|  |  |  |
| --- | --- | --- |
| [Abteilung/Prozess] | [Titel des Dokuments] | REV 000 |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | Verfasser: | [Nachname Vorname] | Datum: | [YYYY-MM-DD] |

Select- Wählt aus Welche Tabelle verwendet werden solte

INAERT-fügt Daten Sätze in die Datenbank

UPDATE- ändert Datensetze

DELETE- Löscht Datensätze

INNER JOIN- fügt zwei Tabellen zusammen

UNION- Mehrere abfragen werden zu einem zusammengefasst

SUMM()- Sumierung einer spalte

AVG()- Durchschnitt berechnen

MIN()- kleinster wert der Spalte

MAX()- Gröster wert der Spalte

COUNT()- Anzahl der zeilen

ORDER BY- Ordent die tbelle(n) anhand verschiedener parametet

GROUP BY- Gruppiert die tbelle(n) anhand verschiedener parametet

|  |  |
| --- | --- |
| VARCHAR | Für Buchstaben |
| DECIMAL | Ganzahlen mit variabel fürs ende |
| DATE | Datum |
| BOOLEAN | Wahr oder Falsch Werte |
| INTEGER | Ganzzahlen |
| FLOAT | Gleitkommazahlen |

|  |  |
| --- | --- |
| 1:1 Beziehung | Die datensätze in den tabellen sind nur mal Ferküpft |
| 1:n Beziehung | Datensätze aus Tabelle 1 können mehrmals verknüpft werden Tabelle 2 Datensätze könen nur einmal vorkommen |
| n:m Beziehung | Mit einer zwischentabelle wird angegeben welche datensätze (tb1) mit welchen datensätzen (tb2) wie in ferbindung stehen |

|  |  |
| --- | --- |
| Attribut | Spalten |
| Primärschlüssel | eine Spalte in einer Tabelle, deren Werte eine Zeile in der Tabelle eindeutig kennzeichnen. |
| Fremdschlüssel | Verweist auf einen primär Schlüssel aus eine anderen Tabelle |
| Redundanz | Dopplete informationen (gilt zu vermeiden) |
| Referenzielle Integrität | // |
| Entität | Daten die sich anhand eines merkmals zusammen fassen lassen |
| Aggregation | Fasst alles in einer gruppe zusammen |
| Tupel | Wertesamlung |

Select anrede, vorname, nachname FROM Personen where wohnort=‘Wien‘

Select alter FROM Personen order by alter desc

Select avg(alter) FROM Personen where geschlecht=’m’

Select vorname, nachname, geschlecht, FROM Personen Where alter>18

Select MAX(alter) FROM Personen where geschlecht=’m’ and wohnort=‘Wien‘

Select MIN(alter) FROM Personen where geschlecht=’w’ and wohnort=‘Graz‘ or wohnort=‘Salzburg‘

Update Personen

SET wohnort=‘Linz‘

WHERE vorname=’Monika’ and Nachname=’Becker’

INSERT INTO Personen (id, anrede, vorname, nachname, geschlecht, wohnort, plz, alter)

VALUES (12, Herr, Marius, Baum, m, Klagenfurt, 9010, 35)

Relationale Datenbank der Vorteil es gibt keine Redundanz bei den daten das bedeutet das so viel Speicher eingespart werden kann vor allem bei größeren Datenbanken

Select vorname, nachname, geschlecht, FROM Personen inner join Orte Personen.plz=orte.plz Where alter>18

INSERT INTO Personen (id, anrede, vorname, nachname, geschlecht, plz, alter)

VALUES (13, Frau, Petra, Kinz, w, 6900, 41)

INSERT INTO Orte (wohnort, plz)

VALUES (Bregenz, 6900)

Select wohnort, count(\*) FROM Personen inner join Orte Personen.plz=orte.plz group by wohnort

