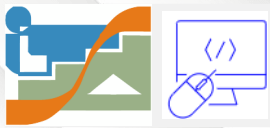


# Automatització de tasques



## DATES





## Objectius

- Definir tipus de dades de DATE (data)
- Presentació i visualització
- Operacions
- Conversió
- Extracció d'informació
- Composició de dates
- Funcions (de dates)
- Altres consideracions

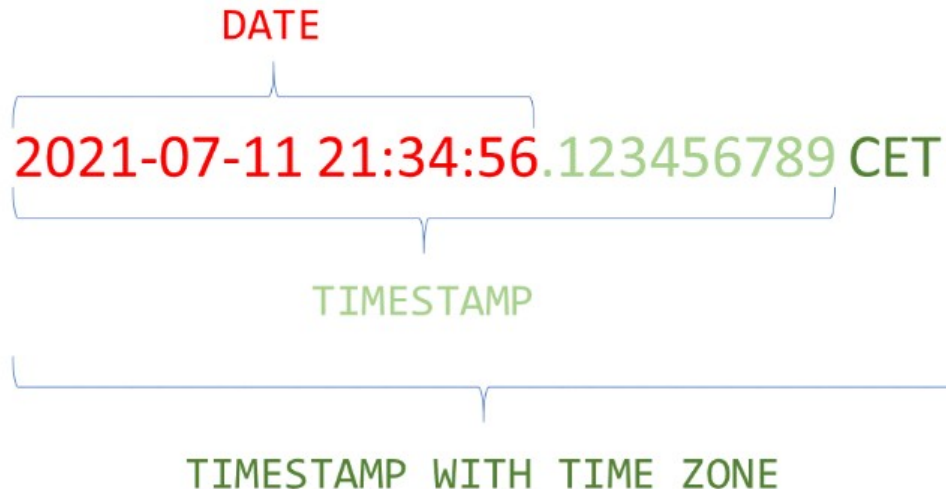


## Tipus

- DATE
- TIMESTAMP
- TIMESTAMP WITH TIME ZONE
- TIMESTAMP WITH LOCAL TIME ZONE
- INTERVAL

INTERVAL YEAR TO MONTH  
INTERVAL DAY TO SECOND

## Oracle datetime types





# Presentació

- funció sysdate -> torna tipus date

DATE  
2021-07-11 21:34:56

```
select sysdate from dual;
```

	SYSDATE
1	06/03/24

```
select to_char(sysdate,'dd-mm-yyyy hh:mi:ss') from dual;
```

	TO_CHAR(SYSDATE,'DD-MM-YYYYHH:MI:SS')
1	06-03-2024 03:58:31

```
select to_char(sysdate,'dd-mm-yyyy hh:mi:ss am') from dual;
```

	TO_CHAR(SYSDATE,'DD-MM-YYYYHH:MI:SSAM')
1	06-03-2024 04:00:15 PM

```
select to_char(sysdate,'dd-mm-yyyy hh24:mi:ss am') from dual;
```

	TO_CHAR(SYSDATE,'DD-MM-YYYYHH24:MI:SSAM')
	06-03-2024 20:31:59 PM

```
alter session set nls_date_format='dd-mm-yyyy hh:mi';  
select sysdate from dual;
```

	SYSDATE
1	06-03-2024 04:06

Prova-ho en SQL Developer



## Presentació

- funció sysdate -> torna tipus date

set SERVEROUTPUT on  
begin

```
dbms_output.put_line(sysdate);  
dbms_output.put_line(to_char(sysdate,'dd-mm-yyyy hh:mi:ss'));  
dbms_output.put_line(to_char(sysdate,'dd-mm-yyyy hh:mi:ss am'));  
dbms_output.put_line(to_char(sysdate,'dd'));  
dbms_output.put_line(to_char(sysdate,'d'));  
dbms_output.put_line(to_char(sysdate,'ddd'));  
dbms_output.put_line(to_char(sysdate,'day'));
```

end;

DATE  
2021-07-11 21:34:56

Procedimiento PL/SQL terminado correctamente.  
06-03-2024  
06-03-2024 04:10:22  
06-03-2024 04:10:22 PM  
06  
3  
066

Prova-ho en SQL Developer





# Presentació

- funció `sysdate` -> torna tipus date

set SERVEROUTPUT on  
begin

```
dbms_output.put_line(to_char(sysdate,'dd-mm-yyyy hh:mi:ss'));  
dbms_output.put_line(to_char(sysdate,'dd/mm/yyyy hh:mi:ss'));  
dbms_output.put_line(to_char(sysdate,'dd.mm.yyyy hh:mi:ss'));  
dbms_output.put_line(to_char(sysdate,'dd mm yyyy hh_mi-ss'));  
end;
```

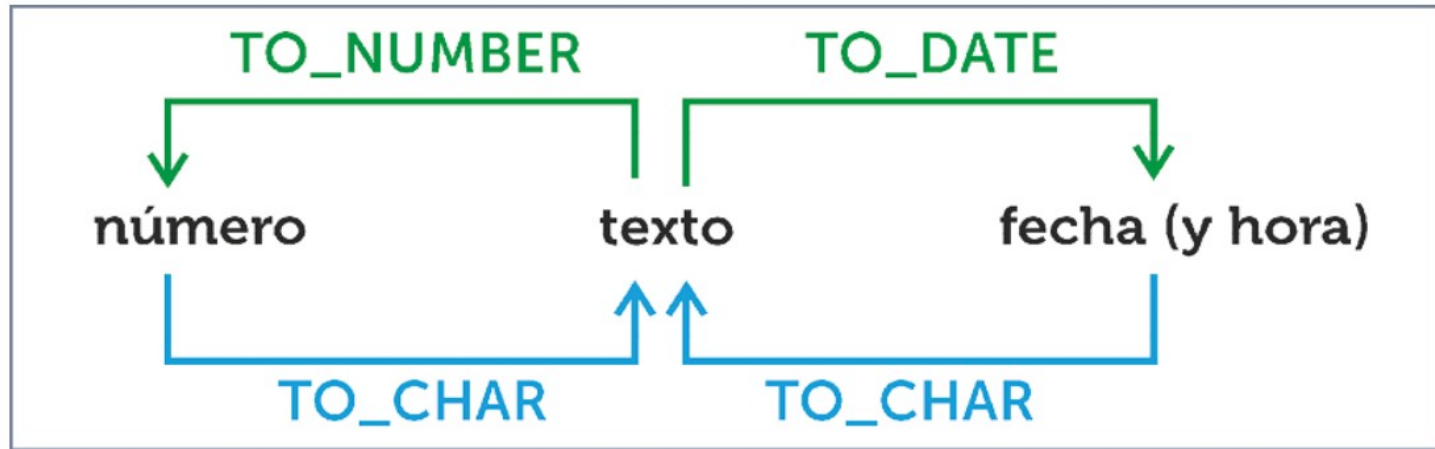
DATE  
2021-07-11 21:34:56

```
06-03-2024 08:36:21  
06/03/2024 08:36:21  
06.03.2024 08:36:21  
06 03 2024 08_36-21
```

Prova-ho en SQL Developer



## Conversions







# Conversions

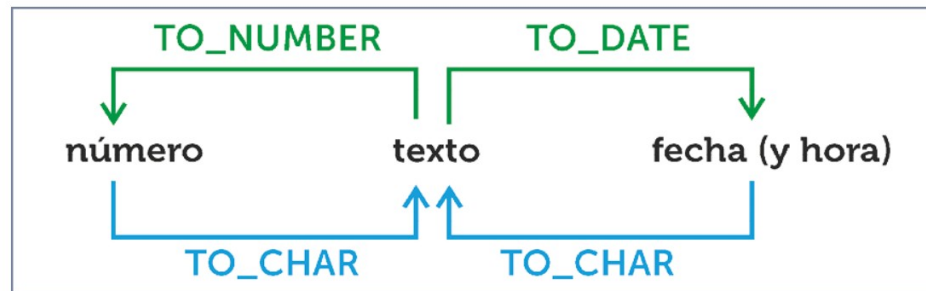
```
set SERVEROUTPUT on  
begin
```

```
  dbms_output.put_line( 5 + '6' );  
  dbms_output.put_line( '5' + '6' );  
  dbms_output.put_line(' ');  
  dbms_output.put_line( '3' || 7 );  
  dbms_output.put_line( 35 || 8 );
```

```
end;
```

```
11  
11  
  
37  
358
```

Implícita



Explícita

Prova-ho en SQL Developer



## Conversió

```
set SERVEROUTPUT on
alter session set nls_date_format='mm-dd-yyyy hh24:mi:ss';
declare    f1 date:= '1/31/2024 5:4:5';
begin
    dbms_output.put_line( f1);
    f1:= to_date ('31/3/2025','dd-mm-yyyy');
    dbms_output.put_line( f1);
end;
```

Session alterado.

01-31-2024 05:04:05

03-31-2025 00:00:00

Procedimiento PL/SQL terminado correctamente.

Prova-ho en SQL Developer



## Extracció d'informació

```
set SERVEROUTPUT on
alter session set nls_date_format='dd-mm-yyyy hh24:mi:ss';
declare    f1 date:= sysdate; dia number; mes number;
begin
    dia := to_char(f1,'dd'); mes := to_char(f1,'mm');
    dbms_output.put_line( f1 );
    dbms_output.put_line( dia );
    dbms_output.put_line( mes );
end;
```

Session alterado.

06-03-2024 16:39:31

6

3

Prova-ho en SQL Developer



## Composició de dates

```
set SERVEROUTPUT on
alter session set nls_date_format='dd-mm-yyyy hh24:mi:ss';
declare    f1 date;  dia number; mes number; anys number;
begin
    dia := 4 ;    mes := 6 ; anys :=2026;
    f1 := to_date ( dia||'-'||mes||'-'||anys, 'dd-mm-yyyy');
    dbms_output.put_line( f1 );
end;
```

```
Session alterado.
|
04-06-2026 00:00:00
```

Prova-ho en SQL Developer



## Funcions de dates ( en ps/sql)

**add\_months(data, num )** compte amb l'últim dia del mes

**last\_day(data)** l'últim dia del mes al que pertany la data

**next\_day(data,dia)** pròxim 'dia' després de data (eg dia=lunes)

**round(data[,mascara])** ..de data (year, month,

**trunc(data[,mascara])** ..de data

**extract(part from data)** ..de data

**months\_between(data1,data2)** num de mesos entre 2 dates





## add\_months

```
set SERVEROUTPUT on
alter session set nls_date_format='dd/mm/yyyy';
begin
    dbms_output.put_line( add_months('15/01/2024',1) );
    dbms_output.put_line( add_months('30/03/2024',1) );
    dbms_output.put_line( add_months('31/03/2024',1) );
    dbms_output.put_line( add_months('28/02/2024',1) );
    dbms_output.put_line( add_months('29/02/2024',1) );
    dbms_output.put_line( add_months('29/02/2024',-1) );
end;
```

```
15/02/2024
30/04/2024
30/04/2024
28/03/2024
31/03/2024
31/01/2024
```

Procedimiento PL/SQL terminado correctamente.

Prova-ho en SQL Developer



## last\_day i altres

```
set SERVEROUTPUT on
alter session set nls_date_format='dd/mm/yyyy';
begin
  dbms_output.put_line( last_day('15/01/2024'));
  dbms_output.put_line( next_day('15/01/2024', 'Lunes') );
  dbms_output.put_line( to_char( sysdate, 'dd-mm-yyyy hh24:mi' ) );
  dbms_output.put_line( round(sysdate) );
  dbms_output.put_line( trunc(sysdate) );
  dbms_output.put_line( months_between( sysdate, '31/3/2023' ) );
  dbms_output.put_line( extract(day from sysdate) );
end;
```

```
31/01/2024
22/01/2024
06-03-2024 20:14
07/03/2024
06/03/2024
11,2207590352449223416965
6
```

TIPO DATO	CAMPO DISPONIBLE
DATE	year, month, day
INTERVAL YEAR TO MONTH	year, month
INTERVAL DAY TO SECOND	day, hour, minute, second
TIMESTAMP	year, month, day, hour, minute, second

Prova-ho en SQL Developer



## Round

```
set SERVEROUTPUT on
alter session set nls_date_format='dd/mm/yyyy hh24:mi';
begin
```

```
  dbms_output.put_line( round( to_date('06/03/2024 11:00')) );
  dbms_output.put_line( round( to_date('06/03/2024 14:00')) );
  dbms_output.put_line('');
  dbms_output.put_line( round( to_date('06/03/2024'),'month' ) );
  dbms_output.put_line( round( to_date('16/03/2024'),'month' ) );
  dbms_output.put_line('');
  dbms_output.put_line( round( to_date('06/03/2024'),'year' ) );
  dbms_output.put_line( round( to_date('16/07/2024'),'year' ) );
```

```
end;
```

«The default model, 'DD', returns the date rounded or truncated to the day with a time of midnight»

```
06/03/2024 00:00
07/03/2024 00:00

01/03/2024 00:00
01/04/2024 00:00

01/01/2024 00:00
01/01/2025 00:00
```

models de formats per a ROUND i TRUNC



## - funció localtimestamp -> torna tipus timestamp

```
set SERVEROUTPUT on
declare
  vd1 date := sysdate;
  vd2 timestamp := localtimestamp;
begin
  dbms_output.put_line( to_char( 'DATE ' || vd1 ));
  dbms_output.put_line( to_char( 'DATE ' || to_char( vd1,'hh24:mi:ss' )));
  dbms_output.put_line( to_char( 'timestamp ' || vd2 ));
  dbms_output.put_line( to_char( 'timestamp ' || to_char (vd2,'hh24:mi:ss.ff6' )));
end;
```

2021-07-11 21:34:56.123456789

TIMESTAMP

Prova-ho en SQL Developer

```
DATE 16/01/25
DATE 10:13:59
timestamp 16/01/25 10:13:59,138000
timestamp 10:13:59.138000
```



### - funció localtimestamp -> torna tipus timestamp

```
set SERVEROUTPUT on
alter session set nls_date_format='dd--mm--yyyy';
alter session set nls_timestamp_format='dd-mm-yyyy hh24_mi_ss.ff';
declare
    vd1 date := sysdate;
    vd2 timestamp := localtimestamp;
begin
    dbms_output.put_line( to_char( 'DATE ' || vd1 ));
    dbms_output.put_line( to_char( 'timestamp ' || vd2 ));
end;
```

2021-07-11 21:34:56.123456789

TIMESTAMP

DATE 16--01--2025

timestamp 16-01-2025 10\_23\_29.471000

Prova-ho en SQL Developer





2021-07-11 21:34:56.123456789 CET

- funció `systimestamp` -> torna tipus `timestamp with time zone`

```
set SERVEROUTPUT on
alter session set nls_date_format='dd--mm--yyyy';
alter session set nls_timestamp_format='dd-mm-yyyy hh24_mi_ss.ff';
alter session set nls_timestamp_tz_format='dd/-mm/-yyy hh24::mi::ss.ff';
declare
    vd1 date := sysdate;
    vd2 timestamp with time zone:= systimestamp;
begin
    dbms_output.put_line( to_char( 'DATE ' || vd1 ));
    dbms_output.put_line( to_char( 'timestamp ' || vd2 ));
end;
```

Prova-ho en SQL Developer



## Altres consideracions

SYSDATE and SYSTIMESTAMP return the **system** date and time

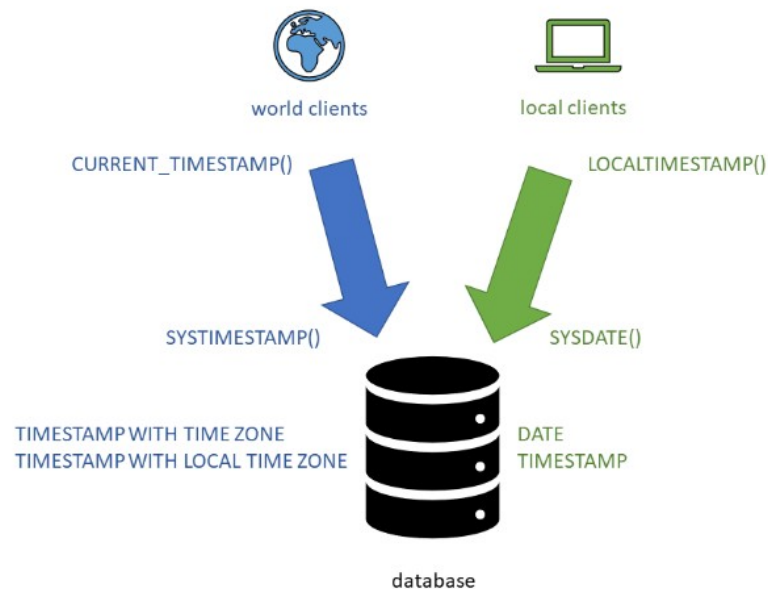
CURRENT\_DATE and CURRENT\_TIMESTAMP and LOCALTIMESTAMP return the current date and time in the **session time zone**.

CURRENT\_DATE returns a **DATE value**

LOCALTIMESTAMP returns a **TIMESTAMP value** while

CURRENT\_TIMESTAMP returns a **TIMESTAMP WITH TIME ZONE value**

## Oracle datetime functions



## Altres consideracions

### Calendari JULIÀ

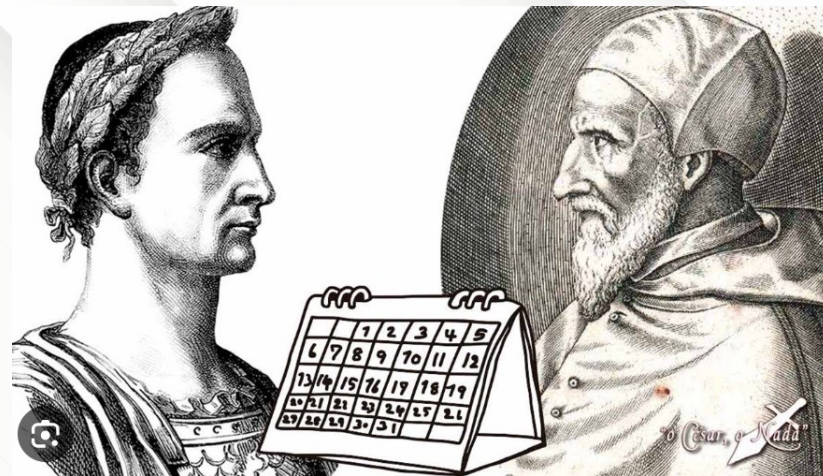
Instaurat per Juli Cèsar l'any 46 a. C., que era una altra versió del calendari egipci, el primer calendari solar conegut

### Calendari GREGORIÀ

Aquest calendari va ser dissenyat per a corregir els errors introduïts pel calendari julià

El julià tenia un xicotet error, pel fet que establia la duració de l'any en 365 dies i 6 hores, quan en realitat era de 365 dies, 5 hores, 48 minuts i 45 segons. D'aquesta manera, cada any la data oficial s'endarreriria 11 minuts i 15 segons respecte a l'astronòmica. La diferència era mínima, però va acumular un desfasament de quasi 10 dies en els més de 1.600 anys que va estar vigent.

Aquest calendari gregorià està dividit en tres tipus d'anys: l'any comú de 365 dies, l'any de traspàs amb 366 dies i l'any secular que és un any que tanca segle.

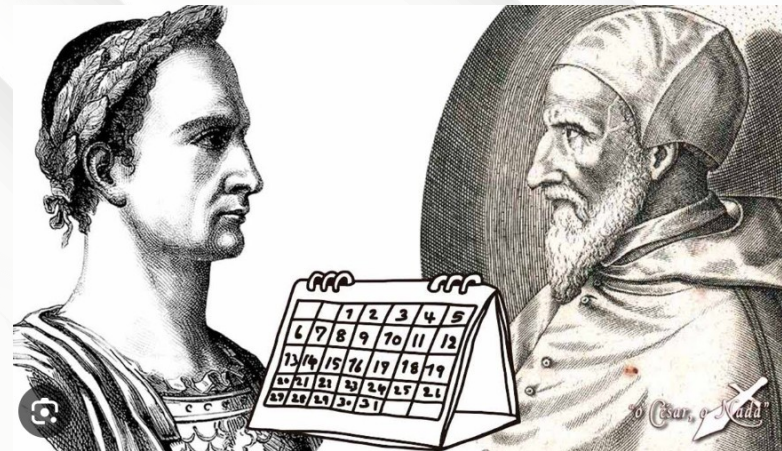


## Altres consideracions

Càlcul Any de traspàs amb 366 dies

a es divisible per b si el resto de la divisió es 0

1. Si l'any és divisible per 4, vaja al pas 2. En cas contrari, vaja al pas 5. (365d)
2. Si l'any és divisible per 100, vaja al pas 3. En cas contrari, vaja al pas 4. (**366d**)
3. Si l'any és divisible per 400, vaja al pas 4. (**366d**) En cas contrari, vaja al pas 5. (365d)
4. L'any és un any de traspàs (té 366 dies).
5. L'any no és un any de traspàs (té 365 dies).





# DATES en ORACLE

---



“

## *Activitat*

*Investiga com funcionen els intervals en ORACLE*

”