

Databases

- *Oracle*
- *Sql server*
- *DB2*
- *Sybase*
- *Informix*
- *Ingress*
- *Mysql*
- *postgres*

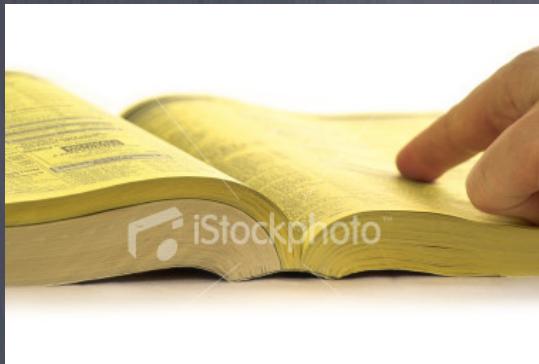
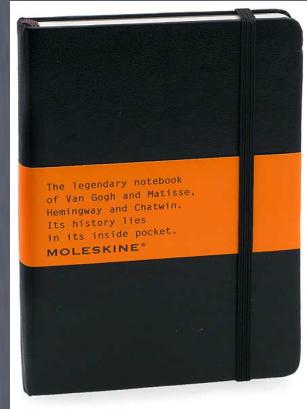


What is a database?

- *A collection of data organized for rapid search and retrieval*

*A database is a storage space for content / **information** (data)*

What is a database



What is a DBMS?

- A *software system designed to manage a database.*



Tabular Data - more examples

<i>Date</i>	<i>Loan-Num</i>	<i>State</i>	<i>Balance</i>	<i>Category</i>	<i>Repaid</i>
1/4/99	123	TN	3000	Farm	0
1/4/99	456	MH	3000	Smallbiz	100
4/4/99	998	MH	12000	Coop	50
4/5/99	765	GJ	212000	Bigbiz	25
4/5/99	666	AP	7000	Govtbiz	10

Relational Database Concepts

- *Relationships*
- *Constraints*
- *Triggers*
- *Stored procedures*
- *Tables*
- *Views*
- *Indexes*



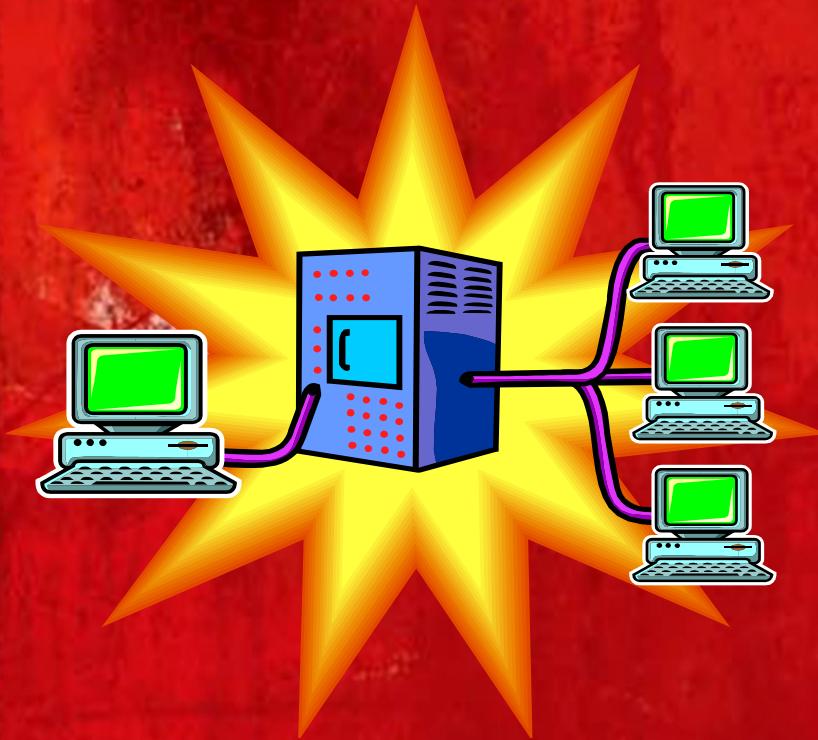
Oracle License Options

Oracle Enterprise edition

Oracle Standard edition

Oracle Express edition

Oracle Connection Process



Oracle Connection Process

Log On

User Name:

system

Password:

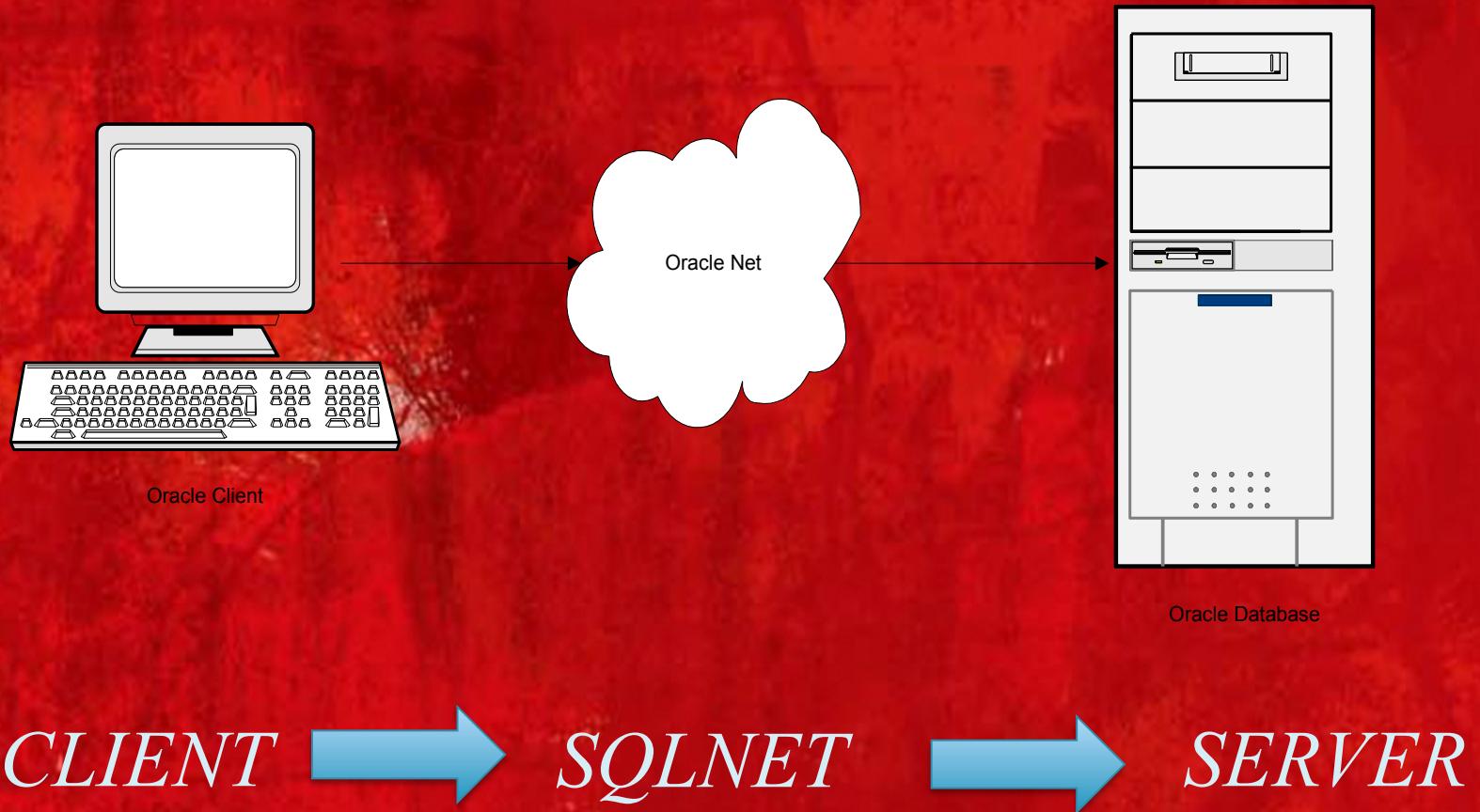
Host String:

prod.cccinternal.com

OK

Cancel

Oracle Connection Process



Oracle Server

Oracle instance + Database

Oracle components

Init file

Lists the resources and sizes

Data

Actual data

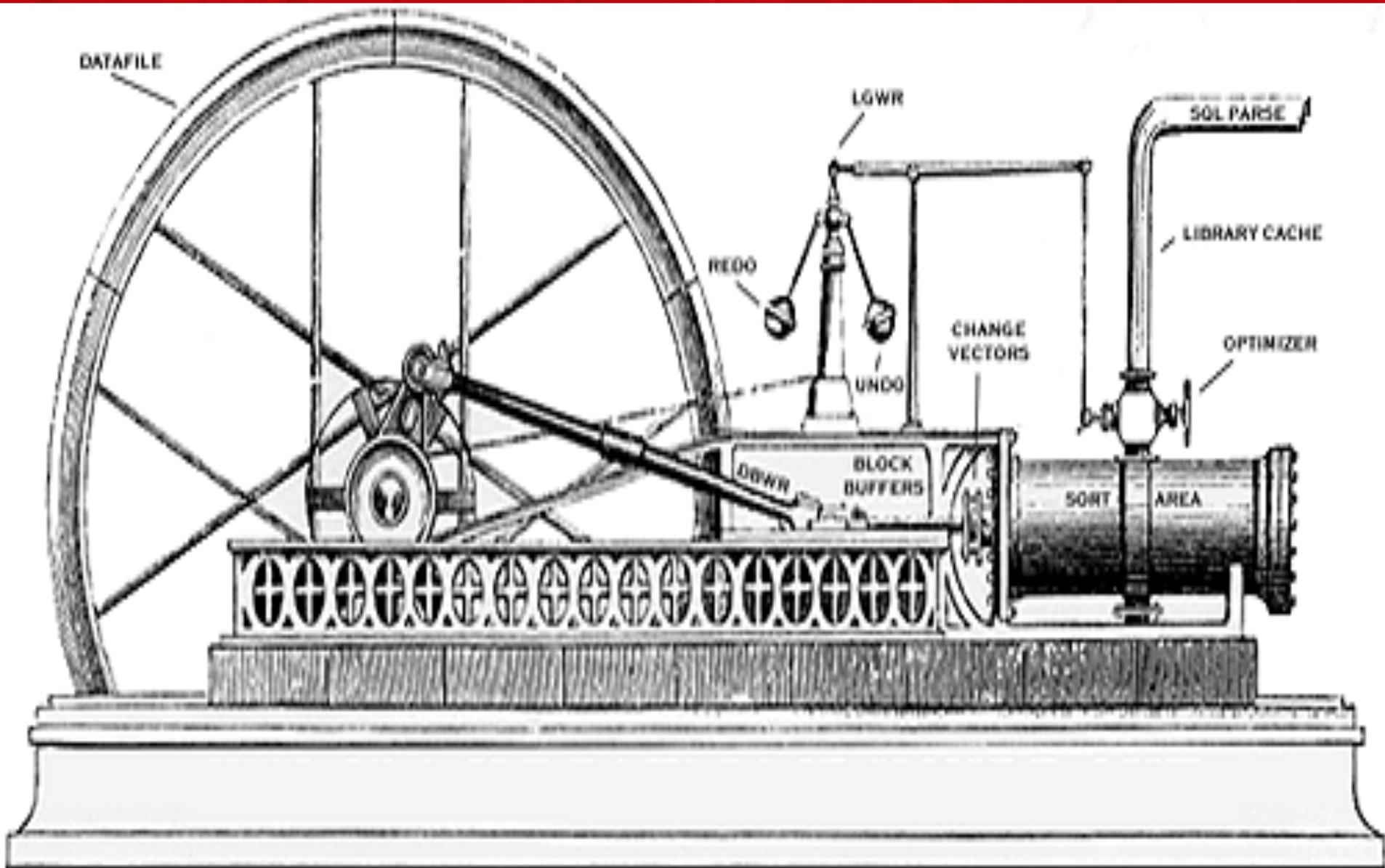
Redo

Changes made to the database-Used for recovery

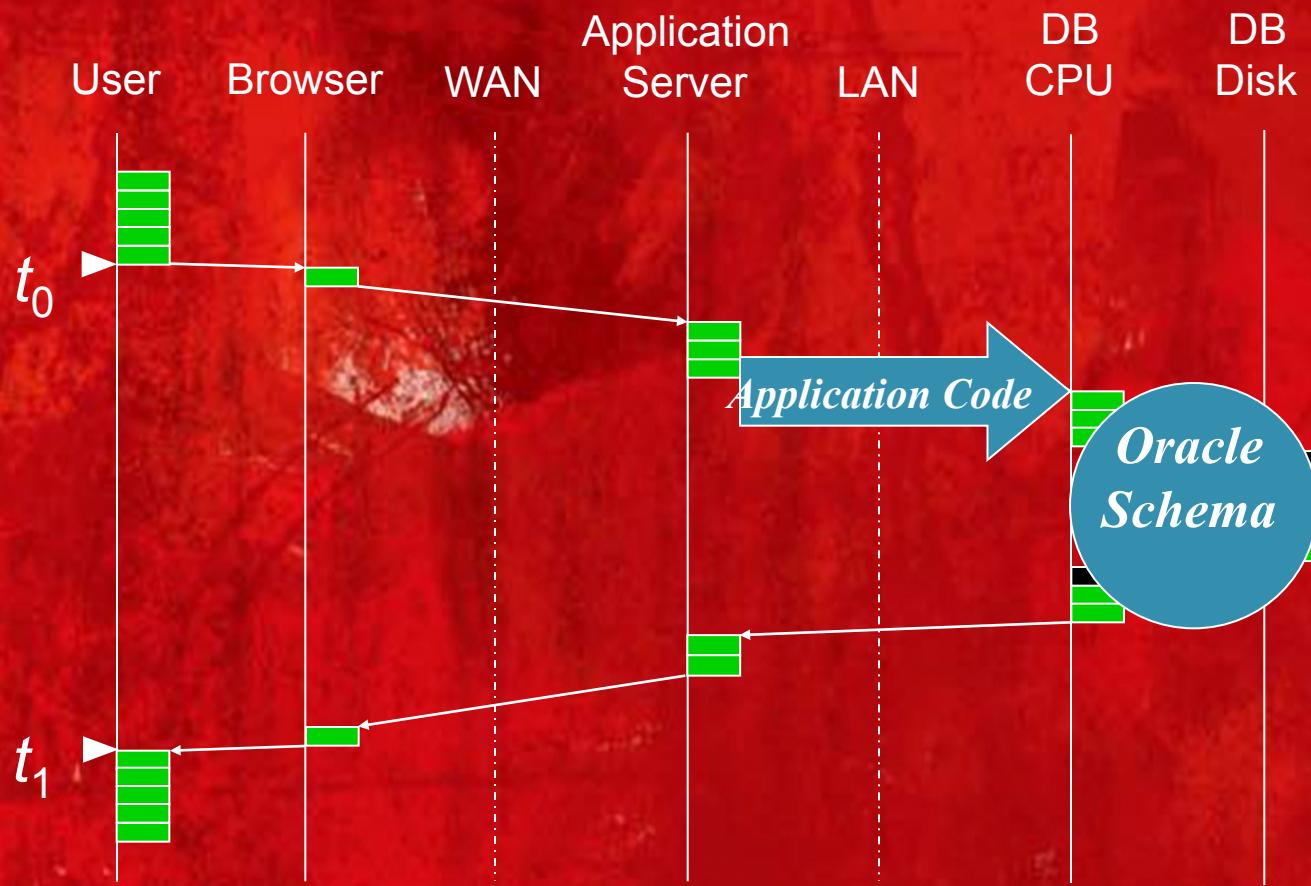
Undo

Consistency

How Oracle engine works



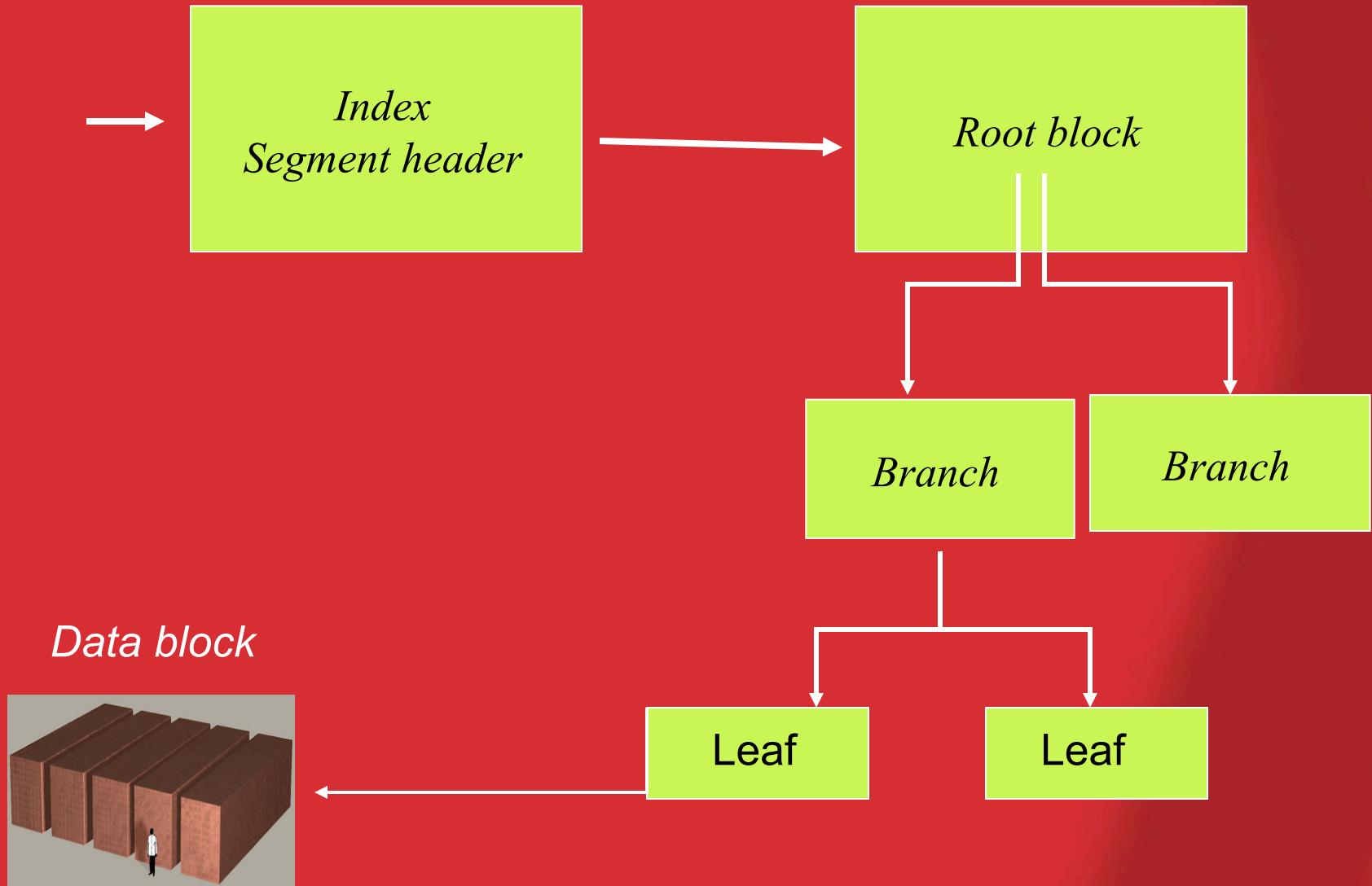
A database transaction



Steps involved

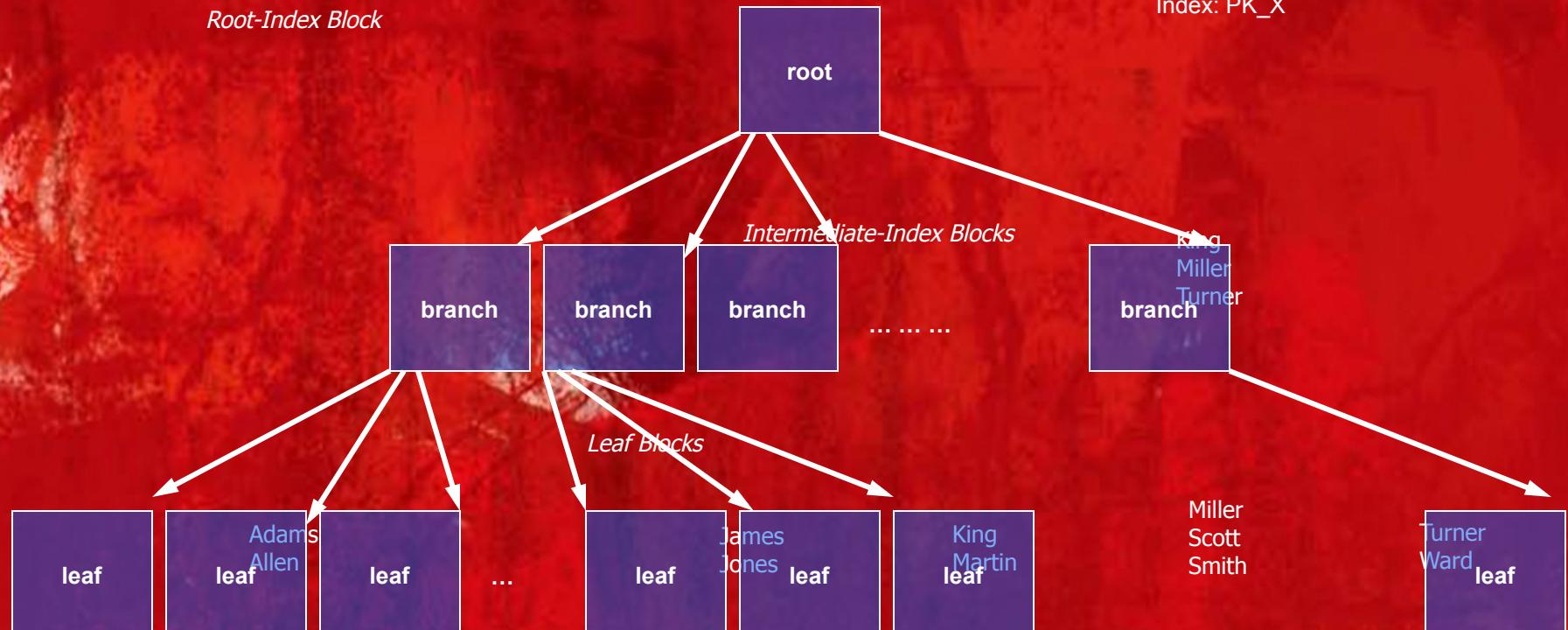
- *Parse*
- *Execute*
- *Fetch*

Index Scan



Root-Index Block

Index: PK_X



Next leaf pointers

Table: X

 \bowtie = rowid pointer

rowid lookup

Data block

Steps involved in updating a Record

Write in Log buffer

Reserve space in Undo

Write plan for rollback

Update the record

❖ *Oracle Instance*

SGA

PGA

Process Structure

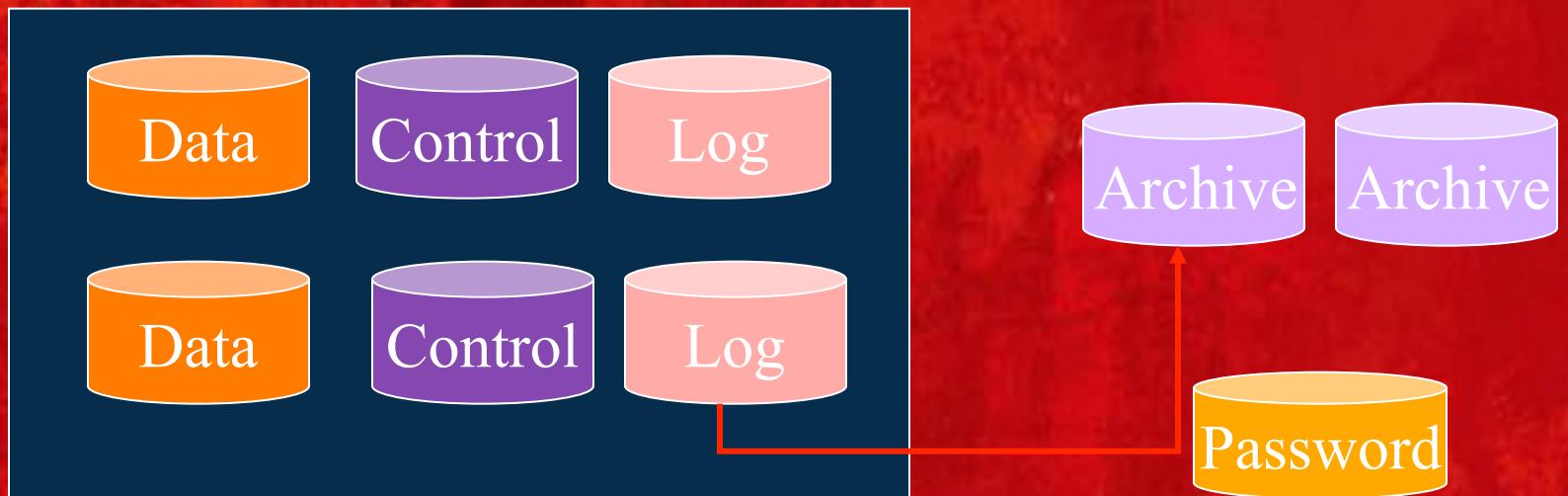
User Process

Server Process

Background process

❖ *How oracle store Data*

Oracle Database



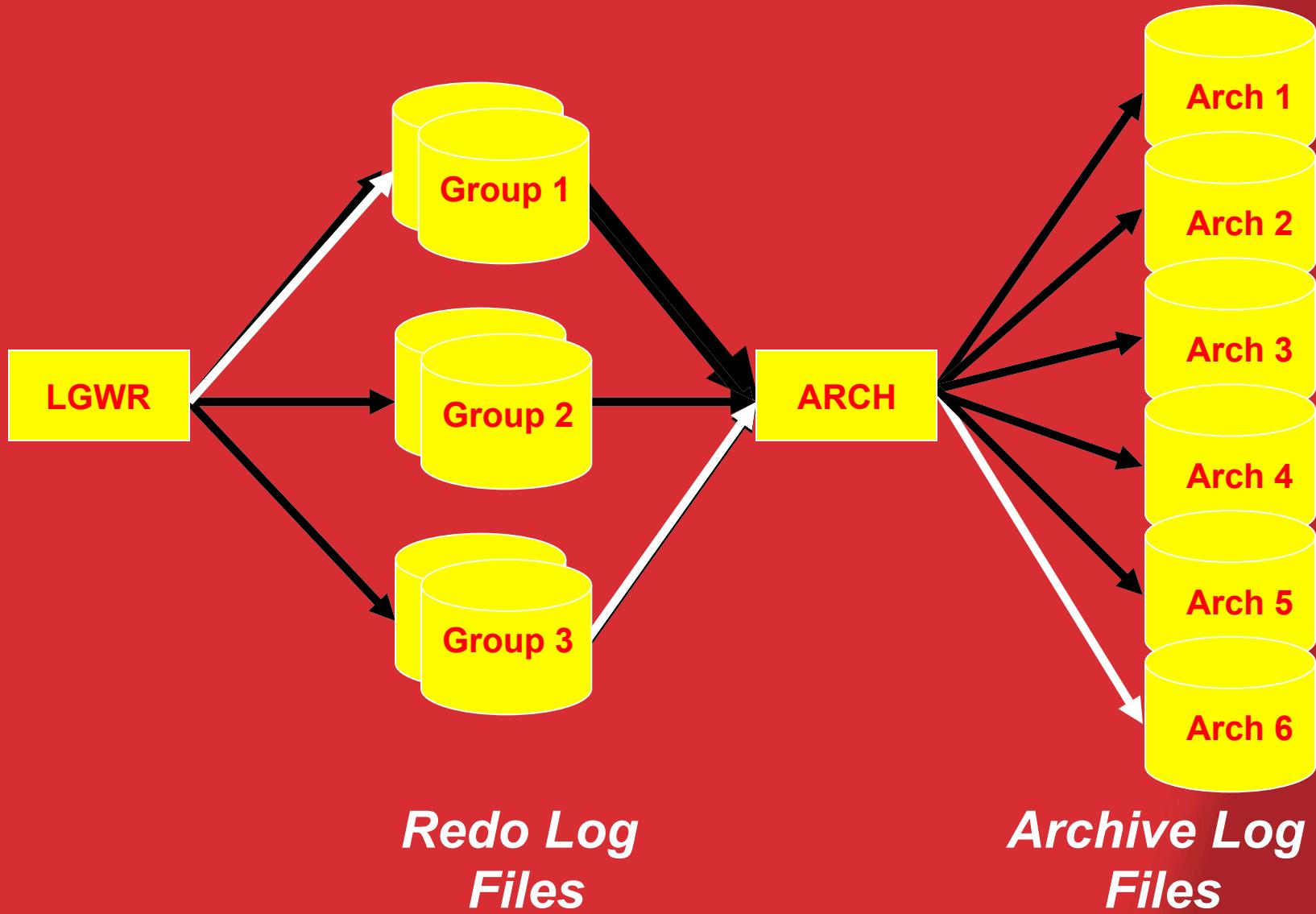


Control files

All Information about Database is stored

- 1. Name of Database*
- 2. Locations of all the files*
- 3. SCN*
- 4. CKPT CNT*

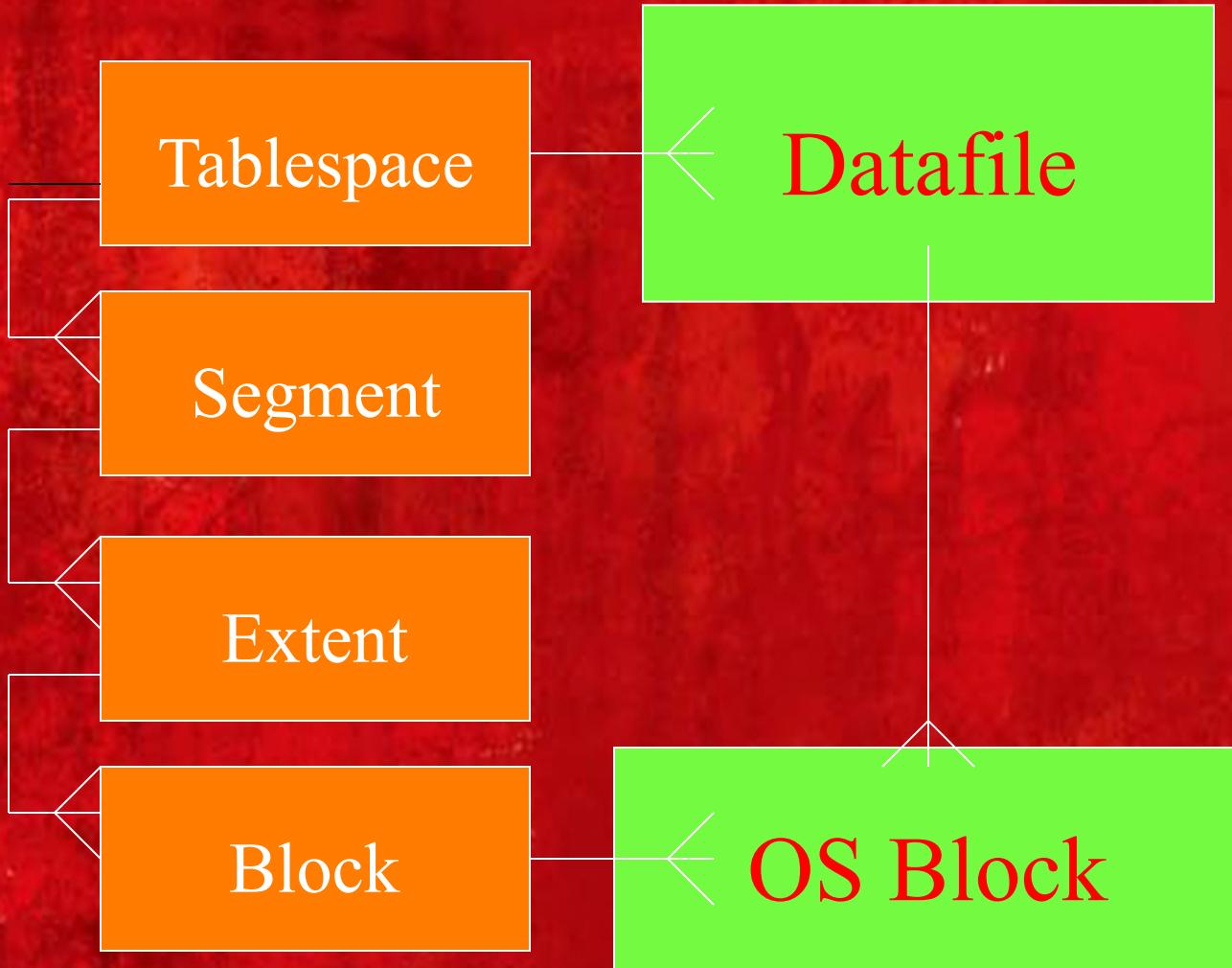
Archive files



Data Files

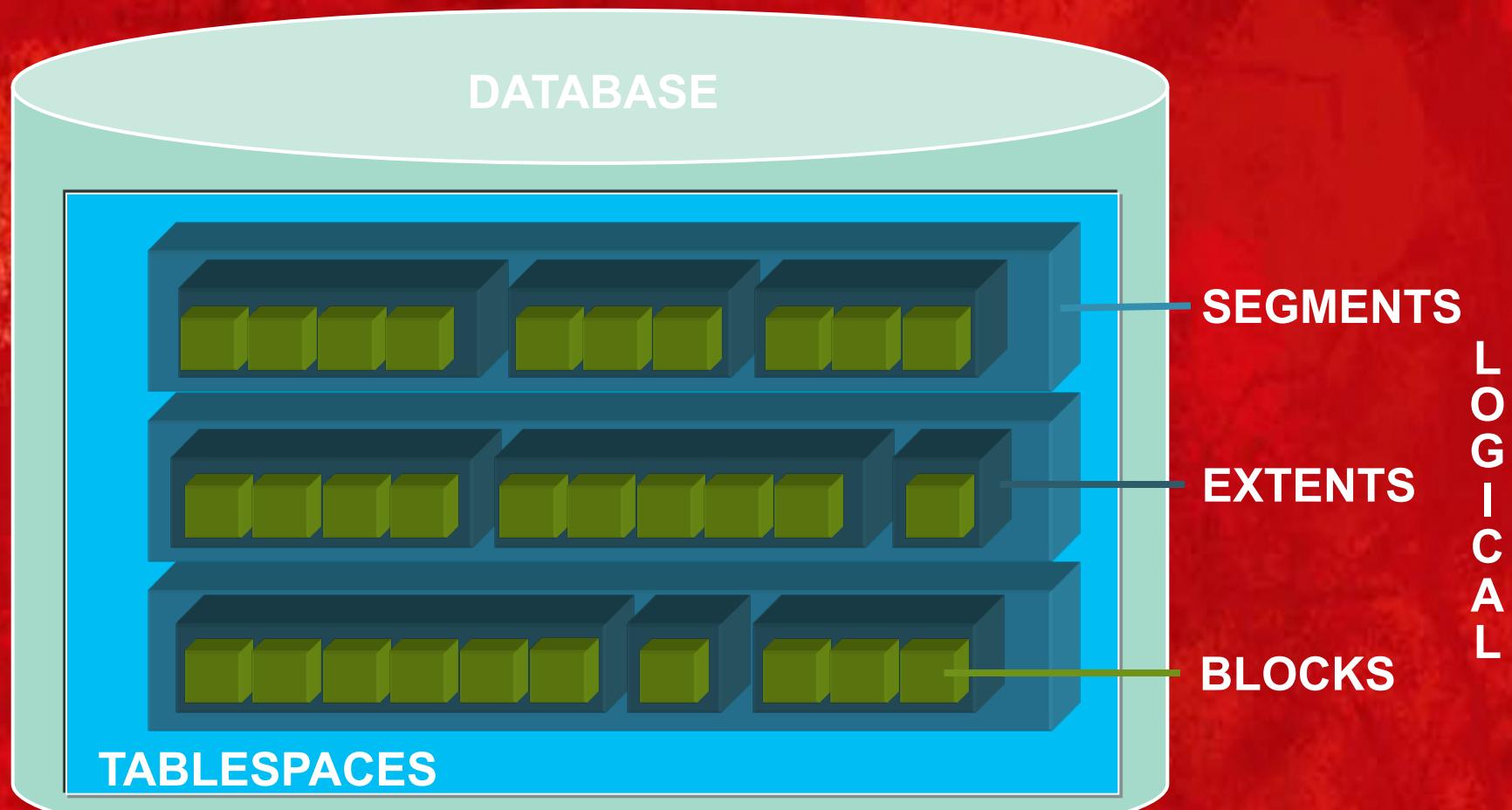


Structures

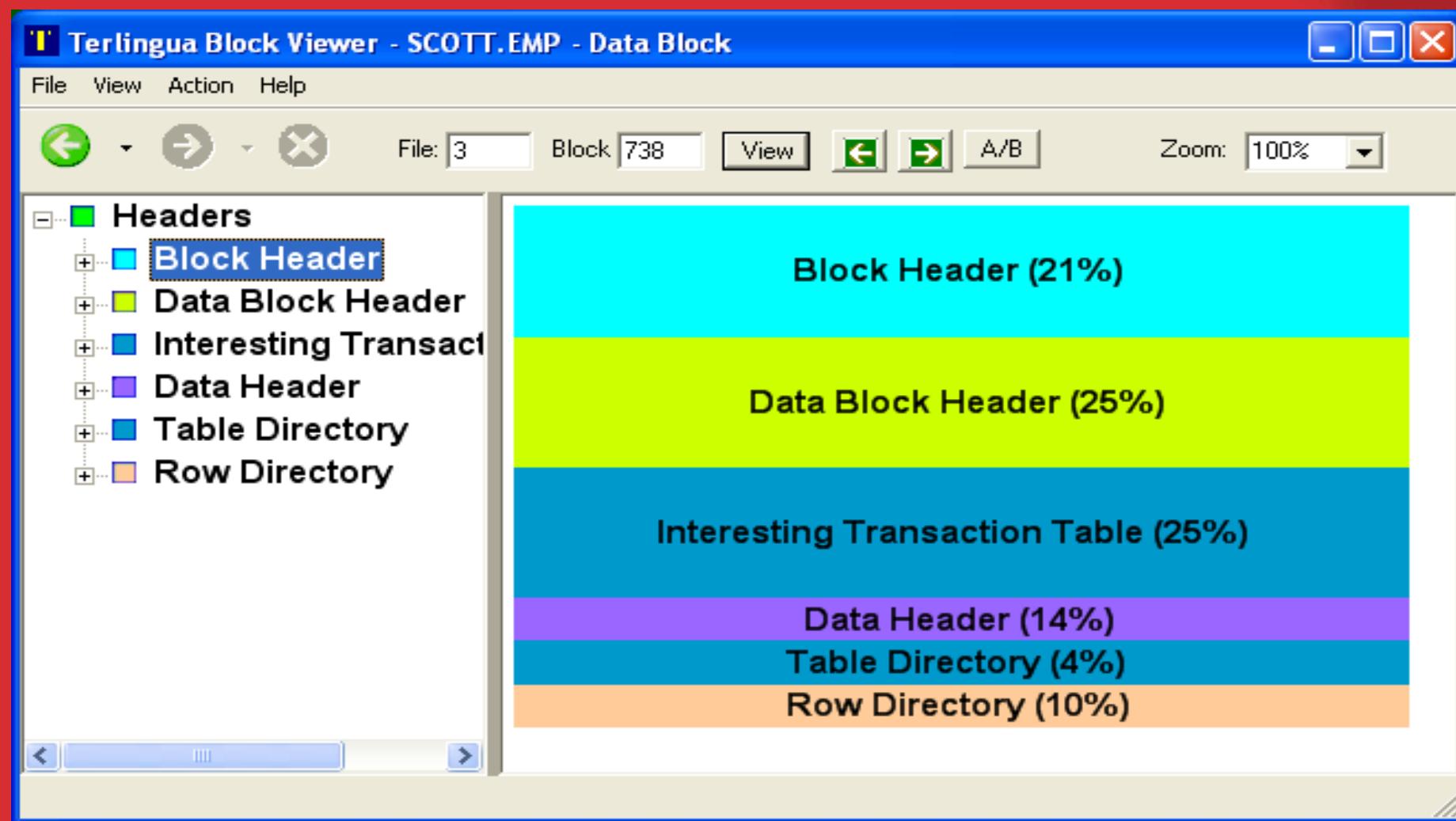


How Oracle store your Data

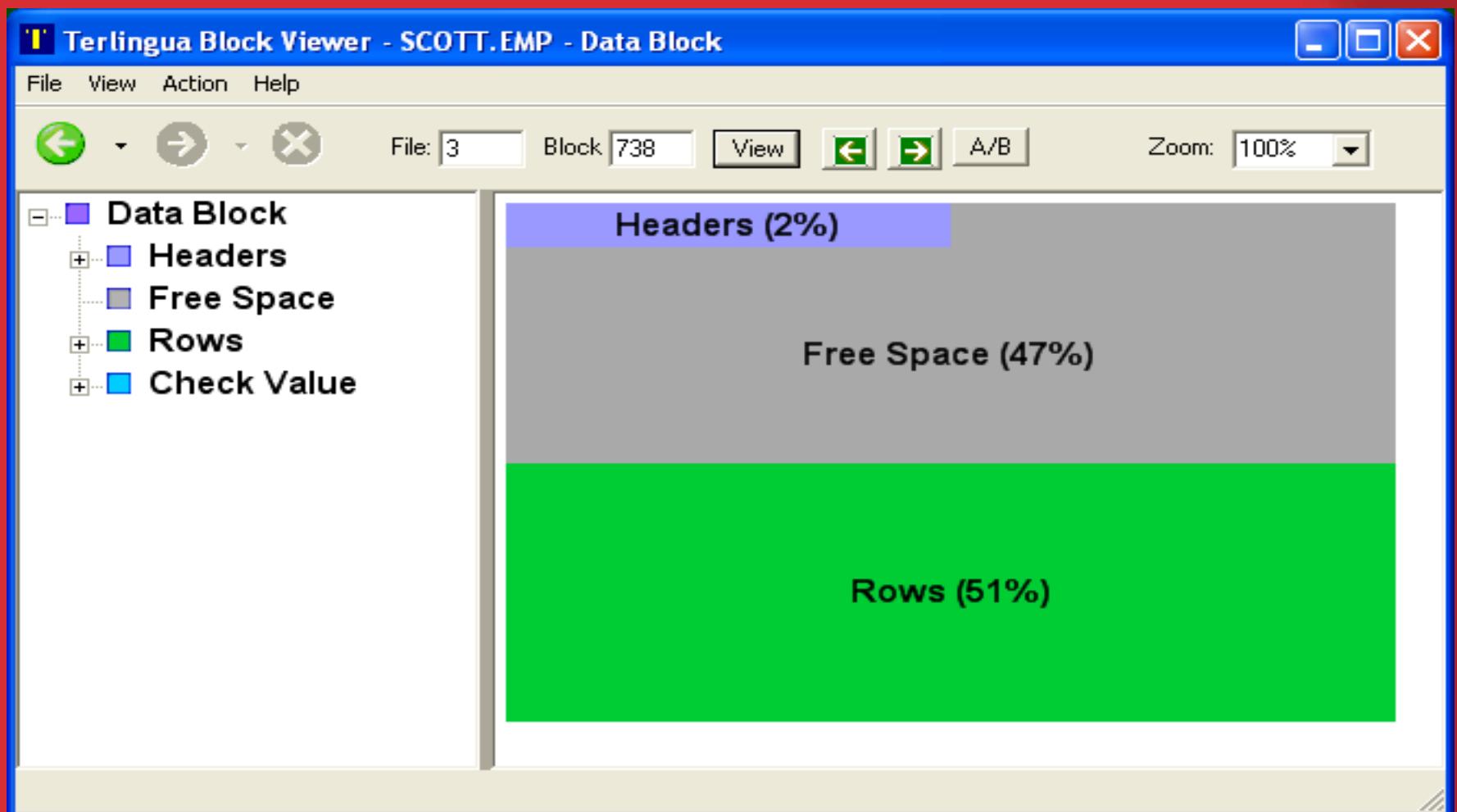
Physical



Datablocks - Organization



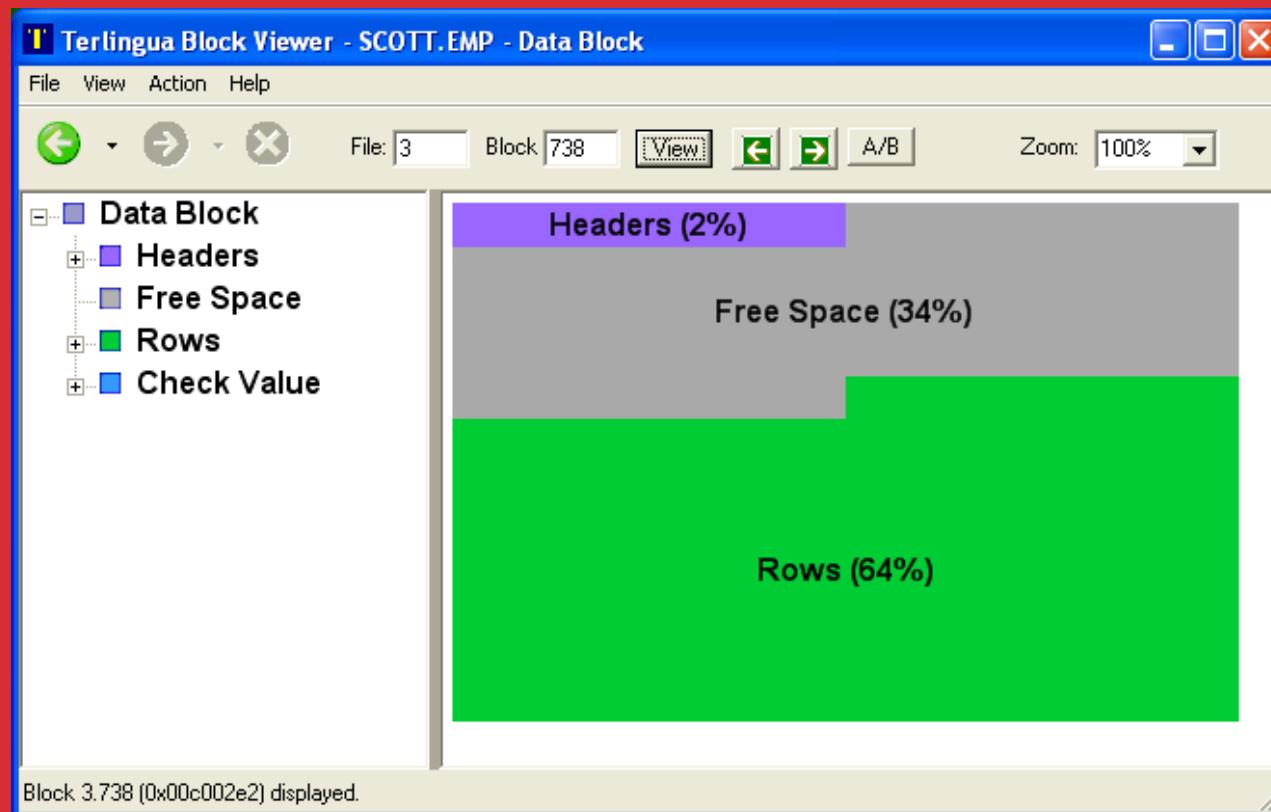
Data block of a table



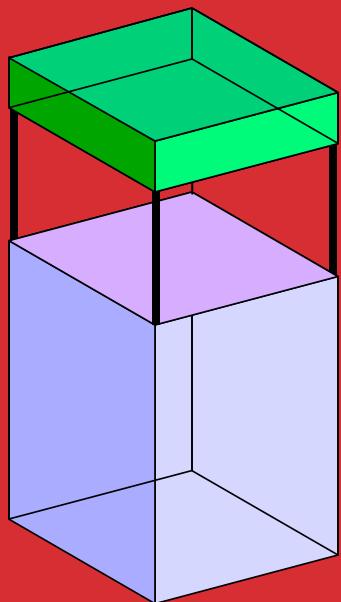
Add a Row: Free space in middle

```
SQL> insert into emp values(5, 1, 'Linda',  rpad(' ', 500));
1 row created.

SQL> commit;
Commit complete.
```



Block Utilization Parameters



INITRANS

PCTFREE

PCTUSED

FREE LISTS

Block Utilization Parameters

- **PCTFREE**

“ % of a block to be reserved for possible updates to rows that already are contained in that block.”

Block Utilization Parameters

- **PCTUSED**

when the amount of data in the block falls below this limit, Oracle considers the block available for insertion of new records.

Block Utilization Parameters

- *Intrans*

*No of transactions can happen for a block
Concurrently*

Block Utilization Parameters

- *Freelists*

Information about free blocks are stored

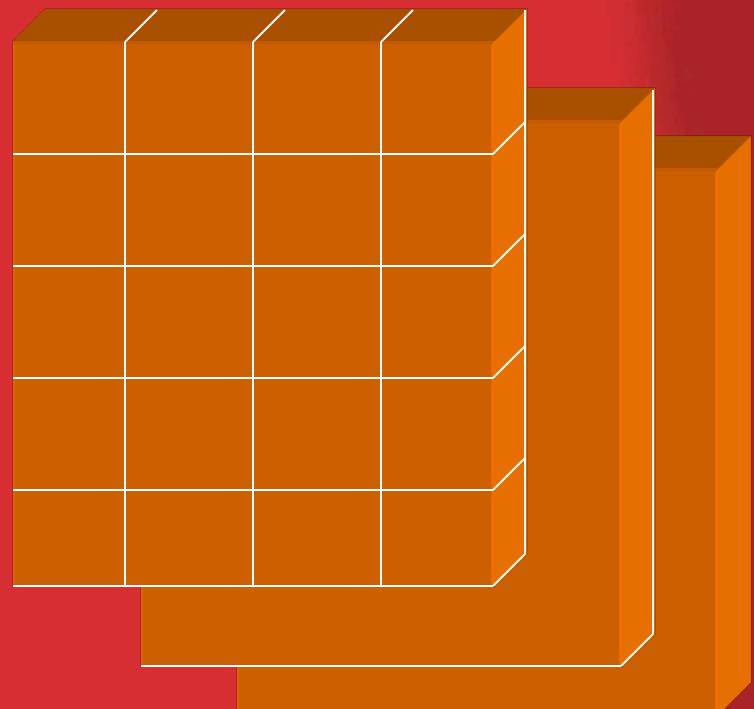
Block Utilization Parameters

*Freelists is the one which informs to
Oracle DBWR about*

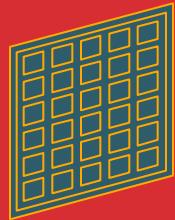
Which blocks are available for inserts

What is Space? Segment

- SEGMENT – A collection of extents
- Common segments are tables and indexes



Types of segments



Table

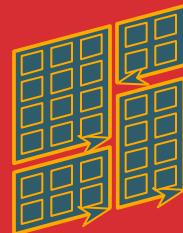
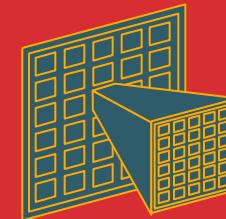
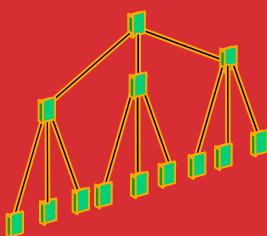


Table Partition



Nested Table



Index

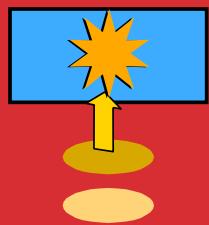


Index Partition

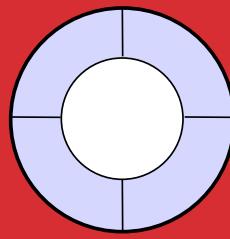


Index Org Table

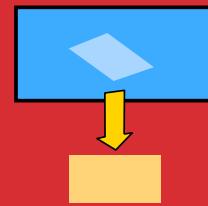
Types of segments



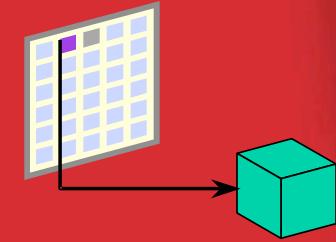
Bootstrap
segment



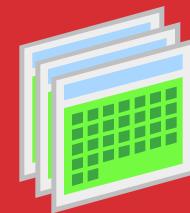
Undo
segment



Temporary
segment



LOB
segment



Cluster

Tablespaces

Tablespaces

SYSTEM SYSAUX

Data

Index

Undo

Temp

Tables

- *Tables*
- *Index Organized Table*
- *Partitioned Table*
- *External table*
- *Temporary Table*

Data Types



VARCHAR



Number



DATE



TIMESTAMP

('2011-11-23 08:00:00', '3:00')

9:00:56 PM



Difference of CHAR and VARCHAR2

INSERT INTO CRICKETS VALUES

(1,'sachin','mum','10-jan-1974');

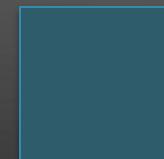
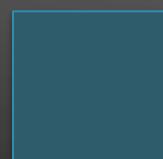
CHAR2(5),

VARCHAR2(5)

M

U

M



M

U

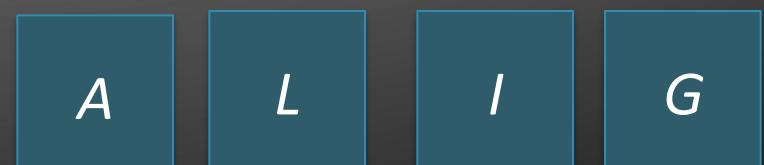
M

INSERT INTO CRICKETS VALUES

(1,'sharma','ALIG','10-jan-1974');

CHAR2(5),

VARCHAR2(5)



INSERT INTO CRICKETS VALUES

(1,'*dravid*', 'ALIGA', '10-jan-1974');

CHAR2(5),

VARCHAR2(5)

A

L

I

G

A

A

L

I

G

A

~~INSERT~~ ■ INTO CRICKETS VALUES

(1,'dravid','ALIGAR','10-jan-1974');

CHAR2(5),

Error --- Too many values



VARCHAR2(5)

Error-- Too many values

Select the Data



One Record

More than one

select

Which columns do you want to see in the result?

from

Select

from which table(s)

Where

What is the condition to filter the rows?

*Select * from vegetables where
name='carrot'*

order by

In which order do you want to see the resulting rows?

group by

How should the rows be grouped/aggregated?

having

What is the condition to filter the aggregated groups?

Display Structure

DESCRIBE



External table

*create or replace directory data_dir as
'/tmp/'*

```
create table external_table
(pno NUMBER(4 ) , pNAME VARCHAR2(10),
loc VARCHAR2(9), century NUMBER(4)
)
```

```
Organization external
( type oracle_loader
default directory data_dir
access parameters
( fields terminated by ',' )
location ('cricket.dat')
)
/
```

Use of External Tables

Information in text file

Information in ms excel file

Information in ms word file



Index Organized Table

Index Organised Tables

Heap-organised table

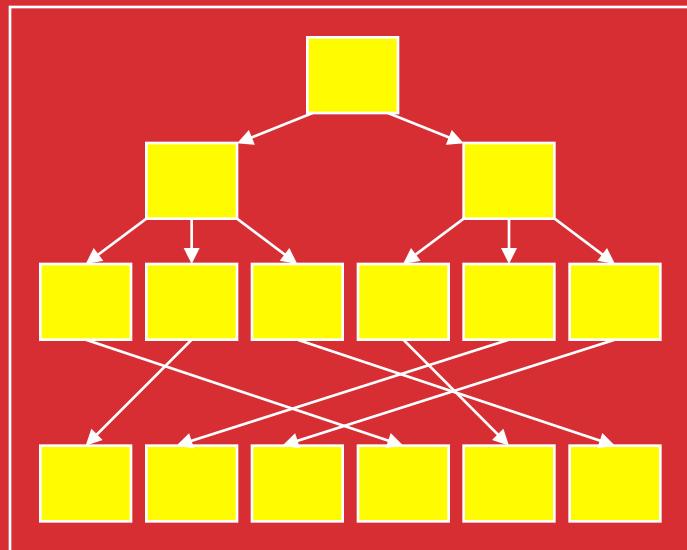
```
CREATE TABLE team  
(  
    team_key VARCHAR2(3),  
    team_name  
    VARCHAR2(50),  
    country_key  
    VARCHAR2(3)  
    CONSTRAINT team_pk  
    PRIMARY KEY  
(team_key);  
)  
ORGANIZATION HEAP;
```

Index-organised table

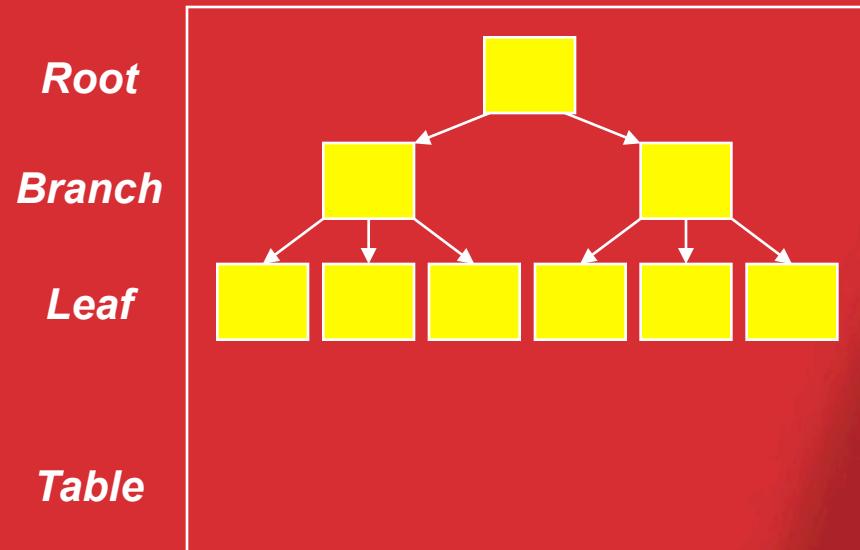
```
CREATE TABLE team  
(  
    team_key VARCHAR2(3),  
    team_name  
    VARCHAR2(50),  
    country_key  
    VARCHAR2(3)  
    CONSTRAINT team_pk  
    PRIMARY KEY  
(team_key);  
)  
ORGANIZATION INDEX;
```

Index Organised Tables

Heap-organised table



Index-organised table





Partitioned Table

Partition Types

Range

List

Hash

combine

Range Partitioning

*CREATE TABLE DEPT (Dno NUMBER(2), DNAME
VARCHAR2(30))*

PARTITION BY RANGE(DNO)

*(partition D1 values less than (10) tablespace TS1,
partition D2 values less than (20) tablespace TS2,
partition D3 values less than (MAXVALUE)
tablespace TS3);*

Range Partitioning

Insert into dept values (1, ‘DEPT 1’);

Insert into dept values (7, ‘DEPT 7’);

Insert into dept values(10, ‘DEPT 10’);

Insert into dept values (15, ‘DEPT 15’);

Insert into dept values (22, ‘DEPT 22’);

Range Partitioning

```
CREATE TABLE PARTITION_BY_RANGE
(BIRTH_MM INT NOT NULL,
BIRTH_DD INT NOT NULL,
BIRTH_YYYY INT NOT NULL)
PARTITION BY RANGE (BIRTH_YYYY, BIRTH_MM,
BIRTH_DD)
(PARTITION P1 values less than (1970,01,01)
tablespace TS1
,PARTITION p2 values less than
(maxvalue,maxvalue,maxvalue) tablespace TS2)
```

List Partitioning

List Partitioning

```
CREATE TABLE PARTITION_BY_LIST
(DEPTID          NUMBER,
DEPTNAME        VARCHAR2(15),
STATE           VARCHAR2(2))

PARTITION BY LIST (STATE)
(PARTITION DEPTS_IN_NORTH VALUES ('AK')
TABLESPACE TS1,
...
PARTITION DEPTS_WITH_NO_REGION VALUES (DEFAULT)
TABLESPACE TS2)
```

Hash Partitioning

*Create table hash_partition (account_id
varchar2(30)) partition by HASH
(account_id) partitions 16*

Hash Partitioning

```
CREATE TABLE PARTITION_BY_HASH  
(FIRST_NAME VARCHAR2(10),  
LAST_NAME  VARCHAR2(10),  
AGE number )  
PARTITION BY HASH (AGE)  
(PARTITION P1_AGE TABLESPACE ts1,  
PARTITION P2_AGE TABLESPACE ts2)
```

```
CREATE TABLE CRICKET (
PNAME VARCHAR2(20),
LOCATION VARCHAR2(20),
DOB DATE,
CENTURIES NUMBER) ;
```

*INSERT INTO CRICKET VALUES
(‘sachin’,’mum’,’10-jan-1974’,50);*

*INSERT INTO CRICKET VALUES
(‘harbhjan’,’mum’,’10-jul-1979’,1);*

*INSERT INTO CRICKET VALUES
(‘dravid’,’ban’,’10-mar-1974’,35);*

*INSERT INTO CRICKET VALUES
(‘ganguly’,’cal’,’10-feb-19970’,40);*

1. rollback ;

*2. Select * from cricket*

*INSERT INTO CRICKET VALUES
(‘sachin’,’mum’,’10-jan-1974’,50);*

*INSERT INTO CRICKET VALUES
(‘harbhjan’,’mum’,’10-jul-1979’,1);*

*INSERT INTO CRICKET VALUES
(‘dravid’,’ban’,’10-mar-1974’,35);*

*INSERT INTO CRICKET VALUES
(‘ganguly’,’cal’,’10-feb-1970’,40);*

1. commit ;

*2. Select * from cricket*

*Select * from cricket*

Pname	Location	Dob	centuries
Sachin	Mum	10-jan-1974	50
harbhajan	Mum	21-mar-1973	1
Dravid	Ban	10-mar-1974	35
ganguly	cal	10-feb-1970	40

Update



*Update cricket set centuries=5 /
where pname='sachin'*

*Select * from cricket*

Update



*Update cricket set location='Del'
where pname='dravid'*

*Select * from cricket*

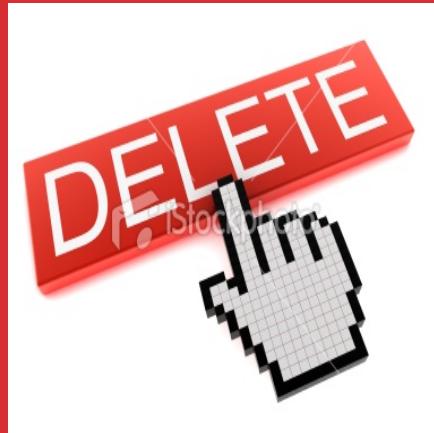
Delete



*Delete from cricket where
pname='dravid'*

*Select * from cricket*

Delete



*Delete from cricket where
location='mum'*

*Select * from cricket*

Truncate table cricket

*Select * from cricket*

Drop table cricket

*Select * from cricket*

```
CREATE TABLE CRICKET (
PNAME VARCHAR2(20),
LOCATION VARCHAR2(20),
DOB DATE
CENTURIES NUMBER)) ;
```

*INSERT INTO CRICKET VALUES
(‘sachin’,’mum’,’10-jan-1974’,50);*

*INSERT INTO CRICKET VALUES
(‘harbhjan’,’mum’,’10-jul-1979’,1);*

*INSERT INTO CRICKET VALUES
(‘dravid’,’ban’,’10-mar-1974’,35);*

*INSERT INTO CRICKET VALUES
(‘ganguly’,’cal’,’10-feb-19970’,40);*

1. commit ;

*2. Select * from cricket*

Adding a column in existing table

Alter table cricket add wickets number ;

Desc cricket

Remove column in an existing table

Alter table cricket drop column wickets ;

Desc cricket

*Select * from cricket*

Pname	Location	Dob	centuries
Sachin	Mum	10-jan-1974	50
harbhajan	Mum	21-mar-1973	1
Dravid	Ban	10-mar-1974	35
ganguly	cal	10-feb-1970	40

Change column in an existing table

*Alter table cricket rename column pname to
playername ;*

Desc cricket

Change the table name

Alter table cricket rename to hockey ;

*Select * from cricket ;*

*Select * from hockey ;*

Index Organised Tables

Heap-organised table

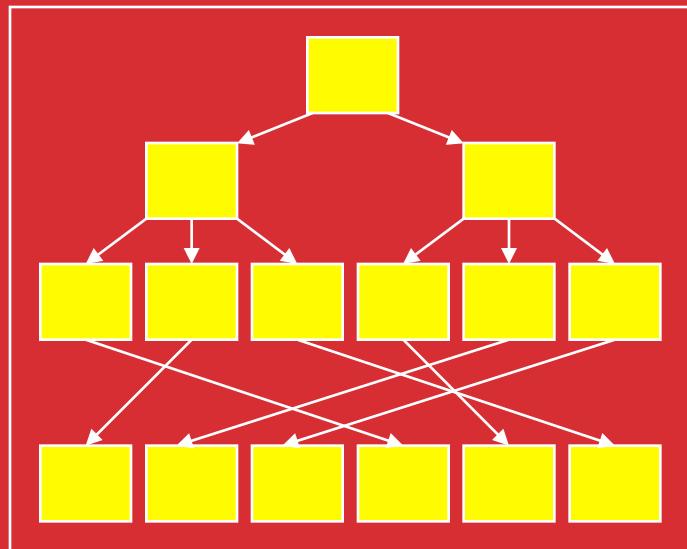
```
CREATE TABLE team  
(  
    team_key VARCHAR2(3),  
    team_name  
    VARCHAR2(50),  
    country_key  
    VARCHAR2(3)  
    CONSTRAINT team_pk  
    PRIMARY KEY  
(team_key);  
)  
ORGANIZATION HEAP;
```

Index-organised table

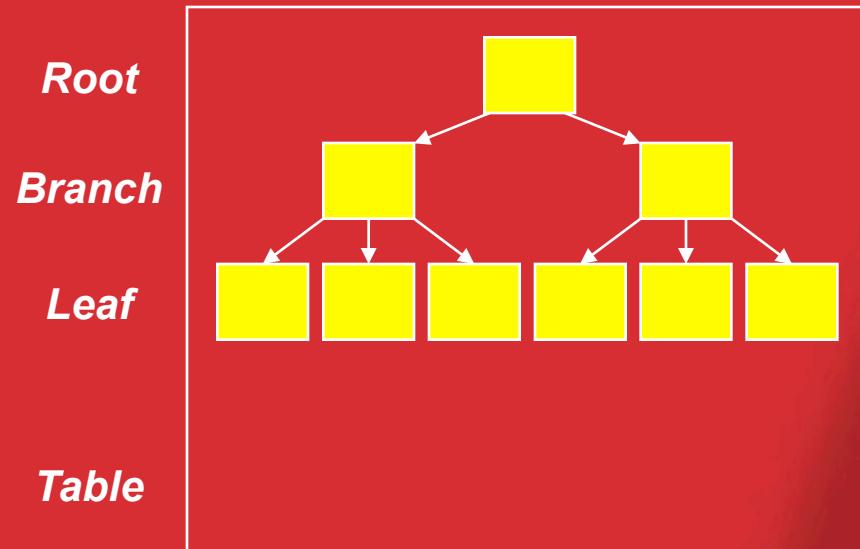
```
CREATE TABLE team  
(  
    team_key VARCHAR2(3),  
    team_name  
    VARCHAR2(50),  
    country_key  
    VARCHAR2(3)  
    CONSTRAINT team_pk  
    PRIMARY KEY  
(team_key);  
)  
ORGANIZATION INDEX;
```

Index Organised Tables

Heap-organised table



Index-organised table



Desc user_objects

Create table iottable

```
CREATE TABLE team
(
    team_key VARCHAR2(3),
    team_name
    VARCHAR2(50),
    country_key
    VARCHAR2(3)
    CONSTRAINT team_pk
    PRIMARY KEY
    (team_key);
)
ORGANIZATION INDEX;
```

*Insert into iottable select * from
user_objects*

commit

Constraints

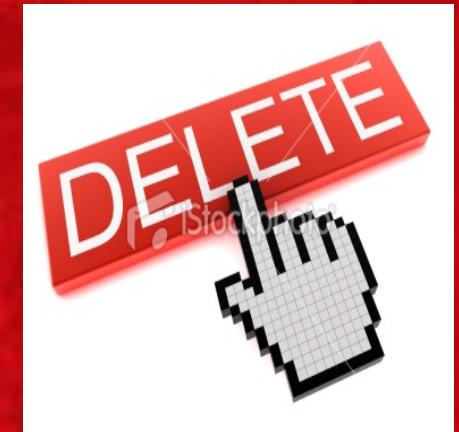
Types of Constraints

Constraint	Description
NOT NULL	<i>Specifies that a column cannot contain null values</i>
UNIQUE	<i>Designates a column or combination of columns as unique</i>
PRIMARY KEY	<i>Designates a column or combination of columns as the table's primary key</i>
FOREIGN KEY	<i>Designates a column or combination of columns as the foreign key in a referential integrity constraint</i>
CHECK	<i>Specifies a condition that each row of the table must satisfy</i>

Data Definition Language



Data Manipulation Language



Data Manipulation Language



Transaction Control



Transaction Control

Commit



Arithmetic expressions





Operators and Functions

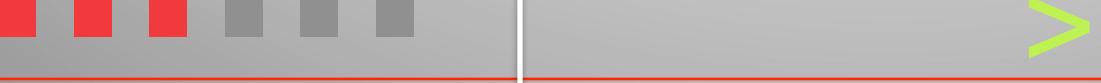


*Select * from friends*



Whose salary is 30000

*Select * from friends
Where sal = 30000*



*Whose salary is greater than
25000*

Where $sal > 25000$



\geq

*salary is greater than or
Equal to 25000*

Where $sal \geq 25000$



*Whose salary less than
20000*

Where sal < 20000



\leq

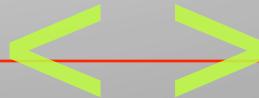
*salary is less than or equal
to 25000*

Where $sal \leq 25000$



Who are all MALES

Where Gender='M'



Who are all not males

Where Gender < > 'M'



!=

Who are all not males

Where Gender != 'M'



$\wedge =$

Who are all not males

Where Gender $\wedge = 'M'$

All three have same meaning

< >

!= You can use any one of this

^=



Whose eno is 103

Where eno = 103



Whose eno are less than 103

Where eno < 103



\leq

*Whose eno are less than
Or equal to 103*

Where eno ≤ 103



>

*Whose eno are greater
than 103*

Where eno > 103

\geq

*Whose eno are greater
than or equal to 103*



< >

Whose eno is not 103

Where eno < > 103



Whose eno is not 103

Where eno != 103



$\wedge =$

Whose eno is not 103

Where eno $\wedge \neq 103$



Next excercise



IN

How to use --IN



IN

Where Degree in
('BA','BCA');



IN

*Where location in
('delhi','chennai');*



IN

Where ename in
(‘neha’,’arjun’);



IN

Who has eno 101 103 105

Where eno in (101,103,105)



Next excercise



NOT IN

How to use -NOT IN



NOT IN

Whose eno is not 101,103

*Where eno not in
(101,103)*



NOT IN

*Who are all not done BA
And MBBS*

*Where degree not in
(‘BA’, ‘MBBS’)*



NOT IN

*Where location not in
(‘chennai’,’delhi’,’mumbai’)*



NOT IN

I don't need Ajay and Tabu

Where ename not in
('AJAY', 'TABU')



NOT IN

No need of DBA ,developer

*Where job not in
('DBA', 'Developer')*



NOT IN

I don't need Anil and Arjun

Where ename not in
('ANIL', 'ARJUN')



Doubts

Can I use 2 Operators

For example > and <

> and in

> and not in



Doubts

Yes

Multiple conditions can be used



AND

How to use – AND



AND

Where gender=Male

AND *JOB=MBBS*

Both conditions should be true

If one fails you don't get correct



*Where job not in ('DBA',
'Developer') and gender='M'*



*Where location not in
(‘DELHI’, ‘CHENNAI’) and
gender=‘M’*



Where $ENO < 103$ and
location not in('CALCUTTA')



Where $ENO > 103$ and
location not in('CHENNAI')



Where $ENO > 103$ and
location not in('CHENNAI')



OR

How to use – OR



OR

Where gender=Male

or JOB=MBBS

One condition should be true



Where *job* = ('DBA',
'Developer') or *gender* = 'M'



Where *location* = ('DELHI', '
CHENNAI') or *gender*='M'



Where $ENO < 103$ or location
= ('CALCUTTA')



Where $ENO < 103$ OR job=
(‘DBA’)



Where GENDER =‘M’ or job= (‘DBA’)



*We will work with few more
operators*



Select sname

```
,    instr(sname,'l')    as instr1  
,    instr(sname,'l',3)  as instr2  
,    instr(sname,'l',3,2) as instr3  
from kidsnew;
```



Select instr(Mississippi, 'is', 1 ,
2) from dual

~~Doubts on Instr~~ --Read the below

1. *What you want to read*
2. *FROM which position*
3. *2nd OCCURANCE*



Why I am learning these commands

To have better display of my results



```
select dname
,      substr(dname,4)    as substr1
,      substr(dname,4,3)  as substr2
,      instr(dname,'l')   as instr1
,      instr(dname,'l',5) as instr2
,      instr(dname,'l',3,2) as instr3
from scott .EMP;
```

■ select bdate

, extract(year from DOB) as

year_of_birth

, extract(month from DOB) as

month_of_birth

, extract(day from DOB) as

day_of_birth

From KIDS



*Select ename,
months_between
(sysdate,DOB)
from KIDS*



```
Select add_months('29-JAN-1996', 1)
add_months_1
,    add_months('29-JAN-1997', 1)
add_months_2
,    add_months('11-AUG-1997', -3)
add_months_3
from dual;
```

~~After 148 there is division symbol~~

select greatest

*(12*6, 148/2, 73)*

*, least (12*6, 148/2, 73)*

from dual ;



```
select sysdate as today  
,  to_char  
(sysdate,'hh24:mi:ss') as time  
, to_char(sysdate, 'Day')  
from dual ;
```



```
select to_char(sysdate,'DAY  
Dy') as day  
,    to_char(sysdate,'MONTH  
mon') as month  
from dual ;
```



*Select round (12.355,2) ,
Round (259.99, -1)From dual ;*

*Note : (-1) means round to one
digit to the left of the decimal
--Nearest tens*



*Select trunc (12.355143,2)
From dual*

*What does Trunc do
will round up the number with
only 2 decimals*

~~"Select mod (9,3) , mod(10,3)~~

From dual ;

*What does mod do
will round up the number with
only 2 decimals*

Sir I want to see the date of
next Saturday

Select sysdate from dual ;

Select next_day ('10-DEC-11', 'Saturday')
from dual

Sir how can see the last day of
Of the month

Select sysdate from dual ;

Select last_day ('10-DEC-11')
from dual

```
■ ■ ■ ■ ■  
select sysdate  
, next_day(sysdate,'SAT') as  
next_sat  
, last_day(sysdate) as last_day  
, round(sysdate,'YY') as round_yy  
from dual ;
```



*We will work with few more
operators*



```
select sname  
,ltrim(ename,'S') as ltrim_s  
,rtrim(ename,'A') as rtrim_s  
from kidsnew
```

*Use upper case or lower case
depends what is your value*

LTRIM($t[,k]$)

Remove characters from the left

RTRIM($t[,k]$)

Remove characters from the right



*select power(2,3), power
(-2,3) ,
mod(8,3), mod(13,0)
from dual;*



*Select mod(9,3),
mod(10,3), mod(11,3)
from dual;*



SELECT RPAD('SENTHIL',5,'X') FROM DUAL;

SELECT RPAD('SENTHIL',4,'X') FROM DUAL;

SELECT RPAD('SENTHIL',3,'X') FROM DUAL;



**SELECT RPAD('SENTHIL', 5) FROM
DUAL;**

**SELECT RPAD('SENTHIL', 4) FROM
DUAL;**

**SELECT RPAD('SENTHIL', 3) FROM
DUAL;**



```
SELECT LPAD('SENTHIL', 12) FROM DUAL;  
SELECT LPAD('SENTHIL', 11) FROM DUAL;  
SELECT LPAD('SENTHIL', 10) FROM DUAL;
```

SELECT LPAD('SENTHIL', 12,'x') FROM

DUAL;

SELECT LPAD('SENTHIL', 11,'x') FROM

DUAL;

SELECT LPAD('SENTHIL', 10,'x') FROM

DUAL;

SELECT LPAD('SENTHIL', 10,'\$') FROM

DUAL;



```
select lower(loc), initcap(ename)
from kidsnew
where upper(hobby) = 'CRICKET'
```

```
select lower(loc), initcap(ename)
from kidsnew
```

'



```
select sname
,      substr(sname,4)  as substr1
,      substr(sname,4,3)   as substr2
From kidsnew
```



Select sname

```
,    instr(sname,'l')    as instr1  
,    instr(sname,'l',3)  as instr2  
,    instr(sname,'l',3,2) as instr3  
from kidsnew;
```



Select instr(Mississippi, 'is', 1 ,
2) from dual

~~Doubts on Instr~~ --Read the below

1. *What you want to read*
2. *FROM which position*
3. *2nd OCCURANCE*



What is dual table

- 1.Sys is the owner*
- 2.Single column and single record*
- 3.Work as calculator and calendar*



Select 3 + 4 from dual

Select 3-2 from dual

*Select 3 * 2 from dual*

Select 4 / 2 from dual;



Why I am learning these commands

To have better display of my results



```
select dname
,      substr(dname,4)    as substr1
,      substr(dname,4,3)  as substr2
,      instr(dname,'l')   as instr1
,      instr(dname,'l',5) as instr2
,      instr(dname,'l',3,2) as instr3
from scott.employees;
```

■ select bdate

, extract(year from bdate) as
year_of_birth
, extract(month from bdate) as
month_of_birth
, extract(day from bdate) as
day_of_birth
from scott.employess where
ename='SMITH'



Select ename,
months_between
(sysdate,hiredate)
from scott.emp where deptno
= 10;



```
Select add_months('29-JAN-1996', 1)
add_months_1
,    add_months('29-JAN-1997', 1)
add_months_2
,    add_months('11-AUG-1997', -3)
add_months_3
from dual;
```

```
■ ■ ■ ■ ■  
select sysdate  
, next_day(sysdate,'SAT') as  
next_sat  
, last_day(sysdate) as last_day  
, round(sysdate,'YY') as round_yy  
from dual ;
```

~~After 148 there is division symbol~~

select greatest

*(12*6, 148/2, 73)*

*, least (12*6, 148/2, 73)*

from dual ;



```
select sysdate as today  
,  to_char  
(sysdate,'hh24:mi:ss') as time  
, to_char(sysdate, 'Day')  
from dual ;
```



```
select to_char(sysdate,'DAY dy  
Dy') as day  
,    to_char(sysdate,'MONTH  
mon') as month  
from dual ;
```



Division

Multiplication



SELECT age/2 FROM kids;

*SELECT age * 5 kids ;*



Addition Subtraction



Select age + 5 from kids;

select age - 3 from kids;



```
SELECT CONCAT (CONCAT  
 (SNAME, ' is '),look)  
 FROM kids  
 WHERE sname= 'PRIYA';
```



```
SELECT CONCAT (CONCAT  
(SNAME, ' is '),look)  
FROM kids  
WHERE sname= 'JOHN';
```



```
SELECT CONCAT (CONCAT  
(SNAME, ' is '), look)  
FROM kids  
WHERE sname= 'SACHIN';
```



```
SELECT CONCAT (CONCAT  
(SNAME, ' is '), look)  
FROM kids  
WHERE sname= 'ISHA';
```



```
SELECT CONCAT (CONCAT  
(SNAME, ' is '),look)  
FROM kids  
WHERE sname= 'SUNIL';
```



```
SELECT sname || age|| loc  
|| hobby "Concatenation"  
FROM kids;
```



*SELECT sno || age|| loc ||
look "Concatenation"
FROM kids;*



LIKE Operator



*You know only few
letters of a word but
you want
To see all*

USE LIKE Operator



Where shame like
‘S%’

Where hobby like
‘%rr%’



Where look like

‘%i%’

Where loc like

‘%e%’



*Select * from kids*

Order by sname

*Select * from kids*

Order by loc



*Select * from kids*

Order by sname asc

*Select * from kids*

Order by desc



Asc =least or lowest

Desc =highest or greatest

Default is ASC



Select distinct Loc from kids



Create table kidsnew

Use the same columns of kids

■ ■ ■ ■ ■
Insert all the five Kid's records

*and it hit **me***

Add 3 columns in kidsnew

Height number (5,2)

Weight number (5,2)

Ambition varchar2(20)

~~FOR~~ Priya

3.2

25.7

President

~~FOR~~ John

2.92

27.8

doctor

FOR Sachin

3.2

29.47

engineer

~~FOR~~ Sunil

2.44

20.25

Doctor

~~FOR~~ ISHA

3.12

25.45

Doctor

```
'SELECT CASE sname  
WHEN 'priya' THEN 'The great'  
WHEN 'isha' THEN 'the fine'  
ELSE 'The good'  
END,  
ambition  
FROM kidsnew;
```

■ ■ ■ ■ ■ ■

```
SELECT CASE LOOK
WHEN 'CUTE' THEN 'The lovable'
WHEN 'serious' THEN 'not funny'
ELSE 'The good'
END,
SNAME
FROM kidsnew;
```



*First letter of the word comes
in upper case*

Select INITCAP (sname) from
newkids ;



*First letter of the word comes
in upper case*

Select INITCAP (LOC) from
newkids ;



*SELECT UPPER(sname)
"Uppercase" FROM kidsnew ;*



*select * from kidsnew
where upper(sname) like
('%\$%');*



**SELECT LOWER (SNAME)
FROM KIDSNEW ;**

~~SELECT LOWER (ambition)~~

FROM KIDSNEW ;

~~select round(345.678) from~~
dual ;



*Select floor (345.678) from
dual ;*



*Select ceil (345.678) from
dual ;*

```
select round(345.678), ceil  
(345.678), floor(345.678)  
from dual;
```

~~SELECT MAX(weight) FROM~~
kidsnew ;

~~SELECT MAX(height) FROM~~
kidsnew ;

~~**SELECT MAX(height) FROM
kidsnew ;**~~

**SELECT MAX(AGE) FROM
kidsnew ;**

~~■~~ **SELECT min (weight) as "Lowest weight" FROM kidsnew;**

~~SELECT~~ *min (height)* as "Lowest height" FROM kidsnew;

~~SELECT~~ min (AGE) as "Lowest
AGE" FROM kidsnew;



Select avg (height) from kidsnew ;



Select avg (weight) from kidsnew ;



Select avg (AGE) from kidsnew ;



Select sum (weight) from kidsnew ;

Select sum (height) from kidsnew ;



Select sum (weight) from kidsnew ;

Select sum (height) from kidsnew ;



Select sum (age) from kidsnew ;



*SELECT AVG (DISTINCT age)
FROM kidsnew;*



*SELECT AVG (DISTINCT height)
FROM kidsnew;*



SELECT LENGTH(LOC)

"Length in characters"

FROM kidsnew;



`SELECT LENGTH('sname')`

"Length in characters"

`FROM kidsnew;`



*SELECT LENGTH(sname)
"Length in characters"
FROM kidsnew;*



*Select replace
('JACK and JUE','J','BL') "Changes"
FROM DUAL;*



```
SELECT REPLACE  
('JACK and JUE','j','BL') "Changes"  
FROM DUAL;
```

*You are asking Oracle to change
the letter j to BL*



Select replace

('ABCDA', 'A','*') replace_A from
dual

SELECT REPLACE

('CAN and CAT','C','F') "Changes"

FROM DUAL;



Savepoint

*Example taken from
Oracle site*



create table sptest (eno number);

insert into sptest values (1);

insert into sptest values (2);

insert into sptest values (3);



savepoint sp_one;



insert into spptest values (10);
insert into spptest values (20);
insert into spptest values (30);



savepoint sp_two;



insert into sptest values (100);
insert into sptest values (200);
insert into sptest values (300);



*select * from spptest;*



rollback to sp_two;



*select * from sptest;*



insert into sptest values (111);
insert into sptest values (222);
insert into sptest values (333);



*rollback to sp_two;
select * from sptest;*



rollback to sp_one;



rollback to sp_one;

*You only find the values that
have existed before savepoint
sp_l:*



Views



Table Emp has the following information

1. *Developer (100)*
2. *DBA (20)*
3. *Manager (5)*



```
CREATE VIEW DBA
SELECT eno, name, dno, salary
FROM emp
WHERE JOB = 'DBA'
```

DBA



```
CREATE VIEW Developer
SELECT eno, name, dno, salary
FROM emp
WHERE JOB = 'DEVELOPER'
```

Developer



```
CREATE VIEW females AS
SELECT name, address, mobileno
FROM emp
WHERE gender = 'F' ;
```

Females



DBA_TABLESPACES
V\$TABLESPACE

DBA

USER_TABLES

USER

TS\$

Does not occupy space in the Database

This is created virtually on a existing table

Views can be altered

Constraints can be created

*View can be - on few columns of many tables
created*



Drop view <name>;



USER_VIEWS

~~CREATE~~ TABLE CRICKETS (

Pno number,
PNAME VARCHAR2(5),
LOCATION CHAR(5),
DOB DATE)) ;

INSERT INTO CRICKETS VALUES

5 records

Create VIEW cricket_vw1 as

select pname from crickets ;

Create VIEW cricket_vw2 as

select pname,loc from crickets

Create VIEW cricket_vw3 as

select pname,loc,dob from crickets



*Select * from cricket_vw1*

*Select * from cricket_vw2*

*Select * from cricket_vw3*