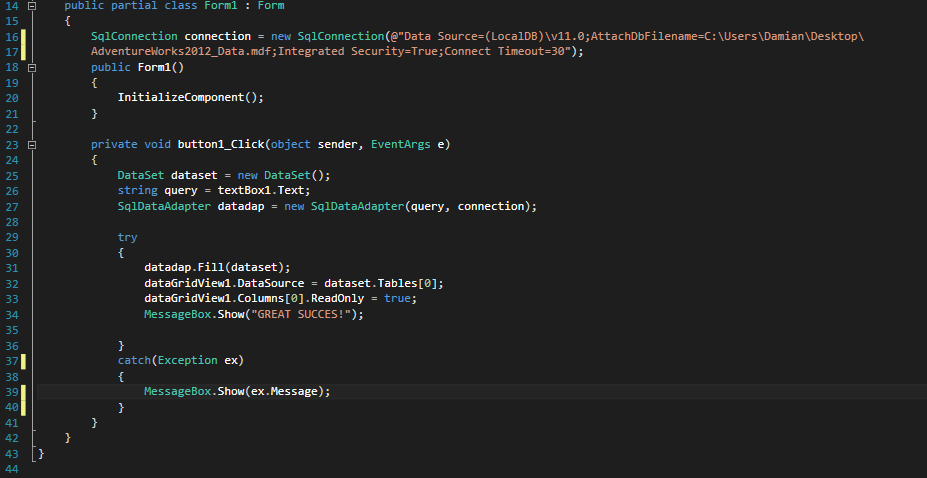
Gebrukersinfo:

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Column Name** | **Data Type** | **Length** | **Allow Nulls** | **Unique** | **Primary Key** |
| Username | Nvarchar | 20 | No | Yes | Yes |
| Password | Nvarchar | 20 | No | No | No |
| Name | Nvarchar | 25 | No | No | No |
| Surname | Nvarchar | 15 | No | No | No |
| Class | Nvarchar | 10 | No | No | No |

Toto info:

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Column Name** | **Data Type** | **Length** | **Allow Nulls** | **Unique** | **Primary Key** |
| Team | Nvarchar | 20 | No | Yes | Yes |
| Player | Nvarchar | 30 | No | No | No |
| Points | int | 3 | Yes | No | No |
| Topscoorder | Nvarchar | 30 | No | Yes | No |
| Result | int | 2 | Yes | No | No |
| Goal Balance | Int | 2 | Yes | No | No |

# Koppeling applicatie 🡪 database



# Handtekeningen groepsleden:

Marco Havermans: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Teun Aarts: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Jeroen Stamkot: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Damian Leijten: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_