

```
mainC){  
    SQL sql;  
    sql.run();  
    return 0;  
}
```

SQL

run(): call run-command in a loop

run-command(string)

batch? run_batch()

ptree = parser()

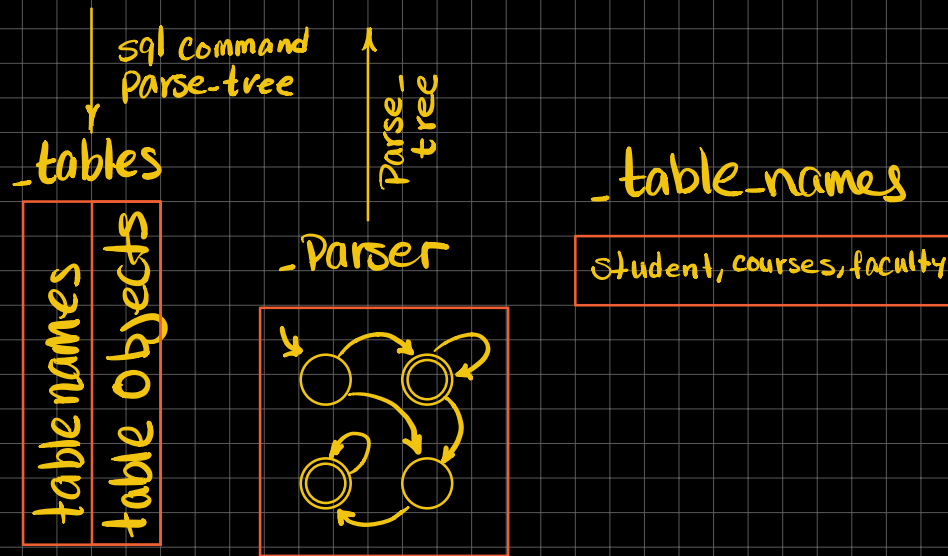
insert()

select()

create-table

drop-table : _tables[...].drop_table(...)

bool get_sql_command(ptree, command)



Table

Table() open table
 Table() create a table
 delete_table
 insert_into
 select, select_all

reindex
 operator<<

_indices

static int serial

0 first

1 Last

2 Dept

Bill	11, 37, 40
Dora	19, 47
Harry	73
Pete	30, 73, 76
Rich	73, 59, 63
Wanda	73, 87, 92
James	10, 38, 47, 58
James	11, 76, 90
Marques	48, 51
Spencer	42
Chloe	10, 22, 30
CS	22, 38, 54
Hum	51, 73, 80
Phu	48, 77, 83
SoC	42

vector of multimap

_name
 _file_name

_fieldnames

last
 first
 age
 dept
 salary

_keys: enum: SELECT, FROM, WHERE, INSERT, ...

_Keywords: maps "SELECT" → SELECT

Workflow

phase 1:

Table:

CTOR: construct new table

insert: write record to file

select-all() read & display
all records

create a table class:

Table (name, vector field-names)

stores - name

stores - field-list

stores - file-name

creates file w/ - file-name

void insert_into (vector field-list)

writes the record unto
binary file

void select-all()

retrieves all records from
file. prints them on console.

```
main() {
    Table t ("employee", { "first", "last", "age" })
    t.insert_into ({ "Joe", "Baker", "23" });
```

→ string

→ vector

t.select_all()

Employee:

first	last	age
Joe	Baker	23

-name ✓
-file-name ✓
-fieldnames ✓

last
first
age
dept
salary

employee.bin

Joe Baker 23	Jane Doe 19	Steve Cust 35	}
--------------------	-------------------	---------------------	---

phase 2:

Table:

_indices is implemented

_select_where is implemented

single condition

=, >, <=

insert-into pushes keys & record numbers
into _indices

_select_where looks up record nums
from _indices & returns a table

_select returns a table

```
main() {
```

→ string

→ vector

```
Table t ('employee', { "first", "last", "age" })
```

```
t.insert_into ({"Joe", "Baker", "23"});
```

```
"
```

```
"
```

```
cout << t.select_all() << endl;
```

```
cout << t.select_where ( { "last", "=",
```

```
'Baker' } )
```

```
}
```