

Taller Cuaderno PL/SQL
Iván Ramiro Pinzón
Bases de Datos
Universidad Externado de Colombia
2023

EJERCICIOS SEGUNDO Y TERCER CICLO (31-100)

31. 'SELECT * FROM ClientePLSQL;': Esta consulta muestra todos los registros de la tabla "ClientePLSQL."

ID_CLIENTE	NOMBRE	DIRECCION	TELEFONO
1	625 Ricky Owens	25136 Kelly Oval, Cantrellbury, Florida 72984	3596036549
2	626 Brenda Cervantes	075 Brown Green Suite 198, Port Matthewberg, Utah 95139	483.526.6042
3	627 Gina Morgan	9409 Alvarado Ramp, Grossbury, Kentucky 37569	581-453-1641
4	628 Carl Williams	24381 Jennifer Mountain Suite 361, Danielleshire, South Dakota 16856	(362)765-7798
5	629 Jerry Malone	981 Fletcher Meadow Apt. 108, East Ashleyberg, Rhode Island 72974	817.439.0622
6	631 Jennifer Brown	8353 Holly Prairie, Port Lawrence, Louisiana 95325	546.345.4397
7	634 Summer Wilson	64559 Russell Fort Apt. 762, South Anthonystad, Oklahoma 45519	526.637.2457
8	636 Gary Griffith	94029 Mark Centers, Port Alishaland, New Hampshire 72598	560-893-7332
9	643 Tyler Avila	873 Richard Harbors Suite 274, Port April, Virginia 82642	881-666-9235
10	644 Audrey Ali	15333 Margaret Radial Apt. 908, Duncanside, Wisconsin 11886	415-304-1601
11	648 Michelle Adams	06816 Stephen Falls Suite 058, North Wanda, New Mexico 62934	3783891153
12	651 Jesse Anderson	1862 Dixon Stravenue Apt. 399, Lewisberg, Arizona 69946	(590)440-0974
13	654 Justin Hamilton	3750 Brown Harbor, Washingtonmouth, Washington 93062	+1-287-434-2337
14	656 Samantha Cooper	28123 Justin Wall, East Thomas, Massachusetts 25128	+1-415-436-6560
15	669 Lisa Gray	56386 Richard Mountain, East Andrew, Maryland 86073	6352862563
16	670 Joseph Mitchell	7731 Jennifer Tunnel Suite 505, Rodriguestown, Utah 09703	827.599.4267

32. 'SELECT * FROM AutoPLSQL;': Muestra todos los registros de la tabla "AutoPLSQL."

	ID_AUTO	MARCA	MODELO	ANO
1	2	Honda	Civic	2023
2	3	Nissan	Versa	2023
3	4	Kia	Rio	2023
4	5	Hyundai	Accent	2023
5	6	Chevrolet	Onix	2023
6	7	Fiat	Argo	2023
7	8	Volkswagen	Gol	2023
8	9	Renault	Kwid	2023
9	10	Ford	Ka	2023
10	11	BMW	Serie 3	2023
11	13	Mercedes-Benz	Clase C	2023
12	14	Audi	A4	2023
13	15	Lexus	ES	2023
14	16	Genesis	G80	2023
15	17	Volvo	S60	2023
16	18	Mazda	6	2023

33. 'SELECT * FROM AlquilerPLSQL;': Muestra todos los registros de la tabla "AlquilerPLSQL."

	ID_ALQUILER	ID_CLIENTE	ID_AUTO	FECHA_INICIO	FECHA_FIN	ID_RESERVA
1	583	844	293	27/08/23	20/09/23	583
2	584	131	208	01/10/22	30/10/22	584
3	585	394	543	19/04/23	11/05/23	585
4	586	774	345	19/07/23	30/07/23	586
5	587	81	250	26/03/23	22/04/23	587
6	588	16	371	23/02/23	25/02/23	588
7	589	999	584	01/05/23	21/05/23	589
8	590	796	711	26/05/23	17/06/23	590
9	591	928	812	08/05/23	15/05/23	591
10	592	223	847	27/11/22	14/12/22	592
11	593	987	961	21/08/23	20/09/23	593
12	594	435	45	18/09/23	29/09/23	594
13	595	81	435	06/05/23	31/05/23	595
14	596	397	466	16/09/23	25/09/23	596
15	597	36	473	06/02/23	01/03/23	597
16	598	726	669	09/11/22	23/11/22	598

34. 'SELECT c.nombre, a.marca, a.modelo FROM ClientePLSQL c JOIN AlquilerPLSQL a ON c.id_cliente = a.id_cliente;': Esta consulta presenta un error debido a que intenta recuperar las columnas 'marca' y 'modelo' de la tabla 'AlquilerPLSQL', pero no existen en esa tabla. En lugar de eso, 'marca' y 'modelo' son columnas inexistentes o no relacionadas con la tabla 'AlquilerPLSQL'.

35. 'SELECT a.marca, a.modelo, a.ano FROM AutoPLSQL a JOIN AlquilerPLSQL al ON a.id_auto = al.id_auto;': Muestra la marca, modelo y año de los autos que han sido alquilados en la tabla "AutoPLSQL." Realiza una unión con la tabla "AlquilerPLSQL" basada en el ID del auto.

	MARCA	MODELO	ANO
1	bit	land	1995
2	lot	level	2023
3	economic	approach	2017
4	pressure	become	1990
5	which	picture	2018
6	account	director	2014
7	gas	approach	1993
8	issue	moment	1999
9	cell	test	2018
10	operation	information	2000
11	image	without	2000
12	art	project	2007
13	ball	off	2004
14	later	too	2003
15	another	season	2003
16	strong	century	2019

36. 'SELECT * FROM AlquilerPLSQL WHERE id_cliente = 1;': Muestra todos los registros de la tabla "AlquilerPLSQL" donde el ID del cliente es igual a 1. En este caso no encontramos ningún registro que cumpla la condición.

ID_AL...	ID_CLIENTE	ID_AUTO	FECHA_I...	FECHA_FIN	ID_RESER...
----------	------------	---------	------------	-----------	-------------

37. 'SELECT * FROM AlquilerPLSQL WHERE id_auto = 1;': Muestra todos los registros de la tabla "AlquilerPLSQL" donde el ID del auto es igual a 1. En este caso no encontramos ningún registro que cumpla la condición.

ID_AL...	ID_CLIENTE	ID_AUTO	FECHA_I...	FECHA_FIN	ID_RESER...
----------	------------	---------	------------	-----------	-------------

38. 'SELECT * FROM AlquilerPLSQL WHERE id_sucursal = 1;': Esta consulta presenta un error debido a que intenta recuperar la columna 'id_sucursal' de la tabla 'AlquilerPLSQL', pero dicha columna no existe en esa tabla. En lugar de eso, 'id_sucursal' es una columna inexistente o no relacionada con la tabla 'AlquilerPLSQL'.

39. 'SELECT * FROM AlquilerPLSQL WHERE fecha_inicio = '2023-09-27';': Esta consulta presenta un error debido a que intenta recuperar los registros de la tabla ALQUILERPLSQL donde la fecha de inicio es '2023-09-27', pero en la tabla el formato de fecha es 'DD/MM/AA'

40. 'SELECT COUNT(*) FROM AlquilerPLSQL;': Cuenta el número total de registros en la tabla "AlquilerPLSQL."

	COUNT(*)
1	999

41. SELECT c.nombre FROM ClientePLSQL c JOIN AlquilerPLSQL a ON c.id_cliente = a.id_cliente JOIN SucursalPLSQL s ON a.id_sucursal = s.id_sucursal WHERE s.nombre = 'Sucursal Central';: Esta consulta presenta un error debido a que intenta recuperar la columna 'id_sucursal' de la tabla 'AlquilerPLSQL', pero dicha columna no existe en esa tabla.

42. SELECT a.marca, a.modelo FROM AutoPLSQL a JOIN AlquilerPLSQL al ON a.id_auto = al.id_auto WHERE al.id_cliente = 1 AND al.fecha_inicio = '2023-09-27';: Esta consulta no arroja ningún resultado debido a que no existe un cliente con un ID igual a 1. Además, es necesario corregir el formato de fecha para que sea adecuado.

MARCA	MODELO
-------	--------

43. SELECT * FROM AlquilerPLSQL WHERE fecha_fin - fecha_inicio > 7;: Muestra todos los registros de la tabla "AlquilerPLSQL" donde la diferencia entre la fecha de inicio y la fecha de finalización es mayor que 7 días.

ID_ALQUILER	ID_CLIENTE	ID_AUTO	FECHA_INICIO	FECHA_FIN	ID_RESERVA
1	583	844	293 27/08/23	20/09/23	583
2	584	131	208 01/10/22	30/10/22	584
3	585	394	543 19/04/23	11/05/23	585
4	586	774	345 19/07/23	30/07/23	586
5	587	81	250 26/03/23	22/04/23	587
6	589	999	584 01/05/23	21/05/23	589
7	590	796	711 26/05/23	17/06/23	590
8	592	223	847 27/11/22	14/12/22	592
9	593	987	961 21/08/23	20/09/23	593
10	594	435	45 18/09/23	29/09/23	594
11	595	81	435 06/05/23	31/05/23	595
12	596	397	466 16/09/23	25/09/23	596
13	597	36	473 06/02/23	01/03/23	597
14	598	726	669 09/11/22	23/11/22	598
15	599	999	715 07/10/22	25/10/22	599
16	600	414	809 29/12/22	10/01/23	600
17	581	433	433 03/10/22	15/10/22	581

44. SELECT c.nombre, COUNT(*) AS numero_alquileres FROM ClientePLSQL c JOIN AlquilerPLSQL a ON c.id_cliente = a.id_cliente GROUP BY c.nombre ORDER BY numero_alquileres DESC LIMIT 1;: Esta sentencia presenta un error porque la palabra clave "LIMIT" no es válida en Oracle SQL.

45. SELECT a.marca, a.modelo, COUNT(*) AS numero_alquileres FROM AutoPLSQL a JOIN AlquilerPLSQL al ON a.id_auto = al.id_auto GROUP BY a.marca, a.modelo ORDER BY numero_alquileres DESC LIMIT 1;; Esta sentencia presenta un error porque la palabra clave "LIMIT" no es válida en Oracle SQL.

46. SELECT s.nombre, COUNT(*) AS numero_alquileres FROM SucursalPLSQL s JOIN AlquilerPLSQL al ON s.id_sucursal = al.id_sucursal GROUP BY s.nombre ORDER BY numero_alquileres DESC LIMIT 1;; Esta sentencia presenta un error porque la palabra clave "LIMIT" no es válida en Oracle SQL.

47. SELECT EXTRACT(MONTH FROM fecha_inicio) AS mes, COUNT(*) AS numero_alquileres FROM AlquilerPLSQL GROUP BY EXTRACT(MONTH FROM fecha_inicio) ORDER BY numero_alquileres DESC LIMIT 1;; Esta sentencia presenta un error porque la palabra clave "LIMIT" no es válida en Oracle SQL.

48. SELECT EXTRACT(DAYOFWEEK FROM fecha_inicio) AS dia_semana, COUNT(*) AS numero_alquileres FROM AlquilerPLSQL GROUP BY EXTRACT(DAYOFWEEK FROM fecha_inicio) ORDER BY numero_alquileres DESC LIMIT 1;; Esta sentencia presenta un error porque la palabra clave "LIMIT" no es válida en Oracle SQL. Además presenta una sintaxis errónea.

49. SELECT * FROM AlquilerPLSQL ORDER BY precio DESC LIMIT 1;; Esta sentencia presenta un error porque la palabra clave "LIMIT" no es válida en Oracle SQL.

50. SELECT * FROM AlquilerPLSQL ORDER BY precio ASC LIMIT 1;; Esta sentencia presenta un error porque la palabra clave "LIMIT" no es válida en Oracle SQL.

51. SELECT * FROM ClientePLSQL WHERE nombre LIKE '%Juan%'; Esta consulta recupera de la tabla ClientePLSQL las filas en las que en la columna 'NOMBRE' se encuentra 'Juan'.

ID_CLIENTE	NOMBRE	DIRECCION	TELEFONO
1	675 Juan Wilson	13908 Chase Estates, South Morganmouth, Wisconsin 86282	+1-200-759-2343
2	483 Juan Baker	15104 Eric Mount, Millermouth, Arizona 23327	9963132346
3	816 Juan Ortiz	39333 McKinney Centers Apt. 637, West Krystalport, Alabama 08808	(783)840-2628

52. SELECT a.marca, a.modelo, a.ano FROM AutoPLSQL a WHERE precio < 10000; La consulta contiene un error ya que intenta recuperar la columna 'precio' de la tabla 'AutoPLSQL' cuando su valor sea mayor a 10000, pero dicha columna no se encuentra en esa tabla.

53. SELECT * FROM AlquilerPLSQL WHERE fecha_inicio BETWEEN '2023-09-01' AND '2023-09-30'; Esta consulta presenta un error debido a que intenta recuperar los registros de la tabla ALQUILERPLSQL donde la fecha de inicio esté entre '2023-09-01' y '2023-09-30', pero en la tabla el formato de fecha es 'DD/MM/AA'

54. SELECT c.nombre, a.marca, a.modelo FROM ClientePLSQL c JOIN AlquilerPLSQL a ON c.id_cliente = a.id_cliente WHERE c.direccion LIKE '%Bogotá%'; Esta consulta presenta un error debido a que intenta recuperar las columnas 'marca' y

'modelo' de la tabla 'AlquilerPLSQL', pero no existen en esa tabla. En lugar de eso, 'marca' y 'modelo' son columnas inexistentes o no relacionadas con la tabla 'AlquilerPLSQL'.

55. SELECT a.marca, a.modelo, a.ano FROM AutoPLSQL a JOIN AlquilerPLSQL al ON a.id_auto = al.id_auto WHERE al.id_reserva = 1; Muestra la marca, modelo y año de autos que han sido alquilados en función del ID de reserva igual a 1. Realiza una unión entre las tablas "AutoPLSQL" y "AlquilerPLSQL."

	MARCA	MODELO	..
1	later	too	2003

56. SELECT * FROM AlquilerPLSQL WHERE id_cliente IN (1, 2, 3); Muestra los registros de alquiler en los que el ID del cliente está en el conjunto de valores (1, 2, 3).

	ID_ALQUILER	ID_CLIENTE	ID_AUTO	FECHA_INICIO	FECHA_FIN	ID_RESERVA
1	861	3	171	20/12/22	10/01/23	861
2	292	3	891	19/08/23	13/09/23	292
3	384	2	141	19/12/22	03/01/23	384

57. SELECT * FROM AlquilerPLSQL WHERE id_auto IN (1, 2, 3); Muestra los registros de alquiler en los que el ID del auto está en el conjunto de valores (1, 2, 3).

	ID_ALQ...	ID_CLIENTE	ID_AUTO	FECHA_INICIO	FECHA_FIN	ID_RESERVA
1	723	911	2	19/04/23	01/05/23	723
2	148	922	3	13/11/22	25/11/22	148
3	329	668	3	03/07/23	07/07/23	329

58. SELECT * FROM AlquilerPLSQL WHERE id_sucursal IN (1, 2, 3); Esta consulta presenta un error debido a que intenta recuperar la columna 'id_sucursal' de la tabla 'AlquilerPLSQL', pero dicha columna no existe en esa tabla. En lugar de eso, 'id_sucursal' es una columna inexistente o no relacionada con la tabla 'AlquilerPLSQL'.

59. SELECT * FROM AlquilerPLSQL WHERE fecha_inicio BETWEEN '2023-09-01' AND '2023-09-30' AND id_cliente IN (1, 2, 3); Esta consulta presenta un error debido a que intenta recuperar los registros de la tabla ALQUILERPLSQL donde la fecha de inicio esté entre '2023-09-01' y '2023-09-30', pero en la tabla el formato de fecha es 'DD/MM/AA'.

60. SELECT * FROM AlquilerPLSQL WHERE fecha_inicio BETWEEN '2023-09-01' AND '2023-09-30' AND id_auto IN (1, 2, 3); Esta consulta presenta un error debido a que intenta recuperar los registros de la tabla ALQUILERPLSQL donde la fecha de inicio esté entre '2023-09-01' y '2023-09-30', pero en la tabla el formato de fecha es 'DD/MM/AA'.

61. SELECT * FROM AlquilerPLSQL WHERE fecha_inicio BETWEEN '2023-09-01' AND '2023-09-30' AND id_sucursal IN (1, 2, 3); Esta consulta presenta un error debido a que intenta recuperar los registros de la tabla ALQUILERPLSQL donde la

fecha de inicio esté entre '2023-09-01' y '2023-09-30', pero en la tabla el formato de fecha es 'DD/MM/AA'. Además de esto, la columna 'id_sucursal' es inexistente en esta tabla.

62. SELECT c.nombre, COUNT(*) AS numero_alquileres FROM ClientePLSQL c JOIN AlquilerPLSQL a ON c.id_cliente = a.id_cliente GROUP BY c.nombre ORDER BY numero_alquileres DESC LIMIT 1; Esta sentencia presenta un error porque la palabra clave "LIMIT" no es válida en Oracle SQL.

63. SELECT a.marca, a.modelo, COUNT(*) AS numero_alquileres FROM AutoPLSQL a JOIN AlquilerPLSQL al ON a.id_auto = al.id_auto GROUP BY a.marca, a.modelo ORDER BY numero_alquileres DESC LIMIT 1; Esta sentencia presenta un error porque la palabra clave "LIMIT" no es válida en Oracle SQL.

64. `SELECT s.nombre, COUNT(*) AS numero_alquileres FROM SucursalPLSQL s JOIN AlquilerPLSQL al ON s.id_sucursal = al.id_sucursal GROUP BY s.nombre ORDER BY numero_alquileres DESC LIMIT 1;` Esta sentencia presenta un error porque la palabra clave "LIMIT" no es válida en Oracle SQL.

65. `SELECT EXTRACT(MONTH FROM fecha_inicio) AS mes, COUNT(*) AS numero_alquileres FROM AlquilerPLSQL GROUP BY EXTRACT(MONTH FROM fecha_inicio) ORDER BY numero_alquileres DESC LIMIT 1;` Esta sentencia presenta un error porque la palabra clave "LIMIT" no es válida en Oracle SQL.

66. `SELECT EXTRACT(DAYOFWEEK FROM fecha_inicio) AS dia_semana, COUNT(*) AS numero_alquileres FROM AlquilerPLSQL GROUP BY EXTRACT(DAYOFWEEK FROM fecha_inicio) ORDER BY numero_alquileres DESC LIMIT 1;` Esta sentencia presenta un error porque la palabra clave "LIMIT" no es válida en Oracle SQL. Además de esto, tiene errores en sintaxis.

67. `SELECT * FROM AlquilerPLSQL ORDER BY precio DESC LIMIT 1;` Esta sentencia presenta un error porque la palabra clave "LIMIT" no es válida en Oracle SQL. Además de esto, la columna 'precio' no existe en AlquilerPLSQL

68. `SELECT * FROM AlquilerPLSQL ORDER BY precio ASC LIMIT 1;` Esta sentencia presenta un error porque la palabra clave "LIMIT" no es válida en Oracle SQL. Además de esto, la columna 'precio' no existe en AlquilerPLSQL

69. `SELECT * FROM ClientePLSQL WHERE nombre LIKE '%Juan%' AND fecha_inicio BETWEEN '2023-09-01' AND '2023-09-30';` Esta consulta presenta un error debido a que intenta recuperar la columna 'fecha_inicio' de la tabla 'ClientePLSQL', pero no existe en esa tabla. En lugar de eso, 'fecha_inicio' es una columna inexistente o no relacionada con la tabla 'ClientePLSQL'.

70. `SELECT a.marca, a.modelo, a.ano FROM AutoPLSQL a WHERE precio < 10000 AND fecha_inicio BETWEEN '2023-09-01' AND '2023-09-30';` Esta consulta presenta un error debido a que intenta recuperar las columnas 'precio' y 'fecha_inicio' de la tabla 'ClientePLSQL', pero no existen en esa tabla. En lugar de eso, 'precio' y 'fecha_inicio' son columnas inexistentes o no relacionadas con la tabla 'ClientePLSQL'.

81. **`CREATE VIEW vista_clientes_alquilados_sucursal AS SELECT c.nombre, a.marca, a.modelo FROM ClientePLSQL c JOIN AlquilerPLSQL a ON c.id_cliente = a.id_cliente JOIN SucursalPLSQL s ON a.id_sucursal = s.id_sucursal WHERE s.nombre = 'Sucursal Central';`**: Esta consulta presenta un error debido a que intenta recuperar la columna 'id_sucursal' de la tabla 'AlquilerPLSQL', pero dicha columna no existe en esa tabla.

82. **`CREATE VIEW vista_autos_alquilados_cliente_fecha AS SELECT a.marca, a.modelo FROM AutoPLSQL a JOIN AlquilerPLSQL al ON a.id_auto = al.id_auto WHERE al.id_cliente = 1 AND al.fecha_inicio = '2023-09-27';`**: Esta vista muestra la marca y modelo de autos alquilados por un cliente específico (ID de cliente igual a 1) en una fecha específica (fecha de inicio igual a '2023-09-27'). En este caso el resultado está completamente vacío, ya que, no existe un cliente con ID igual a 1

⚡ MARCA	⚡ MODELO
---------	----------

83. **`CREATE VIEW vista_alquileres_mas_7dias AS SELECT * FROM AlquilerPLSQL WHERE fecha_fin - fecha_inicio > 7;`**: Esta vista muestra todos los registros de alquiler en los que la diferencia entre la fecha de inicio y la fecha de finalización es mayor que 7 días.

	ID_ALQUILER	ID_CLIENTE	ID_AUTO	FECHA_INICIO	FECHA_FIN	ID_RESERVA
1	583	844	293	27/08/23	20/09/23	583
2	584	131	208	01/10/22	30/10/22	584
3	585	394	543	19/04/23	11/05/23	585
4	586	774	345	19/07/23	30/07/23	586
5	587	81	250	26/03/23	22/04/23	587
6	589	999	584	01/05/23	21/05/23	589
7	590	796	711	26/05/23	17/06/23	590
8	592	223	847	27/11/22	14/12/22	592
9	593	987	961	21/08/23	20/09/23	593
10	594	435	45	18/09/23	29/09/23	594
11	595	81	435	06/05/23	31/05/23	595
12	596	397	466	16/09/23	25/09/23	596
13	597	36	473	06/02/23	01/03/23	597
14	598	726	669	09/11/22	23/11/22	598
15	599	999	715	07/10/22	25/10/22	599
16	600	414	809	29/12/22	10/01/23	600

84. ``CREATE VIEW vista_clientes_mas_alquileres AS SELECT c.nombre, COUNT(*) AS numero_alquileres FROM ClientePLSQL c JOIN AlquilerPLSQL a ON c.id_cliente = a.id_cliente GROUP BY c.nombre ORDER BY numero_alquileres DESC;``:

	NOMBRE	NUMERO_ALQUILERES
1	John Greene	6
2	Pamela Johnson	5
3	Michele Allen	4
4	Haley Carroll	4
5	Jill Sanford	4
6	Andrew Chang	4
7	David Woods	4
8	Stephanie Young	4
9	Melinda Jackson	3
10	Kyle Stephens	3
11	Zachary George	3
12	Steven Velasquez	3
13	Ashley Farmer	3
14	Danielle Williamson	3
15	Lauren Jones	3
16	Travis Myers	3
17	Kevin Smith	2

85. `CREATE VIEW vista_autos_mas_alquileres AS SELECT a.marca, a.modelo, COUNT(*) AS numero_alquileres FROM AutoPLSQL a JOIN AlquilerPLSQL al ON a.id_auto = al.id_auto GROUP BY a.marca, a.modelo ORDER BY numero_alquileres DESC;`

	⚡ MARCA	⚡ MODELO	⚡ NUMERO_ALQUILERES
1	end	hold	5
2	significant	pretty	5
3	lot	level	5
4	tough	year	4
5	analysis	reveal	4
6	economic	approach	4
7	just	morning	4
8	worry	behavior	4
9	scientist	media	4
10	same	fund	4
11	culture	far	4
12	special	back	4
13	similar	few	4
14	heavy	improve	4
15	later	too	3
16	assume	husband	3

86. **``CREATE VIEW vista_sucursales_mas_alquileres AS SELECT s.nombre, COUNT(*) AS numero_alquileres FROM SucursalPLSQL s JOIN AlquilerPLSQL al ON s.id_sucursal = al.id_sucursal GROUP BY s.nombre ORDER BY numero_alquileres DESC;``**: Esta consulta presenta un error debido a que intenta recuperar la columna 'id_sucursal' de la tabla 'AlquilerPLSQL', pero dicha columna no existe en esa tabla.

87. **``CREATE VIEW vista_meses_mas_alquileres AS SELECT EXTRACT(MONTH FROM fecha_inicio) AS mes, COUNT(*) AS numero_alquileres FROM AlquilerPLSQL GROUP BY EXTRACT(MONTH FROM fecha_inicio) ORDER BY numero_alquileres DESC;``**: Esta vista muestra el mes con la mayor cantidad de alquileres y la cantidad de alquileres realizados en ese mes, ordenados de mayor a menor.

	MES	NUMERO_ALQUILERES
1	12	100
2	5	98
3	10	91
4	11	88
5	9	87
6	1	84
7	4	84
8	3	81
9	8	79
10	6	76
11	7	70
12	2	61

88. ``CREATE VIEW vista_dias_semana_mas_alquileres AS SELECT EXTRACT(DAYOFWEEK FROM fecha_inicio) AS dia_semana, COUNT(*) AS numero_alquileres FROM AlquilerPLSQL GROUP BY EXTRACT(DAYOFWEEK FROM fecha_inicio) ORDER BY numero_alquileres DESC;``: Esta sentencia presenta un error de sintaxis, el cual parece estar relacionado con la función EXTRACT, la cual requiere que se especifique un campo válido como el segundo argumento, y "DAYOFWEEK" no es una opción válida.

89. ``CREATE VIEW vista_alquileres_mas_caros AS SELECT * FROM AlquilerPLSQL ORDER BY precio DESC;``: Esta consulta presenta un error debido a que intenta recuperar la columna 'precio' de la tabla 'AlquilerPLSQL', pero dicha columna no existe en esa tabla.

90. ``CREATE VIEW vista_alquileres_mas_baratos AS SELECT * FROM AlquilerPLSQL ORDER BY precio ASC;``: Esta consulta presenta un error debido a que intenta recuperar la columna 'precio' de la tabla 'AlquilerPLSQL', pero dicha columna no existe en esa tabla.

EJERCICIOS TERCER CICLO (91-100):

91.

```
CREATE TRIGGER trg_insert_auto_1000258145
BEFORE INSERT ON AutoPLSQL
FOR EACH ROW
BEGIN
-- Actualizar el número de autos disponibles
UPDATE AutoPLSQL
SET numero_disponibles = numero_disponibles + 1
WHERE id_auto = NEW.id_auto;
END;
```

Este trigger se utiliza para mantener actualizado el número de autos disponibles en la tabla "AutoPLSQL" cada vez que se realiza una inserción de un nuevo registro en esa tabla.

92.

```
CREATE TRIGGER trg_delete_auto
BEFORE DELETE ON AutoPLSQL
FOR EACH ROW
BEGIN
-- Actualizar el número de autos disponibles
UPDATE AutoPLSQL
SET numero_disponibles = numero_disponibles - 1
WHERE id_auto = OLD.id_auto;
END;
```

Este trigger se utiliza para mantener actualizado el número de autos disponibles en la tabla "AutoPLSQL" cada vez que se realiza una eliminación de un registro en esa tabla.

93.

```
CREATE TRIGGER trg_update_auto
BEFORE UPDATE ON AutoPLSQL
FOR EACH ROW
BEGIN
-- Actualizar el número de autos disponibles
IF NEW.numero_disponibles != OLD.numero_disponibles THEN
UPDATE AutoPLSQL
SET numero_disponibles = NEW.numero_disponibles
WHERE id_auto = NEW.id_auto;
END IF;
END;
```

Este trigger actualiza el número de autos disponibles en la tabla AutoPLSQL cada vez que se actualiza un registro en la tabla.

94.

```
CREATE TRIGGER trg_insert_cliente  
BEFORE INSERT ON ClientePLSQL  
FOR EACH ROW  
BEGIN  
  -- Actualizar el número de clientes  
  UPDATE ClientePLSQL  
    SET numero_clientes = numero_clientes + 1;  
END;
```

Este trigger actualiza el número de clientes en la tabla ClientePLSQL cada vez que se inserta un registro en la tabla.

95.

```
CREATE TRIGGER trg_delete_cliente  
BEFORE DELETE ON ClientePLSQL  
FOR EACH ROW  
BEGIN  
  -- Actualizar el número de clientes  
  UPDATE ClientePLSQL  
    SET numero_clientes = numero_clientes - 1;  
END;
```

Este trigger se utiliza para mantener actualizado el número de clientes en la tabla "AutoPLSQL" cada vez que se realiza la eliminación de un registro en esa tabla.

96.

```
CREATE TRIGGER trg_update_cliente  
BEFORE UPDATE ON ClientePLSQL  
FOR EACH ROW  
BEGIN  
  -- Actualizar el número de clientes  
  IF NEW.numero_alquileres != OLD.numero_alquileres THEN  
    UPDATE ClientePLSQL  
      SET numero_alquileres = NEW.numero_alquileres  
      WHERE id_cliente = NEW.id_cliente;  
  END IF;  
END;
```

Este trigger actualiza el número de alquileres de un cliente cada vez que se actualiza el registro del cliente en la tabla.

97.

```
CREATE PROCEDURE proc_calcular_precio_alquiler
(
  IN id_alquiler INT,
  IN id_auto INT,
  IN fecha_inicio DATE,
  IN fecha_fin DATE
)
AS
BEGIN
  -- Calcular el precio del alquiler
  DECLARE
    precio_base NUMERIC(10, 2);
    dias_alquiler INT;
  BEGIN
    precio_base := (SELECT precio FROM AutoPLSQL WHERE id_auto = id_auto);
    dias_alquiler := (fecha_fin - fecha_inicio) + 1;
    SET NEW.precio = precio_base * dias_alquiler;
  END;
END;
```

Es un procedimiento que calcula el precio de un alquiler de un auto.

98.

```
CREATE PROCEDURE proc_listar_alquileres_cliente
(
  IN id_cliente INT
)
AS
BEGIN
  -- Listar los alquileres del cliente
  SELECT *
  FROM AlquilerPLSQL
  WHERE id_cliente = id_cliente;
END;
```

Este procedimiento lista los alquileres de un cliente.

99.

```
CREATE PROCEDURE proc_listar_autos_sucursal
(
  IN id_sucursal INT
)
AS
```



```

BEGIN
  -- Listar los autos de la sucursal
  SELECT *
  FROM AutoPLSQL
  WHERE id_sucursal = id_sucursal;
END;

```

Este procedimiento lista los autos alquilados de una sucursal.

100.

```

CREATE PROCEDURE proc_agregar_auto
(
  IN marca VARCHAR(255),
  IN modelo VARCHAR(255),
  IN ano INT,
  IN numero_disponibles INT
)
AS
BEGIN
  -- Insertar un nuevo auto
  INSERT INTO AutoPLSQL (marca, modelo, ano, numero_disponibles)
  VALUES (marca, modelo, ano, numero_disponibles);
END;

```

Este procedimiento agrega un nuevo auto a la tabla AutoPLSQL

101.

```

CREATE PROCEDURE proc_eliminar_auto
(
  IN id_auto INT
)
AS
BEGIN
  -- Eliminar un auto
  DELETE FROM AutoPLSQL
  WHERE id_auto = id_auto;
END;

```

Este procedimiento elimina un auto de la tabla AutoPLSQL