

Tieme Woldman

DE SQL START-UP:

Data analyse
voor iedereen

VANDUUREN
MEDIA

De SQL-start-up

Tiene Woldman

VANDUUREN
MEDIA

ISBN: 978-94-6356-392-5

NUR: 995

Trefw.: databases, SQL

Omslag: Bruggn Design, Purmerend

Opmaak: Van Duuren Media, Culemborg/Barcelona

Druk: Printforce, Culemborg

Eerste oplage: maart 2025

Copyright © 2025 Van Duuren Media B.V.

Alle rechten voorbehouden. Niets uit deze uitgave mag worden verveelvoudigd, opgeslagen in een geautomatiseerd gegevensbestand, of openbaar gemaakt, in enige vorm of op enige wijze, hetzij elektronisch, mechanisch, door fotokopieën, opnamen, of enige andere manier, zonder voorafgaande toestemming van de uitgever.

Voorzover het maken van kopieën uit deze uitgave is toegestaan op grond van artikel 16B Auteurswet 1912 j° het Besluit van 20 juni 1974, St.b. 351, zoals gewijzigd bij Besluit van 23 augustus 1985, St.b. 471 en artikel 17 Auteurswet 1912, dient men de daarvoor wettelijk verschuldigde vergoedingen te voldoen aan de Stichting Reprorecht. Voor het overnemen van gedeelte(n) uit deze uitgave in bloemlezingen, readers en andere compilatie- of andere werken (artikel 16 Auteurswet 1912), in welke vorm dan ook, dient men zich tot de uitgever te wenden.

Ondanks alle aan de samenstelling van dit boek bestede zorg kan noch de redactie, noch de auteur, noch de uitgever aansprakelijkheid aanvaarden voor schade die het gevolg is van enige fout in deze uitgave.

Delen van dit boek zijn samengesteld met behulp van ChatGPT.

INHOUD

I Kennismaken met SQL en data-analyse	I
Wat is SQL?	2
SQL-opdrachten om informatie uit data te halen	3
De opbouw van een SQL-opdracht	4
SQL werkt met tabeldata uit een database	6
Waarom het nuttig is om SQL te leren	8
Wat is data-analyse?	8
Data-analyse is een proces om inzicht uit data te halen	9
Processtappen van data-analyse	10
Wat data-analyse je oplevert	10
Voorbeelden van wat je met data-analyse kunt	11
Voor wie is dit boek?	12
Geen voorkennis nodig	12
Wat is de inhoud van dit boek? Wat leer je?	13
Basiskennis over databases	13
SQL-tools installeren en data analyseren met SQL	13
Ontbrekende en foute data tackelen	13
Voorbeeldbestanden downloaden	14
Samenvatting	14
Vragen	14
Praktijkoefeningen	15

2	Wat je moet weten van relationele databases	17
	De essentie van gestructureerde data	17
	Ongestructureerde data	18
	Gestructureerde data	18
	De basis van gestructureerde data: de tabel	19
	Opzet van een tabel	20
	Termen	20
	Verschillende datatypen in een tabel	23
	Numeriek	24
	Tekst	24
	Datum	25
	Boolean	26
	NULL als bijzondere veldwaarde	26
	Bindmiddel van gestructureerde data: de relaties tussen tabellen	27
	Primaire sleutel	27
	Tabelrelaties via de vreemde en primaire sleutel	28
	Soorten relaties tussen tabellen	31
	Veel-op-veelrelatie	32
	Container voor gestructureerde data: de relationele database	33
	RDBMS als container van de relationele databases	33
	Samenvatting	34
	Vragen	35
	Praktijkoefeningen	36
3	SQL-tools en database installeren	39
	Database downloaden en installeren	39
	Alpha-database downloaden en installeren	40
	SQL-tools en RDBMS installeren	40
	DBeaver installeren	41
	Kennismaken met DBeaver	43
	Venster Database Navigator	43
	SQL-editor	43
	Relationele database verkennen in DBeaver	44
	Eerst de databaseconnectie maken	45
	Informatievraag als uitgangspunt voor het verkennen	46
	Onderzoeken welke databases beschikbaar zijn	47
	Op verkenning in een database	47
	Samenvatting	56
	Vragen	57
	Praktijkoefeningen	57

4	Basisinzicht verkrijgen met query's	59
	Informatievraag uitwerken	59
	Stel deze vragen om een informatievraag helder en compleet te maken	60
	Basisquery maken	63
	Begin een query altijd met een uitleg in commentaarregels	64
	SELECT...FROM als basis	67
	Sorteren via ORDER BY	72
	Veldwaarde samenstellen uit andere velden via concatenate	74
	Aangepaste veld- of tabelomschrijving geven via een alias	76
	Query opslaan voor later gebruik	79
	Queryresultaat exporteren	80
	Exporteren als HTML-pagina in een browser	80
	Exporteren in CSV-formaat	82
	Samenvatting	83
	Vragen	83
	Praktijkoefeningen	84
5	Je inzicht focussen met filters	87
	Vergelijken en bewerken met operatoren	87
	Vergelijkingsoperatoren	88
	Logische operatoren	89
	LIKE-, IN- en BETWEEN-operatoren	91
	IS NULL en IS NOT NULL	91
	Rekenkundige operatoren	92
	Tekenreeksoperator	93
	Gegevens filteren	93
	Filteren met het WHERE-onderdeel	93
	Verfijnen met IN en BETWEEN voor series	96
	Verfijnen met LIKE voor wildcardvergelijkingen	98
	Unieke waarden filteren met DISTINCT	99
	Resultaat beperken tot de top- of laagste waarden	100
	Samenvatting	102
	Vragen	102
	Praktijkoefeningen	103

6	Je inzicht verbreden via meerdere tabellen	105
	Gegevens uit meerdere tabellen verwerken met joins	105
	Tabellen koppelen via het JOIN-onderdeel	106
	Koppelen met meer dan twee tabellen	109
	Verschillende soorten joins	111
	INNER JOIN	111
	LEFT JOIN	113
	RIGHT JOIN	114
	SELF JOIN	115
	Samenvatting	116
	Vragen	116
	Praktijkoefeningen	117
7	Je inzicht verfijnen met functies en groeperen	119
	SQL-functies voor bewerken en berekenen van gegevens	119
	Alle waarden van een tabelveld als input	121
	Selectie van veldwaarden als input	121
	Functies combineren (nesten)	122
	Functies in verschillende SQL-onderdelen	122
	Soorten functies	122
	Tekstfuncties	123
	Datumfuncties	128
	Numerieke functies	133
	Aggregatefuncties	133
	Aggregatefuncties en groeperen	137
	Samenvatting	142
	Vragen	142
	Praktijkoefeningen	143
8	Onvolledige en onjuiste gegevens oplappen	145
	Gegevens oplappen	145
	Ontbrekende en onjuiste gegevens door andere waarden vervangen	146
	Ontbrekende en onjuiste gegevens reconstrueren	151
	Gegevens naar een ander datatype omzetten	153
	Samenvatting	155
	Vragen	155
	Praktijkoefeningen	156

9	Je inzicht verdiepen met subquery's	157
	Subquery's om de verwerking op te delen	157
	Verwerkingsvolgorde	158
	In welke SQL-onderdelen kun je subquery's gebruiken?	159
	Subquery's voor verschillende situaties	160
	Berekeningen en bewerkingen opsplitsen	161
	Berekeningen en bewerkingen combineren	162
	Dynamisch filteren	165
	Speciale manieren van groeperen	167
	Geavanceerd voorbeeld	169
	Analyseren en debuggen	170
	Samenvatting	171
	Vragen	172
	Praktijkoefeningen	173
10	Gegevens wijzigen en verwijderen met SQL	175
	Datamanipulatie met SQL	175
	Rechten om gegevens te manipuleren	176
	Gegevens wijzigen	176
	Veilige werkwijze	177
	Plaatsnamen en provincies wijzigen	179
	Verwijderen eventuele omlapcode	181
	Gegevens verwijderen	182
	Volg eerst altijd een van de veilige werkwijzen	182
	Samenvatting	183
	Afsluiting en volgende stappen	183
	Vragen	184
	Praktijkoefeningen	184
	Index	185

KENNISMAKEN MET SQL EN DATA- ANALYSE

Vandaag de dag is data een van de waardevolste bezittingen van bedrijven en organisaties. Want stel dat de winst van een bedrijf terugloopt. Dan kun je uit data zoals verkoopgegevens informatie halen over welke producten en diensten winstgevend zijn en op welke je verlies lijdt. Om vervolgens met deze informatie te besluiten welke zaken je afstoot en welke je juist meer aandacht geeft. Met als resultaat een verbeterde winst waarmee je mogelijk een faillissement of ingrijpende reorganisatie voorkomt.

De bijzondere waarde van data geldt overigens niet alleen voor bedrijven, maar voor alle organisaties. Want op dezelfde manier kan bijvoorbeeld een ziekenhuis informatie uit patiëntdata halen waarmee men behandelingen verbetert. Of kan een goede doelen organisatie beter bepalen waar en op welke manier men hulp verleent.

Zo is data de essentiële grondstof voor informatie waarmee organisaties meer inzicht krijgen om betere beslissingen nemen en processen te verbeteren. Daarbij geldt dat om informatie uit data te halen, je net als bij andere grondstoffen een proces nodig hebt om het eindresultaat te realiseren. Om informatie van data te maken, gebruik je het proces data-analyse waarbij databases en de opvraagtaal SQL een belangrijke rol spelen.

Dit boek behandelt de stappen en onderdelen om met SQL data te analyseren voor inzichten waarmee je beslissingen en processen optimaliseert. SQL is een krachtige, maar gelukkig eenvoudige, opvraagtaal voor data die snel antwoord geeft op allerlei informatievraagstukken. Aan het eind van dit boek beheers je SQL en kun je met SQL data analyseren om de gewenste informatie uit data halen.

Wat is SQL?

SQL is een opvraagtaal om informatie uit data of gegevens te halen voor inzicht in je bedrijf of organisatie. Feitelijk is SQL een programmeertaal, maar omdat de nadruk vooral op het opvragen van informatie ligt, spreken we vaak van SQL als opvraagtaal. SQL is de afkorting van *Structured Query Language* en SQL spreek je uit als *sequel* of *S-Q-L*. Het is gebaseerd op het Engels en daardoor eenvoudig te lezen en begrijpen. Ook als je geen programmeur of data-analist bent.

Programmeer- en opvraagtaalen zoals SQL bestaan uit commando's om bepaalde taken uit te voeren. In SQL heb je bijvoorbeeld commando's om gegevens te selecteren en te sorteren. Deze SQL-commando's zijn gebaseerd op het Engels en bestaan uit eenvoudige woorden als **SELECT** om gegevens te selecteren en **ORDER BY** om te sorteren. Hierdoor kan iedereen gemakkelijk met SQL werken en kost het weinig tijd om te leren.

SQL bestaat al langer dan vijftig jaar, maar ondanks die leeftijd staat SQL ook nu nog in de top 10 van de belangrijkste en meestgebruikte programmeertaalen. En dat is begrijpelijk als je de voordelen en sterke punten van SQL bekijkt:

- SQL is een krachtige opvraagtaal waarmee iedereen waardevolle informatie en inzichten uit data kan halen. Dus met SQL kan iedereen data analyseren.
- SQL is eenvoudig wat betreft opzet en daardoor snel te leren en te gebruiken.
- SQL is efficiënte taal waardoor je ook grote hoeveelheden data snel tot informatie kunt verwerken.
- SQL is een internationale standaard waardoor je het met verschillende databases kunt gebruiken.

SQL-opdrachten om informatie uit data te halen

Dit is een indrukwekkende lijst met voordelen. Om te bewijzen dat SQL werkelijk eenvoudig is, kijken we hier alvast even naar een voorbeeld. Straks gaan we wat dieper in op tabellen, maar SQL haalt informatie uit tabellen zoals dit voorbeeld van een tabel met klantgegevens:

123 KlantId	A:Z Voornaam	A:Z Achternaam	A:Z Bedrijf	A:Z Plaats	A:Z Email
1	Luís	Gonçalves	Embraer	São Paulo	luisg@embraer.com.br
2	Leonie	Köhler	[NULL]	Stuttgart	leonekohler@surfeu.de
3	François	Tremblay	[NULL]	Montréal	ftremblay@gmail.com
4	Bjørn	Hansen	[NULL]	Oslo	bjorn.hansen@yahoo.no
5	František	Wichterlová	JetBrains s.r.o.	Praag	frantisekw@jetbrains.com
6	Helena	Holý	[NULL]	Praag	hholy@gmail.com
7	Astrid	Gruber	[NULL]	Wenen	astrid.gruber@apple.at
8	Daan	Peeters	[NULL]	Brussel	daan_peeters@apple.be
9	Kara	Nielsen	[NULL]	Kopenhagen	kara.nielsen@jubii.dk
10	Eduardo	Martins	Woodstock Discos	São Paulo	eduardo@woodstock.com.br
11	Alexandre	Rocha	Banco do Brasil S.A.	São Paulo	alero@uol.com.br
12	Roberto	Almeida	Riotur	Rio de Janeiro	roberto.almeida@riotur.gov.br
13	Fernanda	Ramos	[NULL]	Brasília	fernadaramos4@uol.com.br
14	Mark	Phillips	Telus	Edmonton	mphilips12@shaw.ca
15	Jennifer	Peterson	Rogers Canada	Vancouver	jenniferp@rogers.ca
16	Frank	Harris	Google Inc.	Mountain View	fharris@google.com
17	Jack	Smith	Microsoft Corporation	Redmond	jacksmith@microsoft.com
18	Michelle	Brooks	[NULL]	New York	michelleb@aol.com
19	Tim	Goyer	Apple Inc.	Cupertino	tgoyer@apple.com
20	Dan	Miller	[NULL]	Mountain View	dmiller@comcast.com
21	Kathy	Chase	[NULL]	Reno	kachase@hotmail.com
22	Heather	Leacock	[NULL]	Orlando	hleacock@gmail.com
23	John	Gordon	[NULL]	Boston	johnngordon22@yahoo.com

Afbeelding 1.1: De tabel met klantdata.

Stel nu dat de tabel de naam `Klant` heeft en dat we een lijst van klanten uit Praag willen met hun voornaam, achternaam en e-mailadres. Dan is dit de opdracht die je daarvoor in SQL opstelt:

```
SELECT Voornaam, Achternaam, Email
FROM Klant
WHERE Plaats = 'Praag'
```

Later gaan we uitgebreid in op de onderdelen van deze opdracht. Voor dit moment gaat het erom dat we maar drie regels nodig hebben om met het `SELECT`-commando onze lijst samen te stellen. In de eerste regel geef je aan welke informatie je uit de tabel in je lijst wilt. In de

tweede regel specificeer je uit welke tabel de informatie komt en in de derde regel filter je alleen de klanten uit Praag.

Dit is alles wat je nodig hebt in SQL en is gemakkelijk te volgen, toch? De SQL-opdracht geeft deze informatie als resultaat:

A-Z Voornaam ▾	A-Z Achternaam ▾	A-Z Email ▾
František	Wichterlová	frantisekw@jetbrains.com
Helena	Holý	hholy@gmail.com

Afbeelding 1.2: Lijst met klanten uit Praag.

De opbouw van een SQL-opdracht

SQL-opdrachten hebben een vaste opbouw met herkenbare onderdelen. Laten we kijken naar de opbouw en onderdelen zodat je daar alvast vertrouwd mee raakt.

Opdrachten zijn hetzelfde als query's

Binnen SQL spreekt men vaak van een query in plaats van een opdracht terwijl men daar hetzelfde mee bedoelt. Het begrip query is echter meer ingeburgerd en kom je tegen in Engelstalige documentatie teksten over SQL en daarom gebruiken we vanaf hier de term query.

Query's in Excel

Het begrip query is niet exclusief voor SQL en ken je misschien al vanuit Excel. In Excel kun je namelijk met Microsoft Query externe gegevens ophalen. Verder heeft Excel ook het onderdeel Power Query waarmee je externe gegevens kunt ophalen bewerken.

Queryonderdelen

SQL-query's, zoals ons voorbeeld voor de lijst met klanten uit Praag, zijn opgebouwd uit onderdelen, in het Engels *clauses* genoemd. In het eerste query voorbeeld zijn **SELECT**, **FROM** en **WHERE** de onderdelen waarbij ieder onderdeel zijn eigen functie heeft:

- Het **SELECT**-onderdeel dat bepaalt welke informatie het resultaat bevat.

- Het `FROM`-onderdeel dat aangeeft uit welke tabel de informatie komt.
- Het `WHERE`-onderdeel dat het filter toepast.

In SQL staan de onderdelen van een query in een vaste volgorde. Eerst de `SELECT`, dan de `FROM` en daarna de `WHERE`. Later in dit boek leren we nog meer SQL-onderdelen, inclusief hun plek in SQL. Zoals het `ORDER BY`-onderdeel waarmee we gegevens kunnen sorteren.

Door de vaste volgorde van onderdelen kan de software die een query verwerkt, de query interpreteren en uitvoeren. En kunnen wij query's gemakkelijk lezen en begrijpen omdat we de onderdelen snel herkennen.

Overigens, de onderdelen mogen onder elkaar en naast elkaar staan. Dus deze indeling is correct:

```
SELECT Voornaam, Achternaam, Email  
FROM Klant  
WHERE Plaats = 'Praag'
```

Maar deze indeling is ook prima:

```
SELECT Voornaam, Achternaam, Email FROM Klant WHERE Plaats = 'Praag'
```

Zolang we de onderdelen maar zonder lege tussenregels in de juiste volgorde zetten blijven de werking en het resultaat van een query hetzelfde. Voor de leesbaarheid is het echter vaak beter om de onderdelen over meerdere regels te verdelen. Maar die keuze is aan jou.

Keywords

We weten nu dus dat een SQL-query uit onderdelen in een vaste volgorde bestaat. De onderdelen bestaan echter ook weer uit componenten en dat zijn de zogeheten keywords. Keywords zijn de speciale woorden in de SQL-taal waarmee je een query definieert. Voorbeelden van de keywords zijn `SELECT`, `FROM` en `WHERE` en daarmee bouw je een querybasis als dit voorbeeld:

```
SELECT  
FROM
```

Vervolgens bouw je het `SELECT`-keyword uit met de informatie die je wilt tonen en het `FROM`-keyword met de tabel waar de informatie vandaan komt.

Keywords in hoofdletters

Het is niet verplicht, maar om de keywords snel te herkennen in een query, typen we de keyword in hoofdletters.

SQL werkt met tabeldata uit een database

We zagen al dat SQL met data uit tabellen werkt, maar de tabellen staan niet op zich en zijn onderdeel van een database. Een database is een verzameling van gestructureerde data opdat je de data met bijvoorbeeld SQL kunt verwerken. En als de data in een database in tabellen is gestructureerd, spreken we van een relationele database omdat – zoals we later zullen zien – de tabellen aan elkaar gerelateerd zijn met verwijzingen. Dankzij deze verwijzingen kunnen we data uit verschillende tabellen combineren zodat we nog waardevollere informatie kunt samenstellen.

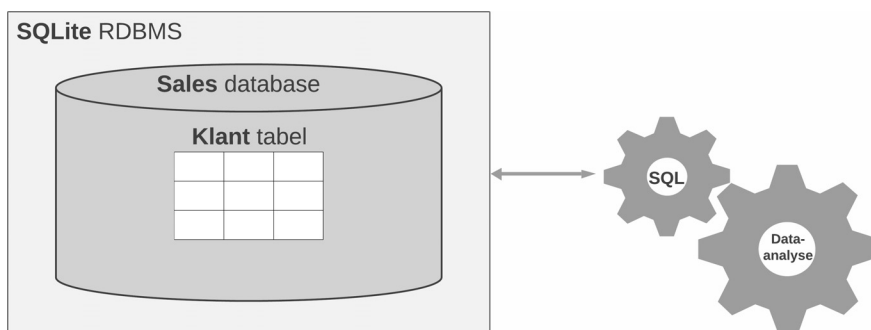
Tabellen en relaties

Dit klinkt misschien allemaal nog een beetje theoretisch, maar als we straks aan de slag gaan met tabellen en SQL dan zie je dat het heel logisch in elkaar zit en eenvoudig te volgen is. Voor dit moment is belangrijk dat SQL met een relationele database werkt waarin data in gerelateerde tabellen is opgeslagen. Met de toevoeging dat een tabel uit rijen en kolommen bestaat zoals in het voorbeeld van afbeelding 1.1.

Relationeel Database Management System (RDBMS)

Verder is nu nog handig om te weten dat een database speciale software nodig heeft om de data in de database mee te beheren en verwerken via SQL. De combinatie van de database en de beheer- en verwerkingssoftware noemen we het Relationeel Database Management Systeem (RDBMS). Voorbeelden van RDBMS'en zijn commerciële versies als Oracle en Microsoft SQL Server en opensourceversies als PostgreSQL, MySQL en SQLite.

Het RDBMS, een database, de tabellen en SQL komen als volgt samen:



Afbeelding 1.3: Het samenspel van RDBMS, database, tabel en SQL.

SQLite

In dit boek ga je werken met het SQLite RDBMS. SQLite is een open-source-RDBMS en beschikbaar voor Windows, macOS en Linux. Verder draait SQLite op hardware van een smartphone tot een server die je alleen voor databases gebruikt.

Als je een Android-telefoon hebt, werk je (zonder dat je het waarschijnlijk beseft) al met SQLite, omdat dit RDBMS standaard op alle Android-smartphones draait.

Overigens, met de SQL-kennis die je opdoet via SQLite, werk je net zo gemakkelijk met andere RDBMS'en omdat de SQL-taal universeel is. Wel zijn er soms kleine verschillen in benamingen en keywords, maar die heb je zo onder de knie.

Verder kom je in de praktijk naast databases ook datawarehouses, datalakes en big data tegen. Dit zijn varianten van databases en ondanks dat er technische verschillen zijn, kun je ze voor data-analyse op dezelfde manier benaderen. Want alle varianten werken met SQL waardoor SQL als taal alleen nog maar universeler is. Wat maakt dat jouw kennis van SQL en data-analyse nog breder inzetbaar en daarvoor nog waardevoller is.

Waarom het nuttig is om SQL te leren

Oké, je hebt inmiddels een eerste indruk van wat SQL is en kan. Maar waarom zou jij SQL moeten leren? Is dat niet iets voor programmeurs of voor specialisten? Nee, niet per se. Natuurlijk gebruiken programmeurs en specialisten zoals data-analisten SQL. Maar door z'n eenvoud en kracht kan iedereen met SQL data analyseren. Dus jij ook. Waardoor je niet langer afhankelijk bent van de drukbezette IT-afdeling of data-analisten om een omzetanalyse te maken waarmee je het resultaat van je team of afdeling kunt verbeteren.

Snel en slagvaardig zelf je informatie samenstellen

In plaats van te wachten totdat IT eindelijk tijd heeft voor jouw verzoek, ga je zelf aan de slag en heb je in een handomdraai de informatie die je nodig hebt. Hierdoor ben je veel slagvaardiger en speel je snel in op nieuwe ontwikkelingen, situaties en kansen.

Meer inzicht door verschillende invalshoeken

Verder kun je met SQL je data eenvoudig van verschillende kanten bekijken waardoor je nog meer inzicht verkrijgt die je helpen om je beslissingen, producten, diensten en processen te optimaliseren. Zo kun je verkoopgegevens met SQL bijvoorbeeld per jaar, seizoen en maand rapporteren. Of per regio of postcodegebied. Of per klantgroep of markt. Hierdoor ontdek je uitzonderingen en trends die je anders mist en die je nu in staat stellen om je aandacht op de meest kansrijke verbeteringen te richten.

Wat is data-analyse?

Het begrip data-analyse is al meerdere keren gevallen. In de zakelijke wereld lijkt het soms alsof data-analyse een relatief nieuw principe van de laatste jaren is, maar in de werkelijkheid analyseren mensen al eeuwenlang gegevens. Alleen de huidige vorm van data-analyse is nog vrij nieuw en dat komt door:

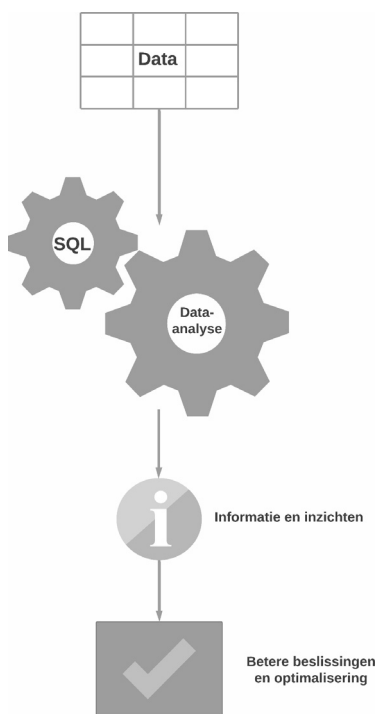
- De enorme groei van data door de verdergaande digitalisering van onze samenleving met naast computers ook tablets en smartphones.
- De beschikbaarheid van tools zoals SQL om die grote hoeveelheden data (big data) te analyseren.

- De ontwikkeling dat bedrijven en organisaties steeds meer waarde hechten aan data-gestuurd beslissingen nemen in plaats van alleen te vertrouwen op kennis en ervaring.

Dit heeft geleid tot specialistische functies als data-analist en data-scientist. Prima ontwikkelingen die bedrijven en organisaties helpen om waardevolle informatie uit hun data te halen. Maar zoals we al zagen, is dat dankzij SQL niet alleen voor specialisten weggelegd. Want jij kunt met SQL immers ook data analyseren.

Data-analyse is een proces om inzicht uit data te halen

Voor een goed begrip is het goed om te benoemen wat we precies onder data-analyse verstaan. Data-analyse is het proces waarbij je data omvormt tot informatie om nuttige inzichten te krijgen. Met als doel om trends, patronen en verbanden in informatie te ontdekken waarmee je beter onderbouwde beslissingen neemt en processen optimaliseert:



Afbeelding 1.4: Het proces van data-analyse met SQL.

In ons eerdere voorbeeld in dit hoofdstuk bestaat de data uit klantgegevens zoals voornaam, achternaam en plaats. In die basisvorm heeft deze data niet direct een bijzondere waarde voor ons. Maar als we met SQL de data analyseren en een lijst opstellen van onze klanten in Praag wordt de data ineens waardevolle informatie als we weten dat er binnenkort een beurs in Praag is waar we onze allernieuwste producten showen.

Processtappen van data-analyse

Een data-analyse begint altijd met een vraag zoals: “Wat is het gemiddelde aantal treinpassagiers in de ochtendspits op het traject Meppel-Zwolle?” Daarna ga je data analyseren om deze vraag te beantwoorden. In de praktijk zie je dan voor het gehele data-analyseproces vaak deze stappen:

- 1 De beginvraag om inzicht in een probleem of vraagstuk te krijgen als uitgangspunt voor de data-analyse nemen.
- 2 Uitzoeken welke databases en informatiebronnen beschikbaar zijn waarmee men de vraag mogelijk kan beantwoorden.
- 3 De databases verkennen om te achterhalen of men de benodigde informatie uit de databases kan halen en om inzicht te krijgen in hoe de data in de databases gestructureerd is.
- 4 SQL-query's maken die de benodigde gegevens uit de database halen en deze via bewerkingen, berekeningen en groeperen omvormen tot de gevraagde informatie.
- 5 De resultaten van de query's opleveren en gebruiken als input voor het oplossen van het oorspronkelijke probleem of vraagstuk.

Uiteraard kun je deze stappen aan jouw situatie en ervaring aanpassen. Weet je bijvoorbeeld van een eerdere data-analyse welke databases beschikbaar zijn en welke data die databases bevatten? Dan sla je de stappen 2 en 3 over en maak je meteen de benodigde SQL-query's om de gewenste informatie te presenteren.

Wat data-analyse je oplevert

Data-analyse geeft je een beter inzicht in je werk en processen en dat biedt deze mogelijkheden en voordelen:

- Goed onderbouwde beslissingen nemen omdat je uitgaat van objectieve en controleerbare gegevens.
- Hogere efficiëntie bereiken omdat je processen nauwgezet kunt analyseren en optimaliseren.
- Vroege signalering van problemen en risico's doordat je uitzonderingen eerder opmerkt.
- Nieuwe kansen en ontwikkelingen ontdekken doordat je patronen en trends in data herkent.
- Voordeel op concurrenten als je data sneller analyseert dan concurrenten en je daarmee onderscheidt in de markt.

Voorbeelden van wat je met data-analyse kunt

Bij data-analyse denk je al snel aan het analyseren van verkoopgegevens van commerciële bedrijven, maar data-analyse is veel breder dan dat en waardevol voor allerlei soorten organisaties. Kijk maar naar het volgende overzicht van data-analysetoepassingen voor verschillende sectoren:

Sector	Data-analysetoepassing
Logistiek	▪ Voorraadbeheer optimaliseren door te voorspellen welke producten wanneer nodig zijn. ▪ Analyse van levertijden en signaleren knelpunten in de leveringen.
Marketing en verkoop	▪ Prijsbepaling op basis van historische verkoopgegevens. ▪ Analyseren van het effect van marketing-campagnes.
Gezondheidszorg	▪ Voorspellen van het aantal patiënten in een ziekenhuis om personeelsplanning te optimaliseren. ▪ Analyseren van de effectiviteit van behandelingen en medicijnen.
Onderwijs	▪ Inzicht in factoren die bijdragen aan studievertraging of uitval. ▪ Signaleren van welke studenten extra ondersteuning nodig hebben.

Sector	Data-analysetoepassing
Winkels en retail	■ Bepalen van welke acties en kortingen het meest effectief zijn. ■ Analyseren van bestverkochte producten en seizoenspatronen.
IT en tech	■ Optimaliseren van cloudgebruik en kosten op basis van gebruiksdata. ■ Analyseren van gebruikersfeedback om software of apps te verbeteren.
Overheden	■ Voorspellen van economische trends en werkgelegenheidscijfers. ■ Analyse van verkeersdata om files te verminderen.
Human Resources	■ Analyseren van medewerkerstevredenheid en betrokkenheid. ■ Bepalen van de meest effectieve wervingskanalen voor nieuwe medewerkers.

Tabel 1.1: Voorbeelden van data-analyse toepassingen per sector.

Dit is slechts een fractie van wat je allemaal kunt doen met data-analyse en is vooral bedoeld om de veelzijdigheid ervan te laten zien. In welke sector of in wat voor soort organisatie je ook werkt, data-analyse is altijd een zeer waardevol instrument om je processen en resultaten te optimaliseren.

Voor wie is dit boek?

Dit boek is voor iedereen die inzicht uit data wil halen om betere beslissingen te nemen en processen te optimaliseren. Dus het maakt niet uit of je magazijnmedewerker, boekhouder, teamleider, balie-medewerker, eenmanszaak eigenaar, consultant, verpleegkundige, stafmedewerker of laborant bent. Data-analyse met SQL zoals dit boek je leert, is nuttig voor iedereen.

Geen voorkennis nodig

Om dit boek voor iedereen toegankelijk te maken, heb je geen specifieke voorkennis nodig voor dit boek. Stap voor stap leer je SQL en

data-analyse aan de hand van praktische uitleg en realistische voorbeelden en opdrachten. Want zelf aan de slag gaan en oefenen is de beste en leukste manier om data-analyse met SQL te leren.

Het is niet noodzakelijk, maar wel handig als je wat ervaring met Excel of andere spreadsheetprogramma's hebt. Want dan ben je al gewend aan werken met data in rijen en kolommen. Misschien werk je zelfs met tabellen in een spreadsheet en dat is helemaal mooi. Maar ook zonder deze ervaring is dit boek prima te volgen.

Wat is de inhoud van dit boek? Wat leer je?

Dit boek volgt een stap-voor-stap aanpak en schakelt na de benodigde informatie over databases meteen over naar de praktijk van data-analyse met SQL.

Basiskennis over databases

Je begint met de nodige achtergrond over relationele database en hoe men data in tabellen organiseert en aan elkaar relateert. Want met die kennis ben je goed voorbereid om met SQL aan de slag gaan.

SQL-tools installeren en data analyseren met SQL

Vervolgens installeer je de benodigde SQL-tools, inclusief de database en daarna ga je direct aan de slag met SQL. Eerst maak je eenvoudige, maar toch al waardevolle query's. Vervolgens bouw je de query's gaandeweg uit met meer queryonderdelen voor steeds verdergaander, breder en verfijnder inzicht.

Ontbrekende en foute data tackelen

Tussendoor kijken we ook nog naar hoe je onvolledige en soms foute data opspoort, en hoe je die op kunt lappen zodat je met dergelijke data nog steeds zinvolle data-analyses kunt maken. Want in de praktijk komen we geregeld data met gebreken tegen, maar daar laten we ons niet door uit het veld slaan. Integendeel, aan het eind van dit boek kun jij met je SQL-kennis zeer geavanceerde data-analyses maken. Zelfs als de onderliggende data op bepaalde punten hapert.