Oracle DB – Cheat Sheet on Functions & Format Models
(Not an Exhaustive List)

Case Manipulation
- LOWER(col/expr)
- UPPER(col/expr)
- INITCAP(col/expr)

Character Manipulation
- CONCAT(col,col)
- SUBSTR(col,start_no)
- LENGTH(string)
- INSTR(string,'value')
- LPAD(string,width,'value')
- RPAD(string,width,'value')
- TRIM('value' FROM 'string')
- TRIM(LEADING/TRAILING/BOTH 'value' FROM 'string')
- REPLACE(str1,chars_to_replace,new_chars)

Examples:
- SELECT CONCAT('Hello', 'World')
  FROM DUAL;
- SELECT SUBSTR('HelloWorld', 1, 5)
  FROM DUAL;
- SELECT LENGTH('HelloWorld')
  FROM DUAL;
- SELECT INSTR('HelloWorld', 'W')
  FROM DUAL;
- SELECT last_name, LPAD(salary,10,'*')
  FROM employees;
- SELECT first_name, RPAD(salary,10,'*')
  FROM employees;
- SELECT TRIM('H' FROM 'HelloWorld')
  FROM DUAL;

Number Function
- ROUND(col/expr,dec_places) note: 0 is default; neg_no rounds to left of decimal point
- TRUNC(col/expre,dec_places) note: 0 is default; neg_no rounds to left of decimal point
- MOD(val1,val2)

Date Function

- SYSDATE note: today's date
- MONTHS_BETWEEN(date1,date2)
- ADD_MONTHS(date,no_months_to_add)
- NEXT_DAY(date, 'DOW') note: DOW is day of week ex: 'Friday"
- LAST_DAY(date)
- ROUND(date,'YEAR') note: 7/1 or after rounds to next year
- ROUND(date,'MONTH') NOTE: 16 or after rounds to next month
- TRUNC(date,'YEAR')
- TRUNC(date,'MONTH')

Conversion Functions

- TO_CHAR(number,'format model')
- TO_DATE(string,'format model')
- TO_NUMBER(string,'format model') note: skip the format for now; just want a number

Examples:
SELECT TO_CHAR(SYSDATE, 'fmMonth ddth, YYYY') FROM DUAL;
Returns the format: August 6th, 2004
SELECT TO_CHAR(SYSDATE, 'Month dd, YYYY') FROM DUAL;
Returns the format: August 06, 2004
SELECT TO_CHAR(SYSDATE, 'fmMON dd, YYYY') FROM DUAL;
returns: AUG 6, 2004
SELECT TO_CHAR(SYSDATE, 'fmMonth ddth, Day, YYYY') FROM DUAL;
Returns the format: August 6th, Friday, 2004

$3000.00	TO_CHAR(salary, '$9999.99')
4,500	TO_CHAR(salary, '9,999')
9,000.00	TO_CHAR(salary, '9,999.99')
0004422	TO_CHAR(salary, '0009999')
27-	TO_CHAR(-27,'9,999MI')
(27)	TO_CHAR(-27,'9,999PR')

```
SELECT TO_DATE('June 19 2004', 'fxMonth DD RRRR') AS Convert
FROM DUAL;
SELECT TO_DATE('June19 2004', 'fxMonthDD RRRR') AS Convert
FROM DUAL;
SELECT TO_DATE('July312004', 'fxMonthDDRRRR') AS Convert
FROM DUAL;
```

Format Model – number to characters
To_CHAR(number,'format model')

9	number place holder
0	force 0
$	float $
L	local currency float
.	force decimal
,	force comma
MI	minus sign on right
PR	parens around negative number
EEEE	scientific notation
B	display leading zero as blank

Format Model – dates/time to characters
To_CHAR(date,'format model')

General:

- fm to remove blanks/zeros in output
- fx to force an exact match
- sp to spell out a number
- text that is the same as the format model but is not used for conversion must be in double quotes
- case sensitive – upper case, lower case, title case, mixed case

Elements

YYYY	4-digit year
YEAR	spell out year upper case
MM	2-digit month
MONTH	spell out month upper case
MON	3-char month abbreviation upper case
DY	3-char abbrev for DOW upper case
DAY	spell out DOW upper case
DD	2-digit day
HH	2-digit hour on a 12-hour clock
HH24	2-digit hour on a 24-hour clock
MI	minutes
SS	seconds
AM	mark AM/PM
DDth	add suffix; 1st, 2nd, etc.
DDspth	spell out day with suffix

Format Model – character to date
To_DATE(string,'format model')

- Describe the format of the date using elements specified above
- YYYY vs. RRRR

0-49	RR is 2000
50-99	RR is 1900
YY	always current century