

FORMULARIUM

MongoDB

SQL

PL/SQL

Formularium MongoDB

use test

show dbs

db.dropDatabase()

db.createCollection(Name, options)

show collections

db.collectionnaam.drop()

db.collectionnaam.insertOne();

db.collectionnaam.insertMany()

db.collectionnaam.updateOne();

db.collectionnaam.updateMany()

db.collectionnaam.updateOne({SELECTIE_CRITERIA},
{UPDATE_SPECIFICATIE}

db.collectionnaam.deleteOne({});

db.collectionnaam.deleteMany({})

db.collectionnaam.deleteOne({SELECTIE_CRITERIA})

db.collectionnaam.findOne(); db.collectionnaam.find()

db.collectionnaam.find().sort({key:1})

db.collectionnaam.find().skip(n)

db.collectionnaam.find().limit(n)

db.collectionnaam.find().count()

db.collectionnaam.countDocuments()

db.collectionnaam.find({SELECTIE_CRITERIA},
{PROJECTIE})

SELECTIE_CRITERIA:

field: value

field:{\$lt:value}

\$lte

\$gt

\$gte

\$ne

\$in:[value1, value2, ...]

\$nin:[value1, value2, ...]

\$exists

/value/

^ vooraan

\$ achteraan

i

doc_field.subdoc_field: ...

field:[value1,value2]

field:{\$all:[value1,value2]}

field.n: value

\$size

switch naar DB test; indien niet bestaande
creatie DB test

toon alle DB

drop huidige DB

creatie collectie

toon alle collecties in de huidige DB

drop collectionnaam

voegt 1 of meerdere doc toe aan een collectie

update van 1 of meerdere docs van een collectie

verwijdert 1 of meer docs van de collectie

verwijdert het eerste doc dat voldoet aan de
criteria

toont 1 of meerdere docs

sorteert oplopend (1) of afdalend (-1) op key

slaats een aantal docs over

beperkt de output tot een aantal docs

telt het aantal items

geeft het aantal documenten weer

veld gelijk aan een bepaalde waarde

veld kleiner dan value

kleiner dan of gelijk aan

groter dan

groter dan of gelijk aan

niet gelijk aan

gelijk aan value1 of gelijk aan value2, ...

niet gelijk aan value1 en niet gelijk aan value2,
...

is aanwezig

value komt voor

moet beginnen met

moet eindigen met

case insensitive

beperking op veld van subdocument

veld dat exact de 2 values heeft

veld dat de 2 values heeft, ongeacht volgorde of
andere elementen

op positie n van het veld staat value

aantal waarden in een array

Vervolg formularium MongoDB

UPDATE_SPECIFICATIE

\$set:{field1:value1}

nieuwe waarde toekennen aan bestaande key of key bijmaken

\$unset:{field1:value1}

maakt een set ongedaan

\$push:{field1:value1}

voegt *value1* toe aan array *field1*; array MOET BESTAAN

\$push:{field1:{\$each:[value1,value2, ...]}}

voegt *value1*, *value2*, ... toe aan array *field1*; array MOET BESTAAN

Formularium SQL

Algemeen

*	alle kolommen
AS "Kolom naam"	kolom alias, AS is optioneel
DISTINCT	Enkel unieke waarden, onderdrukt dubbels
tabel1 t1	Alias T1 voor tabel1
tabel1 t1 JOIN tabel2 t2 USING (key)	Koppelt 2 tabellen (naam PK-kolom is gelijk aan naam FK-kolom)
tabel1 t1 JOIN tabel2 t2 ON t1.FK = t2.PK (of ON t1.PK = t2.FK)	Koppelt 2 tabellen waarbij FK-kolom waarde uit de ene tabel gelijk moet zijn aan waarde PK-kolom uit de andere tabel
...LEFT [OUTER] JOIN...	alle velden linkertabel worden getoond
...RIGHT [OUTER] JOIN...	alle velden rechtertabel worden getoond
...FULL [OUTER] JOIN...	alle velden van beide tabellen worden getoond
SELECT t1.kolom	selectie uit tabel(len)
FROM tabel1	Selecteert de tabel waaruit de kolommen komen.
WHERE voorwaarde	filtreert de rijen met de voorwaarde
GROUP BY	waarop moeten rijen gegroepeerd worden
HAVING voorwaarde	filtreert de rijen na groepering met de voorwaarde
ORDER BY criteria1, criteria2, ...	sorteert (oplopend) op basis van kolom, kolomalias, kolompositie of een expressie.
Criteria bij ORDER BY [ASC DESC] [NULLS FIRST LAST]	op- of aflopend, met lege waarden eerst of laatst.

Precedentie

(...), *, /, +, -, , < > =, IS LIKE IN, BETWEEN, !=, NOT, AND, OR	volgorde van bewerkingen
---	--------------------------

Vergelijkingsoperatoren

<, >, <=, >=	kleiner dan, groter dan, ... of gelijk aan
!= <>	verschillend van
=	is gelijk aan
IS [NOT] NULL	is (niet) gelijk aan NULL
BETWEEN ondergrens AND bovengrens	tussen twee waarden, inclusief de twee waarden
IN (waarde1, waarde2,...)	is gelijk aan een van de waarden tussen de haakjes
LIKE 'patroon'	komt overeen met een patroon, hoofdlettergevoelig
% (in patroon)	willekeurig aantal karakters
_ (in patroon)	één willekeurig teken

Logische operatoren

AND	vereist dat beide voorwaarden waar zijn
NOT	vereist dat de voorwaarde niet waar is (o.a. NOT IN, NOT BETWEEN, ...)
OR	vereist dat minstens één voorwaarde waar is

Conversiefuncties

TO_CHAR(getal, formaat)	conversie van getal naar een tekst
TO_CHAR(datum, formaat)	conversie van datum naar een tekst
TO_DATE(string, formaat)	conversie van tekst naar een datum
TO_NUMBER(string, formaat)	conversie van tekst naar een getal

Vervolg formularium SQL

Formaatmodel werken met getallen

9	Placeholder voor positie voor een cijfer, geen nullen voor getal indien getal te kort is (wel na komma)
0	Placeholder voor positie voor een cijfer, indien onvoldoende cijfers wordt links opgevuld met nullen.
.	Decimale punt
,	Komma tussen duizendtallen
FM	Onderdrukt spaties en nullen, zowel vooraan als achteraan (maar niet als 0 als placeholder gebruikt wordt)
L	Symbool lokale munteenheid sessie
MI	Om een minteken te plaatsen bij negatieve getallen. Kan zowel voor als achter getal, op de plaats van MI wordt dan het minteken geplaatst.
S	Plus- of minteken voor getal

Formaatmodel werken met datums

'tekst'	tekst, letterlijk in het formaat
Day	dag van de week voluit (hoofdlettergevoelig => DAY, Day, day)
DY	dag van de week in drie letters
D	Numeriek dag van de week(1= zondag, 7 = zaterdag)
DD	numerieke dag van de maand(01-31)
DDD	Numerieke dag van het jaar(001-366)
W	Numerieke week van de maand
WW	Numerieke week van het jaar
th	Na DD zorgt voor aangeven rangtelwoord st,nd
MON	maand, drie letters (hoofdlettergevoelig =>MON, Mon, mon)
Month	maand, voluit geschreven (hoofdlettergevoelig=>MONTH, Month, month)
MM	Numerieke maand, in 2 cijfers (01-12)
YY	jaar in 2 cijfers
Year	jaar uitgespeld
YYYY	jaar in 4 cijfers
HH	Uur van de dag (0-12)
HH12	Uur van de dag (0-12)
HH24	Uur van de dag (0-23)
MI	Minuten
SS	Seconten
MS	Miliseconden(000-999)
AM, am PM, pm	Tijdsindicator bij 12 uren systeem

Getallen

* + - /	rekenkundige expressies
MOD(deler, deeltal)	modulo, de rest van deler na deling door deeltal op gehele getallen
ROUND(getal[, n])	rondt getal af op n cijfers na de komma (geen n is 0, n < 0 is n plaatsen voor de komma).
TRUNC(getal[, n])	kapt getal af op n cijfers na de komma (geen n is 0, n < 0 is n plaatsen voor de komma).

Vervolg formularium SQL

Karakterfuncties

'dit is tekst'	Tekst, hoofdlettergevoelig
	Concatenatie tussen kolommen of met tekst.
INITCAP(tekst)	zet de eerste letter van elk woord in hoofdletters, de rest in kleine letters
LOWER(tekst)	zet tekst om naar kleine letters
UPPER(tekst)	zet tekst om naar hoofdletters
CONCAT(c ₁ , c ₂ , ..., c _n)	Voegt de tekst c ₁ , c ₂ tot en met c _n samen.
INSTR(tekst,subtekst)	Geeft de eerste positie van de subtekst in de tekst.
LENGTH(tekst)	geeft aantal karakters in tekst terug
LPAD(c1, n, c2)	vult c1 wordt links aan tot lengte n met karakters van c2
REPLACE(tekst, s, r)	Ieder voorkomen van s in tekst vervangen door r
RPAD(c1, n, c2)	vult c1 rechts aan tot lengte n met karakters van c2
SUBSTR(tekst, start[, n])	tekstdeel uit tekst vanaf positie start, n karakters lang (of tot het einde). Het eerste karakter heeft positie 1. Negatieve argumenten tellen vanaf het eind.
TRIM(tekst1,tekst2)	verwijdert karakters tekst2 aan het begin en het einde van tekst1
LPAD(tekst, getal[,opvulling])	Vult de string links op met opvulling (of spaties) tot getal_aantal indien tekst te weinig karakters bevat. Indien tekst langer dan getal_lengte, wordt de tekst rechts afgekapt.
RPAD(tekst, getal[,opvulling])	Vult de string rechts op met opvulling (of spaties) tot getal_aantal indien tekst te weinig karakters bevat. Indien tekst langer dan getal_lengte, wordt de tekst rechts afgekapt.
REPLACE(tekst, teVervangen, vervanger)	Vervangt alle teVervangen karakter(s) in tekst door de vervanger karakter(s)

Datums rekenen

datum1– datum2	geeft aantal dagen (getal) tussen twee datums (type = numeric)
CURRENT_DATE	Huidige datum (in de tijdzone van de huidige sessie)
CURRENT_TIMESTAMP	Huidige datum en tijd (in tijdzone sessie)
SYSDATE	Huidige datum en tijd (in tijdzone van database server)
MONTHS_BETWEEN(datum1, datum2)	Geeft maanden tussen datum1 en datum2 weer
ADD_MONTHS(datum, n)	Telt n maanden bij datum
NEXT_DAY(datum, 'dagweek')	Geeft datum dagweek volgend op datum
LAST_DAY(datum)	Geeft laatste dag van de maand
ROUND(datum, eenheid)	Rond datum af met eenheid 'month' of 'year'
TRUNC(datum, eenheid)	Kapt datum af met eenheid 'month' of 'year'
EXTRACT(DOW FROM datum)	Geeft de numerieke dag van de week (zondag =0 en zaterdag = 6)
EXTRACT(DAY FROM datum)	Geeft de numerieke dag van de maand van de datum.
EXTRACT(MONTH FROM datum)	Geeft de numerieke maand van het jaar op van de datum.
EXTRACT(YEAR FROM datum)	Geeft het jaartal in 4 cijfers van de datum.

Vervolg formularium SQL

Conditionele functies

NVL(a, b)	het eerste niet-NULL argument wordt genomen
COALESCE(a, b, c, ...)	het eerste niet-NULL argument wordt genomen
NULLIF(expr1, expr2)	Null als expr1=expr2 en anders expr1

Conditionele expressie

CASE WHEN <conditionele expressie> THEN <returnwaarde> [WHEN <conditionele expressie> THEN <returnwaarde>] ELSE <returnwaarde> END	Wanneer de conditionele expressie true is, wordt de bijhorende returnwaarde gegeven.
De WHENTHEN ... clausule kan zo vaak als nodig herhaald worden; indien geen enkele conditionele expressie true is, wordt de returnwaarde bij ELSE teruggegeven.	

Groepfuncties

AVG(expr)	Gemiddelde van alle meegegeven niet NULL waarden
COUNT(expr)	aantal rijen (NULL waarden tellen niet mee)
COUNT(*)	Aantal meegegeven rijen
MAX(expr)	maximum waarde van alle niet NULL waarden
MIN(expr)	minimum waarde van alle niet NULL waarden
SUM(expr)	Som van alle niet NULL waarden

Subquery

Haalt waarden uit 1 kolom op (kunnen meerdere waarden zijn). Subquery geeft 1 waarde terug:

Expressie [=, <, >, >=, <=, !=] (subquery)	Expressie wordt vergeleken op basis van meegegeven operator met teruggegeven waarde.
---	--

Subquery geeft meerdere waarden terug:

Expressie IN (subquery)	Expressiewaarde is gelijk aan 1 van de weergegeven waarden van de subquery
Expressie NOT IN (subquery)	Expressiewaarde mag niet gelijk zijn aan eender welke van de teruggegeven waarden van de subquery.
Expressie operator ANY (subquery)	Expressiewaarde komt overeen volgens gebruikte operator met één van de weergegeven waarden van de subquery.
Expressie operator ALL (subquery)	Expressiewaarde komt overeen volgens gebruikte operator met alle van de weergegeven waarden van de subquery.
=, <, >, >=, <=, !=	Mogelijke operatoren om te gebruiken met ANY of ALL.

DML

COMMIT;	Bevestigen wijziging
ROLLBACK [TO savepoint-naam] ;	Ongedaan maken wijzigingen [tot ...]
SAVEPOINT naam;	Instellen tussenpunt in transactie
INSERT INTO tabel[(kolomnaam, kolomnaam, ...)] VALUES (waarde [, waarde, ...]);	1 rij data toevoegen aan een tabel (eventueel ipv value DEFAULT.
UPDATE tabel SET kolomnaam = waarde [, kolomnaam=waarde, ...] [WHERE conditie];	Data wijzigen in een bepaalde kolom in een bepaalde tabel [in bepaalde rijen]
DELETE FROM tabel [WHERE conditie];	Rijen verwijderen uit een tabel.

Vervolg formularium SQL

DDL

CREATE TABLE naam (kolomnaam datatype [DEFAULT expressie] [CONSTRAINT naam type ...] [, kolomnaam datatype ...]);	Een tabel creëren met kolommen en de nodige constraints.
CREATE TABLE naam (kolomnaam datatype [DEFAULT expr] [kolom_constraint] [, [, tabel_constraint] [, ...]);	Constraints kunnen op kolomniveau of op tabelniveau gemaakt worden, met een specifieke syntax
kolomnaam datatype CONSTRAINT constraint_naam constraint_type	Kolomconstraint (voor PK, NOT NULL, UNIQUE)
, CONSTRAINT constraint_naam constraint_type(kolomnaam)	Tabelconstraint (voor PK en UNIQUE)
kolomnaam datatype CONSTRAINT constraint_naam REFERENCES tabel(PK)	Kolomconstraint voor FK
kolomnaam datatype , CONSTRAINT constraint_naam FOREIGN KEY(kolomnaam) REFERENCES tabel(PK)	Tabelconstraint voor FK
kolomnaam datatype CONSTRAINT constraint_naam CHECK(conditie)	Kolomconstraint voor check
kolomnaam datatype , CONSTRAINT constraint_naam CHECK(conditie)	Tabelconstraint voor check
DROP TABLE [IF EXISTS] naam[,...] [CASCADE RESTRICT];	Verwijderen van 1 [of meerdere] tabellen [met of zonder relaties/constraints]
ALTER TABLE naam ADD type nodige_info	Toevoegen van kolom met datatype, constraint op kolomniveau ...
ALTER TABLE naam DROP type nodige_info	Verwijderen van kolom, constraint ...
CREATE [OR REPLACE] VIEW naam_vu [(kolomnaam[, kolomnaam ...])] AS SELECT kolom[,kolom,...] FROM table [WHERE conditie];	Het maken van een view op basis van een subquery
DROP VIEW viewnaam_vu;	Verwijderen van een view
CREATE SEQUENCE sequence_naam_seq [INCREMENT [BY] waarde] [MINVALUE waarde <u>NO MINVALUE</u>] [MAXVALUE waarde <u>NO MAXVALUE</u>] [START [WITH] waarde] [[NO] CACHE aantal] [[NO] CYCLE];	Het maken van een sequentie
nextval(sequentienaam)	Ophalen volgende waarde sequentie
DROP SEQUENCE sequentienaam;	Verwijderen van sequentie
CREATE INDEX naam ON tabel(kolom);	Index maken op een kolom

Formularium PL/SQL

Anoniem blok

```
[DECLARE]
BEGIN
    -- statements
[EXCEPTION]
END;
```

Declaraties

```
variabele_naam [CONSTANT] datatype [NOT NULL] [:= | DEFAULT expressie];
variabele_naam tabel.kolom%TYPE;
```

SELECT in PL/SQL

```
SELECT select_list
INTO {variabele_naam [, variabele_naam] ...}
FROM tabel
...
```

Afdruk in de console

```
DBMS_OUTPUT.PUT_LINE();
```

Exception-handler

```
EXCEPTION
    WHEN exception1 [OR exception2...] THEN
        -- statements;
    [WHEN exception3 [OR exception4...] THEN
        -- statements;
    [WHEN OTHERS THEN
        -- statements;
```

Enkele voorgedefinieerde exceptions

NO_DATA_FOUND	ORA-01403
TOO_MANY_ROWS	ORA-01422
ZERO_DIVIDE	ORA-01476
VALUE-ERROR	ORA-06502
DUP_VAL_ON_INDEX	ORA-00001

Niet-voorgedefinieerde exceptions → zelf declareren en koppelen

```
e_exception EXCEPTION;
PRAGMA EXCEPTION_INIT(e_exception, -ora-foutcode);
```

Vervolg formularium PL/SQL

User-defined exceptions

Stap1	declareren	<code>e_exception EXCEPTION;</code>
Stap2	opwerpen	<code>RAISE e_exception;</code>
Stap3	opvangen in EXCEPTION-handler	

Exception-gerelateerde functies

SQLERRM	Geeft de foutmelding weer die bij een error hoort
SQLCODE	Geeft de numerieke waarde weer die bij een error hoort
0	Er is geen exception
1	Het gaat om een user-defined exception
+100	NO_DATA_FOUND exception
Negatief getal	Een andere Oracle error

Zelf een fout opwerpen en doorgeven naar applicatie:

`RAISE_APPLICATION_ERROR(error-nr, foutmelding);`

- Error-nr: zelf te kiezen getal tussen -20 000 en -20 999

Procedure

```
CREATE [OR REPLACE] PROCEDURE procedure_naam
    [(parameter1 [IN|OUT|IN OUT] datatype1,
      parameter2 [IN|OUT|IN OUT] datatype2, ...)]
IS|AS
    [declaratie lokale variabelen;]
BEGIN
    -- statements;
[EXCEPTION]
END [procedure_naam];
```

Vervolg formularium PL/SQL

Functie

```
CREATE [OR REPLACE] FUNCTION functie_naam
    [(parameter1 [IN|OUT|IN OUT] datatype1,
      parameter2 [IN|OUT|IN OUT] datatype2, ...)]
RETURN datatype
IS|AS
    [declaratie lokale variabelen;]
BEGIN
    -- statements;
    RETURN expressie;
[EXCEPTION]
END [functie_naam];
```

Trigger

```
CREATE [OR REPLACE] TRIGGER trigger_naam
{BEFORE | AFTER}
{DELETE | INSERT | UPDATE [OF kolom [,kolom] ] } [ OR ...]
ON tabelnaam
[FOR EACH ROW [WHEN (voorwaarde) ] ]
[DECLARE
    declaratie lokale variabelen;]
BEGIN
    -- statements;
[EXCEPTION]
END [trigger_naam];
```

IF

```
IF voorwaarde THEN
    -- statements
[ELSIF voorwaarde THEN
    -- statements]
[ELSE
    -- statements]
END IF;
```

Vervolg formularium PL/SQL

Basic loop

LOOP

-- statements

EXIT [WHEN *voorwaarde*];

END LOOP;

WHILE loop

WHILE *voorwaarde* LOOP

-- statements

END LOOP;

FOR loop

FOR counter IN [REVERSE] *beginwaarde* .. *eindwaarde* LOOP

-- statements

END LOOP;

FOR loop using subquery

FOR *record* IN (*subquery*) LOOP

-- statements

END LOOP;

SQL cursor attributen

SQL%FOUND	Boolean, is TRUE als het laatste DML-statement rijen gewijzigd heeft
SQL%NOTFOUND	Boolean, is TRUE als het laatste DML-statement geen rijen gewijzigd heeft
SQL%ROWCOUNT	Geheel getal dat aangeeft hoeveel rijen gewijzigd werden door het laatste DML-statement