Voy a estructurar la información del archivo en una tabla para facilitarte la consulta y el aprendizaje. Aquí tienes:

| **Categoría** | **Tema** | **Descripción/Detalles** |
| --- | --- | --- |
| **Introducción** | Procedimientos y funciones en PL/SQL | Subprogramas que pueden ser invocados por usuarios. Funciones devuelven un valor; procedimientos no lo hacen. |
| **Funciones predefinidas** | Agrupación | Caracteres, numéricas, fechas, conversión de datos, manejo de nulos, misceláneas. |
| **Funciones caracteres** | LENGTH | Devuelve la longitud de un tipo CHAR. Ejemplo: LENGTH('HOLA MUNDO') → 10 |
|  | INSTR | Busca una subcadena en otra cadena. Ejemplo: INSTR('AQUI ES DONDE', 'BUSCA') → 18 |
|  | REPLACE | Reemplaza texto en una cadena. Ejemplo: REPLACE('HOLA MUNDO','HOLA','VAYA') → VAYA MUNDO |
|  | SUBSTR | Extrae parte de una cadena. Ejemplo: SUBSTR('HOLA MUNDO', 6, 5) → MUNDO |
|  | UPPER, LOWER | Convierte texto a mayúsculas/minúsculas. Ejemplo: UPPER('hola mundo') → HOLA MUNDO |
|  | RTRIM, LTRIM, TRIM | Elimina espacios en blanco. Ejemplo: TRIM(' Hola Mundo ') → Hola Mundo |
| **Funciones numéricas** | MOD | Devuelve el resto de una división. Ejemplo: MOD(20,15) → 5 |
|  | TRUNC | Trunca un número, dejando la parte entera. Ejemplo: TRUNC(9.99) → 9 |
|  | ROUND | Redondea al entero más cercano. Ejemplo: ROUND(9.99) → 10 |
| **Funciones fechas** | SYSDATE | Devuelve la fecha del sistema. |
|  | TRUNC | Trunca una fecha, eliminando horas, minutos y segundos. Ejemplo: TRUNC(SYSDATE) |
| **Conversión de datos** | TO\_DATE | Convierte una cadena a tipo fecha. Ejemplo: TO\_DATE('01/12/2006', 'DD/MM/YYYY') |
|  | TO\_CHAR | Convierte una expresión a texto. Ejemplo: TO\_CHAR(SYSDATE, 'DD/MM/YYYY HH24:MI:SS') |
|  | TO\_NUMBER | Convierte texto a número. Ejemplo: TO\_NUMBER('10.21', '99.99') |
| **Manejo de nulos** | NVL | Devuelve un valor si la expresión es NULL. Ejemplo: NVL(PRECIO, 0) |
| **Misceláneas** | DECODE | Permite lógica if-else. Ejemplo: `DECODE(co\_pais, 'ESP', 'ESPAÑA', 'MEX', 'MÉXICO', 'PAIS' |
| **Procedimientos definidos** | Ventajas | Reutilización de código y división funcional. |
|  | Sintaxis | PROCEDURE nom\_proc [(param1,...)] IS BEGIN sentencias; END; |
|  | Parámetros | Tipos: IN (entrada), OUT (salida), IN OUT (ambos). Ejemplo: PROCEDURE ejemplo(IN nombre VARCHAR2, OUT edad NUMBER) |
|  | Métodos de invocación | Posicional, nominal, mixta. Ejemplo nominal: REGISTRAR\_CLIENTE(P\_NOMBRE => 'Juan', P\_ID => 1) |
| **Funciones definidas** | Declaración | Similar a procedimientos pero devuelven valores. Sintaxis: FUNCTION nom\_funcion(...) RETURN tipo IS BEGIN RETURN(expresión); END; |
| **Sobrecarga y recursión** | Subprogramas | Permite subprogramas con el mismo nombre pero parámetros diferentes. Ejemplo: cálculo factorial. |
| **Administración** | Consultar | SELECT object\_name FROM user\_procedures; |
|  | Eliminar | DROP PROCEDURE nombre; o DROP FUNCTION nombre; |

Si necesitas que detalle aún más algo, dime y lo adaptaré.