Data Manipulation Language (DML)

search for information in a database : **SELECT**

```
SELECT { <expression> [AS <column name>] } [ , ... n ]
[ INTO  <new table> ]
FROM  <table>  or  <table join>
[ WHERE  <line filtering condition> ]
[ GROUP BY  { <expression> } [ , ... n ] ]
[ HAVING  <line filtering condition s> ]
[ ORDER BY  { <expression> } [ , ... n ] ]
[ COMPUTE  { <expression> } [ , ... n ] BY { <attribute> } [ , ... n ] ]
[ COMPUTE  { <expression> } [ , ... n ]
```

inserting rows into a table : **INSERT INTO**

```
INSERT  [INTO]  <table>  [ ( <attribute 1>, <attribute 2>, ..., <attribute n> ) ]
      { VALUES { ( <val_attribute 1>, <val_attribute 2>, ..., <val_attribute n> ) } [ , ... n ]  |
        <query SELECT> }
```

modifying the rows of a table : **UPDATE**

```
UPDATE  <table>  SET  { <attribute>=<expression> } [ , ... n ]
[ WHERE  <line filtering condition> ]
```

removing rows from a table : **DELETE**

```
DELETE  [FROM]  <table>  [ WHERE  <line filtering condition> ]
```

Data Manipulation Language (DML)

creating a table : **CREATE TABLE**

```
CREATE TABLE  <table>
(   {        <column name>  <type>
       [ NULL | NOT NULL ]
       [ IDENTITY  [ ( <initial value>, <increment> ) ]  |  DEFAULT ( <constant> | <expression> ) ]
       [ { <constraint on columns> } [ , ... n ] ]
    }  [ , ... n ] ,
   [ { <constraint on columns> } [ , ... n ] ]
)
```

changing columns or constraints in a table : **ALTER TABLE**

```
ALTER TABLE  <table>
{   ALTER COLUMN  <column name>  <type> ... |
    ADD  { <column definition> | <constraint> } [ , ... n ]  |
    DROP {  [CONSTRAINT] { <constraint name> } [ , ... n ]  |
                COLUMN { <column name> } [ , ... n ]
            } [ , ... n ]
}
```

NB : statements between the square brackets **[ ]** are optional.

{ *<content>* } [, ... n] means that ***<content>*** can be repeated one or more times.
(each ***<content>*** is separated with a comma).