# Tekkis Python Developer Assessment

## Task 1

A.

1. SELECT \* FROM customer WHERE customer.gender = ‘female’

2.

B.

1. SELECT

2. CONCAT(customer.first\_name, ' ', customer.last\_name) AS name,

3. COUNT(order.id) as number\_of\_orders

4. FROM

5. order

6. JOIN

7. customer on order.fk\_customer = customer.id

8. GROUP BY

9. name;

10.

C.

1. SELECT

2. customer.\*,

3. SUM(order.sum) as money\_spend

4. FROM

5. order

6. JOIN

7. customer on order.fk\_customer = customer.id

8. GROUP BY

9. customer.id;

10.

D.

1. SELECT

2. order.order\_nr

3. FROM

4. order

5. JOIN

6. order\_item on order\_item.fk\_order = order.id

7. GROUP BY

8. order.id

9. HAVING

10. COUNT(order\_item.fk\_order)>=2;

11.

## Task 2

Normalized Data Models

**Advantages:**

1. **Reduced Data Duplication:** Avoids storing the same data in multiple places, leading to efficient use of storage.
2. **Improved Data Integrity:** Less risk of data anomalies and inconsistencies due to the single source of truth.
3. **Easier Maintenance and Updates:** Changes to data need to be made in only one place.
4. **Facilitates Enforcement of Constraints:** Easier to apply data rules and constraints at the table level.

**Disadvantages:**

1. **Complexity in Design:** More tables and relationships to manage, which can be harder to understand.
2. **Slower Query Performance:** Requires more joins and lookups, potentially slowing down data retrieval.
3. **Less Flexibility:** May not be ideal for certain types of data or applications needing quick read access and aggregation.

Denormalized Data Models

**Advantages:**

1. **Faster Query Performance:** Fewer joins and lookups are needed, speeding up data retrieval.
2. **Simplified Queries:** Easier access to data with fewer tables and columns involved.
3. **Suitable for Specific Use Cases:** Beneficial for reporting and analytical purposes where quick read access is crucial.

**Disadvantages:**

1. **Increased Storage Space:** Storing duplicate data leads to higher storage requirements.
2. **Risk of Data Inconsistencies:** Potential for anomalies and inconsistencies due to multiple data sources.
3. **Complex Maintenance and Updates:** Data modifications need to be made in multiple places.
4. **Challenges in Data Constraint Enforcement:** Rules and constraints need to be managed at multiple places, increasing complexity.