SQL Statements & Outputs

1.
SELECT employees.employeeid, employees.employee_firstname, job_history.job_title, job_history.jobstartdate, job_history.jobenddate
FROM employees
INNER JOIN job_history
ON employees.employeeID = job_history.employeeID
ORDER BY employee_firstname;

```
rjoshi26=> SELECT employees.employeeid, employees.employee_firstname, job_history.job_title, job_history.jobstartdate, job_history.job
enddate
rjoshi26-> FROM employees
rjoshi26-> INNER JOIN job_history
rjoshi26-> ON employees.employeeID = job_history.employeeID
rjoshi26-> ORDER BY employee_firstname;
 employeeid | employee_firstname |     job_title     | jobstartdate | jobenddate
------------+--------------------+-------------------+--------------+------------
          1 | Bethany            | Junior Doctor     | 2008-05-17   | 2010-05-16
          1 | Bethany            | Doctor            | 2010-05-17   |
          9 | Cain               | Inventory Analyst | 2003-08-25   | 2010-05-16
          9 | Cain               | Inventory Manager | 2010-05-17   |
          2 | Cooper             | Assistant Doctor  | 2009-04-01   | 2010-05-16
          2 | Cooper             | Doctor            | 2010-05-17   |
          7 | Gillian            | IT Generalist     | 2007-09-04   | 2010-05-16
          7 | Gillian            | IT Specialist     | 2010-05-17   |
         10 | Kieran             | Junior Doctor     | 2010-05-17   |
          4 | Leah               | Nurse             | 2010-05-17   |
          5 | Lucian             | Doctor            | 2010-05-17   |
          3 | Noble              | Nurse             | 2010-05-17   |
          8 | Oleg               | Cleaning Staff    | 2010-05-17   |
          6 | Oprah              | Support Resource  | 2003-08-01   | 2010-05-16
          6 | Oprah              | Receptionist      | 2010-05-17   |
(15 rows)
```

2.
SELECT patientid, appointmentid, scheduled_date, scheduled_time FROM appointments
WHERE scheduled_date BETWEEN '2019-10-01' AND '2019-10-07' AND memployeeid = 1;

```
rjoshi26=> SELECT patientid, appointmentid, scheduled_date, scheduled_time FROM appointments
rjoshi26-> WHERE scheduled_date BETWEEN '2019-10-01' AND '2019-10-07' AND memployeeid = 1;
 patientid | appointmentid | scheduled_date | scheduled_time
-----------+---------------+----------------+----------------
         1 |             1 | 2019-10-02     | 09:00:00
         1 |             2 | 2019-10-03     | 09:00:00
         1 |             3 | 2019-10-04     | 09:00:00
         1 |             4 | 2019-10-05     | 09:00:00
         1 |             5 | 2019-10-06     | 09:00:00
(5 rows)
```

3.

SELECT inventoryID, medications.medication_name,sum(quantity_consumed) FROM
inventory_usage_tracker INNER JOIN medications
ON inventory_usage_tracker.inventoryid = medications.minventoryid WHERE complaintID = 3
AND inventoryID IN (SELECT minventoryid from medications)
GROUP BY inventoryID, medications.medication_name;

```
rjoshi26=> SELECT inventoryID, medications.medication_name,sum(quantity_consumed) FROM inventory_usage_tracker INNER JOIN medications
rjoshi26-> ON inventory_usage_tracker.inventoryid = medications.minventoryid WHERE complaintID = 3
rjoshi26-> AND inventoryID IN (SELECT minventoryid from medications)
rjoshi26-> GROUP BY inventoryID, medications.medication_name;                                        ]
 inventoryid | medication_name | sum
-------------+-----------------+-----
           4 | Tylanol         |   7
(1 row)
```

4.

SELECT inventoryID, consumables.consumable_name,sum(quantity_consumed) FROM
inventory_usage_tracker INNER JOIN consumables
ON inventory_usage_tracker.inventoryid = consumables.cinventoryid WHERE complaintID = 3
AND inventoryID IN (SELECT cinventoryid from consumables)
GROUP BY inventoryID, consumables.consumable_name;

```
rjoshi26=> SELECT inventoryID, consumables.consumable_name,sum(quantity_consumed) FROM inventory_usage_tracker INNER JOIN consumables
rjoshi26-> ON inventory_usage_tracker.inventoryid = consumables.cinventoryid WHERE complaintID = 3
rjoshi26-> AND inventoryID IN (SELECT cinventoryid from consumables)
rjoshi26-> GROUP BY inventoryID, consumables.consumable_name;
 inventoryid | consumable_name | sum
-------------+-----------------+-----
           1 | Gloves          |   3
           2 | Masks           |   3
(2 rows)
```

5.

SELECT claims.patientid, complaints_appointment_tracker.appointmentid,
claims.amount_submitted, claims.amount_covered
FROM claims
INNER JOIN complaints_appointment_tracker
ON claims.complaintid = complaints_appointment_tracker.complaintid WHERE
complaints_appointment_tracker.appointmentid = 6;

```
rjoshi26=> SELECT claims.patientid, complaints_appointment_tracker.appointmentid, claims.amount_submitted, claims.amount_covered
rjoshi26-> FROM claims
rjoshi26-> INNER JOIN complaints_appointment_tracker
rjoshi26-> ON claims.complaintid = complaints_appointment_tracker.complaintid WHERE complaints_appointment_tracker.appointmentid = 6;
 patientid | appointmentid | amount_submitted | amount_covered
-----------+---------------+------------------+----------------
         2 |             6 |          8300.90 |        7700.90
(1 row)
```

_____

SQL Statements for Purchase Order:
To view daily order requirements, run the following SQL Statement:

SELECT DISTINCT purchase_order.vendorID, purchase_order.vendor_name,
inventory_usage_view.inventoryID, purchase_order.product_name, inventory_usage_view.sum
FROM purchase_order INNER JOIN inventory_usage_view
ON purchase_order.inventoryid = inventory_usage_view.inventoryID;

```
rjoshi26=> SELECT DISTINCT purchase_order.vendorID, purchase_order.vendor_name, inventory_usage_view.inventoryID, purchase_order.produ
ct_name, inventory_usage_view.sum
rjoshi26-> FROM purchase_order INNER JOIN inventory_usage_view
rjoshi26-> ON purchase_order.inventoryid = inventory_usage_view.inventoryID;
 vendorid |     vendor_name      | inventoryid | product_name | sum
----------+----------------------+-------------+--------------+-----
        1 | Molestie Tor         |           2 | Masks        |   3
        5 | Sapien Industries    |           4 | Tylanol      |   4
        6 | Aliquam Corporation  |           6 | Cylanol      |  15
(3 rows)
```