

1.- Obtener la información de los pedidos realizados por el Empleado con DNI 72850874F.

```
SELECT*
FROM Pedidos p
LEFT JOIN Empleados e ON p.cod_pedido=e.cod_empleado
WHERE p.cod_empleado = 3;
```

	cod_pedido	fecha_pedido	fecha_envio	fecha_entrega	estado	descripcion	cod_empleado
▶	P_00002	2020-01-01	2020-01-02	NULL	ENVIADO	NULL	3
	P_00008	2020-01-05	2020-01-06	2020-01-16	DEVUELTO	NULL	3
	P_00013	2020-01-08	NULL	NULL	ESPERA	NULL	3

2.- Obtener el nombre, DNI y teléfono de los empleados con pedidos que aún no han sido enviados.

```
SELECT e.nombre, e.DNI, e.tlfno
FROM Empleados e
LEFT JOIN Pedidos p ON e.cod_empleado=p.cod_pedido
WHERE e.cod_empleado = 2;
```

	nombre	DNI	tlfno
▶	William Wyler	43207030N	555303030

```
SELECT e.nombre, e.DNI, e.tlfno
FROM Empleados e
LEFT JOIN Pedidos p ON e.cod_empleado=p.cod_pedido
WHERE e.cod_empleado = 3;
```

	nombre	DNI	tlfno
▶	David Lean	72850874F	555505050

3.- Obtener la información de los empleados que no han realizado ningún pedido.

```
SELECT*
FROM Empleados e
LEFT JOIN Pedidos p ON e.cod_empleado = p.cod_pedido
WHERE e.departamento <> 'logistic';
```

cod_empleado	DNI	Nombre	email	tlfno	extension	oficina	departamento	informa_a
1	59840570K	George Roy Hill	georgeroyhill@Metrowoldwynmayer.com	555202020	EXT_01	OF_01	marketing	NULL
2	43207030N	William Wyler	williamwyler@warnerbro.com	555303030	EXT_02	OF_02	marketing	1
4	51190963R	Cecil B. DeMille	cecilbdemille@Metrowoldwynmayer.com	555404040	EXT_03	NULL	marketing	3
6	16170520W	Brian de Palma	briandepalma@warnerbro.com	555606060	EXT_02	OF_02	marketing	5
7	37615276H	Roman Polansky	romanpolanskyk@Metrowoldwynmayer.com	555808080	EXT_01	OF_01	marketing	2

4.- Obtener la información de los pedidos que no tienen ningún empleado asociado.

```
SELECT*
FROM Pedidos p
RIGHT JOIN Empleados e ON p.cod_pedido=e.cod_empleado
WHERE e.cod_empleado IS NULL;
```

cod_pedido	fecha_pedido	fecha_envio	fecha_entrega	estado	descripcion	cod_empleado	cod_empleado	DNI	Nombre	email	tlfno
P_00001	2020-01-01	2020-01-02	2020-01-09	TERMINADO	NULL	2	NULL	NULL	NULL	NULL	NULL
P_00002	2020-01-01	2020-01-02	NULL	ENVIADO	NULL	3	NULL	NULL	NULL	NULL	NULL
P_00003	2020-01-02	2020-01-03	NULL	ENVIADO	NULL	4	NULL	NULL	NULL	NULL	NULL
P_00004	2020-01-02	2020-01-03	2020-01-10	DEVUELTO	NULL	5	NULL	NULL	NULL	NULL	NULL
P_00005	2020-01-04	2020-01-05	2020-01-15	TERMINADO	NULL	6	NULL	NULL	NULL	NULL	NULL
P_00006	2020-01-04	2020-01-05	2020-01-15	TERMINADO	NULL	7	NULL	NULL	NULL	NULL	NULL
P_00007	2020-01-04	2020-01-05	2020-01-15	TERMINADO	NULL	2	NULL	NULL	NULL	NULL	NULL
P_00008	2020-01-05	2020-01-06	2020-01-16	DEVUELTO	NULL	3	NULL	NULL	NULL	NULL	NULL
P_00009	2020-01-06	2020-01-07	NULL	ENVIADO	NULL	4	NULL	NULL	NULL	NULL	NULL

5.- Obtener la información del pago relacionado en el pedido que tiene como identificador P_00002.

```
SELECT*
FROM Pagos pa
LEFT JOIN Pedidos p ON pa.cod_cliente=p.cod_pedido
WHERE pa.cod_pedido = 'P_00002';
```

cod_pago	cod_cliente	cantidad	num_plazos	num_pago	cuantia_plazo	cod_pedido	pago_master	cod_pedido	fecha_pedido	fecha_envio	ft
2	1	1000	2	1	500	P_00002	NULL	NULL	NULL	NULL	NU

6.- Obtener la información de los pedidos sin pagos asociados.

```
SELECT p.*
FROM Pagos pa
Right JOIN Pedidos p ON pa.pago_master=p.cod_pedido
WHERE pa.cod_pedido IS NULL;
```

cod_pedido	fecha_pedido	fecha_envio	fecha_entrega	estado	descripcion	cod_empleado
P_00001	2020-01-01	2020-01-02	2020-01-09	TERMINADO	NULL	2
P_00002	2020-01-01	2020-01-02	NULL	ENVIADO	NULL	3
P_00003	2020-01-02	2020-01-03	NULL	ENVIADO	NULL	4
P_00004	2020-01-02	2020-01-03	2020-01-10	DEVUELTO	NULL	5
P_00005	2020-01-04	2020-01-05	2020-01-15	TERMINADO	NULL	6
P_00006	2020-01-04	2020-01-05	2020-01-15	TERMINADO	NULL	7
P_00007	2020-01-04	2020-01-05	2020-01-15	TERMINADO	NULL	2
P_00008	2020-01-05	2020-01-06	2020-01-16	DEVUELTO	NULL	3
P_00009	2020-01-06	2020-01-07	NULL	ENVIADO	NULL	4
P_00010	2020-01-06	2020-01-07	NULL	ENVIADO	NULL	5
P_00011	2020-01-07	2020-01-08	NULL	ENVIADO	NULL	6
P_00012	2020-01-07	2020-01-08	NULL	ENVIADO	NULL	7
P_00013	2020-01-08	NULL	NULL	ESPERA	NULL	3
P_00014	2020-01-08	NULL	NULL	ESPERA	NULL	2
P_00015	2020-01-08	NULL	NULL	ESPERA	NULL	2
P_00016	2020-01-09	NULL	NULL	ESPERA	NULL	2

7.- Obtener el nombre de los empleados con pedidos con pagos asociados. El resultset obtenido debe eliminar repeticiones, además los nombres se deben mostrar de ordenados alfabéticamente de manera descendente.

```
SELECT e.Nombre
FROM Empleados e
LEFT JOIN Pagos pa ON e.cod_empleado=pa.pago_master
WHERE pa.pago_master IS NOT NULL
ORDER BY Nombre ASC;
```

	Nombre
▶	David Lean
	David Lean
	William Wyler
	William Wyler

8.- Obtener el nombre de los empleados con pedidos que no tienen pagos asociados. El resultset obtenido debe eliminar repeticiones, además los nombres se deben mostrar de ordenados alfabéticamente de manera ascendente.

```
SELECT e.Nombre
FROM Empleados e
LEFT JOIN Pagos pa ON e.cod_empleado=pa.pago_master
WHERE pa.pago_master IS NULL
ORDER BY Nombre ASC;
```

	Nombre
▶	Brian de Palma
	Cecil B. DeMille
	George Roy Hill
	Roman Polansky
	Sydney Pollack

```
SELECT *
FROM Pagos pa1
INNER JOIN Pagos pa2 ON pa1.cod_cliente=pa2.cod_cliente
WHERE pa1.cod_cliente = 1;
```

cod_pac	cod_cli	canti	num_pla	num_pag	cuantia_p	cod_pedido	pago_mast	cod_pago	cod_cliente	canti	num_pla	num_	cuantia_	cod_pedido	pago_ma
1	1	100	0	1	0	P_00001	NULL	1	1	100	0	1	0	P_00001	NULL
1	1	100	0	1	0	P_00001	NULL	2	1	1000	2	1	500	P_00002	NULL
1	1	100	0	1	0	P_00001	NULL	5	1	500	2	1	0	NULL	2
1	1	100	0	1	0	P_00001	NULL	6	1	500	2	2	0	NULL	2
2	1	1000	2	1	500	P_00002	NULL	1	1	100	0	1	0	P_00001	NULL
2	1	1000	2	1	500	P_00002	NULL	2	1	1000	2	1	500	P_00002	NULL
2	1	1000	2	1	500	P_00002	NULL	5	1	500	2	1	0	NULL	2
2	1	1000	2	1	500	P_00002	NULL	6	1	500	2	2	0	NULL	2
5	1	500	2	1	0	NULL	2	1	1	100	0	1	0	P_00001	NULL
5	1	500	2	1	0	NULL	2	2	1	1000	2	1	500	P_00002	NULL
5	1	500	2	1	0	NULL	2	5	1	500	2	1	0	NULL	2
5	1	500	2	1	0	NULL	2	6	1	500	2	2	0	NULL	2
6	1	500	2	2	0	NULL	2	1	1	100	0	1	0	P_00001	NULL
6	1	500	2	2	0	NULL	2	2	1	1000	2	1	500	P_00002	NULL
6	1	500	2	2	0	NULL	2	5	1	500	2	1	0	NULL	2
6	1	500	2	2	0	NULL	2	6	1	500	2	2	0	NULL	2

```
SELECT c.email
FROM Clientes c
INNER JOIN Pagos pa ON c.cod_cliente=pa.cod_cliente
WHERE pa.cod_cliente = 3;
```

[illegible]

```

SELECT c.email
FROM Clientes c
INNER JOIN Pagos pa ON c.cod_cliente=pa.cod_cliente
WHERE pa.cod_cliente = 4;

```

email
gracekelly@warnerbro.com
gracekelly@warnerbro.com
gracekelly@warnerbro.com
gracekelly@warnerbro.com
gracekelly@warnerbro.com

11.- Obtener los pagos aplazados del pago maestro con identificador 14.

```

SELECT pa1.*
FROM Pagos pa1
INNER JOIN Pagos pa2 ON pa1.cod_pago=pa2.pago_master
WHERE pa2.pago_master = 14;

```

cod_pago	cod_cliente	cantidad	num_plazos	num_pago	cuantia_plazo	cod_pedido	pago_master
14	3	24000	6	1	2100.025	P_00006	NULL
14	3	24000	6	1	2100.025	P_00006	NULL
14	3	24000	6	1	2100.025	P_00006	NULL
14	3	24000	6	1	2100.025	P_00006	NULL
14	3	24000	6	1	2100.025	P_00006	NULL
14	3	24000	6	1	2100.025	P_00006	NULL

12.- Obtener los pagos aplazados del pedido P_00005.

```

SELECT pa1.*
FROM Pagos pa1
INNER JOIN Pagos pa2 ON pa1.cod_pago=pa2.pago_master
WHERE pa2.pago_master = 9;

```

cod_pago	cod_cliente	cantidad	num_plazos	num_pago	cuantia_plazo	cod_pedido	pago_master
9	3	8400.1	4	1	2100.025	P_00005	NULL
9	3	8400.1	4	1	2100.025	P_00005	NULL
9	3	8400.1	4	1	2100.025	P_00005	NULL
9	3	8400.1	4	1	2100.025	P_00005	NULL