

1) Always Capitalize Keywords

- Keywords such as `SELECT`, `WITH`, `SUM`, `WHERE`, etc. should *always* be capitalized.

Correct Formatting

```
1 SELECT
2   name    AS name,
3   age     AS age,
4   gender  AS gender
5 FROM users
```

Incorrect Formatting

```
1 -- Keywords should be capitalized
2 select
3   name    AS name,
4   age     AS age,
5   gender  AS gender
6 from users
```

2) Naming

Names should be in singular

Correct Formatting

```
1 SELECT
2   *
3 FROM m_loan AS loan
```

Incorrect Formatting

```
1 -- Loans alias is in plural
2 SELECT
3   *
4 FROM m_loan AS loans
```

Names should be in `snake_case`

- Any table, variable, CET, column name etc. should be in `snake_case` (lowercase words separated by underscores)
- Names should only contain letters, numbers and underscores

Correct Formatting

```
1 SELECT
2   loan.id      AS loan_id,
3   client.name  AS client_name,
4   client.town  AS client_town
5 FROM m_loan AS loan
6 INNER JOIN m_client AS client ON
7   loan.client_id = client.id
8 DECLARE start_date DATE DEFAULT '2021-09-01'
```

Incorrect Formatting

```
1 -- Column name not in snake_case
2 -- Column name containing spaces or dashes
3 SELECT
4   loan.id      AS Loan_Id,
5   client.name  AS "Client Name",
6   client.town  AS "Client-town"
7 FROM m_loan AS loan
8 INNER JOIN m_client AS client ON
9   loan.client_id = client.id
10 -- Variable name is not in snake_case
11 DECLARE startdate DATE DEFAULT '2021-09-01'
```

Names should be descriptive [↗](#)

Correct Formatting

```
1 SELECT
2   loan.id      AS loan_id,
3   client.name  AS client_name,
4   client.town  AS client_town
5 FROM m_loan AS loan
6 INNER JOIN m_client AS client ON
7   loan.client_id = client.id
```

Incorrect Formatting

```
1 -- Column names are not descriptive enough, we don't know what type of name, town or id they are
2 -- Table names with one letter very bad for readability - it's hard to know what they are outside
3 SELECT
4   ml.id  AS id,
5   mc.name AS name,
6   mc.town AS town
7 FROM m_loan AS ml
8 INNER JOIN m_client AS mc ON
9   l.client_id = c.id
```

Use the AS keyword for aliasing

- The `AS` keyword is optional in SQL, but omitting it makes it very difficult to read SQL - especially for beginners. To make reading our SQL easier, we should always use the `AS` keyword.

Correct Formatting

```
1 SELECT
2   user_first_name AS first_name
3 FROM client
4 SELECT
5   client.user_first_name AS first_name,
6   date_of_birth.dob      AS birthday
7 FROM client
8 INNER JOIN dobs AS date_of_birth ON
9   client.id = date_of_birth.client_id
```

Incorrect Formatting

```
1 -- missing AS keyword for column alias
2 SELECT
3   user_first_name first_name
4 FROM client
5 -- missing AS keyword for table alias
6 SELECT
7   client.user_first_name AS first_name,
8   date of birth.dob      AS birthday
```

Boolean field names should start with `has_`, `is_`, or `does_`

Correct Formatting

```
1 SELECT
2   IF(end_date IS NULL, 1, 0) AS is_active,
3   IF(sla IS NOT NULL, 1, 0) AS has_sla
4 FROM loan
```

Incorrect Formatting

```
1 -- Column names not suitable for boolean values
2 SELECT
3   IF(end_date IS NULL, 1, 0) AS active,
4   IF(sla IS NOT NULL, 1, 0) AS sla
5 FROM loan
```

2) Indentation must be 2 spaces per indent with no tabs

Correct Formatting

```
1 SELECT
2   user_first_name AS first_name
3 FROM client
```

Incorrect Formatting

```
1 -- 4 spaces for indentation instead of 2
2 SELECT
3     user_first_name AS first_name
4 FROM client
```

3) Use one space after coma

Correct Formatting

```
1 DATE_ADD(sale.date_of_sale, INTERVAL 1 DAY)
```

Incorrect Formatting

```
1 -- Missing space after the comma
2 DATE_ADD(sale.date_of_sale,INTERVAL 1 DAY)
```

4) Use spaces between infix operators

- "Infix Operators" are usually mathematical symbols such as `+`, `/`, and `=`. Infix operators should always be spaced.

Correct Formatting

```
1 INNER JOIN dim_payment_plan AS payment_plan ON
2   payment_plan.payment_plan_id = master_mgmt.payment_plan_id
```

Incorrect Formatting

```
1 -- Missing space before/after "=" sign
2 INNER JOIN dim_payment_plan AS payment_plan ON
3   payment_plan.payment_plan_id=master_mgmt.payment_plan_id
```

5) Parentheses

- The opening parenthesis should terminate the line.
- The closing parenthesis should be lined up under the first character of the line that starts the multi-line construct.
- The contents of the parentheses should be indented one level.

Correct Formatting

```
1 WITH sample AS (  
2     SELECT  
3         client_id,  
4     FROM main_summary  
5     WHERE  
6         sample_id = '42'  
7 )
```

Incorrect Formatting

```
1 -- No indentation inside the parentheses  
2 -- Terminating parenthesis on shared line  
3 WITH sample AS (  
4     SELECT  
5         client_id,  
6         client_name  
7 )
```

6) CTEs, Temporary Tables and Sub-Queries

- Use CTEs and Temporary Tables, not subqueries
- Keep a empty line between blocks of code
- CTEs names should be descriptive

Correct Formatting

```
1 WITH client_sample AS (  
2     SELECT  
3         client_id,  
4         client_name  
5     FROM main_summary  
6     WHERE  
7         sample_id = '42'  
8 )  
9  
10 SELECT  
11     *  
12 FROM client_sample  
13 LIMIT 10
```

Incorrect Formatting

```
1  -- No empty line between the CTE and the SELECT statement
2  -- CTE name is not descriptive
3  WITH a AS (
4      SELECT
5          client_id,
6          client_name
7      FROM main_summary
8      WHERE
9          sample_id = '42'
10 )
11 SELECT
12     *
13 FROM a
14 LIMIT 10
15 -- Sub-query used instead of a CTE
16 SELECT
17     *
18 FROM (
19     SELECT
20         client_id,
21         client_name
22     FROM main_summary
```

1) SELECT

- Align column names aliases at AS
- Always specify the table for each column when multiple tables are joined, even if the column is in one table only

Correct Formatting

```
1  SELECT
2      loan.id      AS loan_id,
3      client.id    AS client_id,
4      client.name  AS client_name,
5      client.town  AS client_town
6  FROM m_loan AS loan
7  INNER JOIN m_client AS client ON
8      loan.client_id = client.id
```

Incorrect Formatting

```
1  -- Column aliases not alligned
2  -- "name" and "town" fields don't have a table name
3  SELECT
4      loan.id AS loan_id,
5      client.id AS client_id,
6      name AS client_name,
```

2) CASE and IF() statements

- Follow the following syntax for CASE statements
- Follow the following syntax for nested IF() statements
- Prefer IF() statements where there is a single condition

Correct Formatting

```
1 CASE
2   WHEN
3     country_code = 'KE' AND
4     town IS NULL
5   THEN 'Kenya'
6   WHEN
7     country_code = 'UG'
8   THEN 'Uganda'
9   ELSE 'Other'
10 END AS country_name
11 -- IF() statement with a single condition
12 IF(country_code = 'KE', 'Kenya', 'Other') AS country_name
13
14 -- Nested IF() statement
15 IF((country_code = 'KE' AND town IS NULL), 'Kenya',
16   IF(country_code = 'UG', 'Uganda',
17     IF(country_code = 'NG', 'Nigeria',
```

Incorrect Formatting

```
1 -- No indentation, conditions at the wrong locations
2 CASE
3 WHEN country_code = 'KE' AND town IS NULL
4 THEN 'Kenya'
5 WHEN country_code = 'UG'
6 THEN 'Uganda'
7 ELSE 'Other'
8 END AS country_name
9 -- Incorrect indentation
10 IF(country_code = 'KE',
11   'Kenya',
12   'Other') AS country_name
13
14 -- Incorrect indentation and line breaks
15 IF((country_code = 'KE' AND town IS NULL), 'Kenya',
16 IF(country_code = 'UG', 'Uganda',
17 IF(country_code = 'NG', 'Nigeria', 'Other')))) AS country_name
```

3) FROM

- Good practice: specify the database, schema / dataset and table name: it allows better understanding of where the data is coming from and it can be ran in different environments.

Good

```
1 SELECT
2   loan.id      AS loan_id,
3   client.name AS client_name,
4   client.town AS client_town
5 FROM dwanalytics.core.m_loan AS loan
6 INNER JOIN dwanalytics.core.m_client AS client ON
7   loan.client_id = client.id
```

Not as good

```
1 SELECT
2   loan.id      AS loan_id,
3   client.name AS client_name,
4   client.town AS client_town
5 FROM m_loan AS loan
6 INNER JOIN m_client AS client ON
7   loan.client_id = client.id
```

4) JOINS

- Explicitly define the type of join, do not use JOIN by itself
- ON should be at the end of the line and the columns joins should be below and indented

Correct formatting

```
1 SELECT
2   loan.id      AS loan_id,
3   client.name AS client_name
4 FROM dwanalytics.core.m_loan AS loan
5 INNER JOIN dwanalytics.core.m_client AS client ON
6   loan.client_id = client.id AND
7   loan.country = client.country
```

Incorrect formatting

```
1 -- JOIN used instead of INNER JOIN
2 -- Columns join "loan.client_id = client.id" is not indented in row following the table name
3 SELECT
4   loan.id      AS loan_id,
5   client.name AS client_name
6 FROM dwanalytics.core.m_loan AS loan
7 JOIN dwanalytics.core.m_client AS client ON loan.client_id = client.id AND
```


5) WHERE / HAVING

- Every argument must have its own line and be indented
- AND / OR should be at end of lines
- Multiple values for a filter should be indented from the condition and each value in its own line

Correct formatting

```
1  SELECT
2      *
3  FROM m_loan
4  WHERE
5      disbursed_on_date = CURRENT_DATE() AND
6      (country = 'KE' OR
7       country IS NULL) AND
8      town IN (
9          'Naivasha',
10         'Nairobi',
11         'Mombasa')
```

Incorrect formatting

```
1  -- AND at start of line
2  -- Country filter not indented on its own line
3  -- town filtered values are not indented on their own line
4  SELECT
5      *
6  FROM m_loan
7  WHERE
8      disbursed_on_date = CURRENT_DATE()
9  AND (country = 'KE' OR country IS NULL)
10 AND town IN ('Naivasha', 'Nairobi', 'Mombasa')
```

Full Example

Here is a query that is in use right now. How easy is it to read when you first look at it? How much harder would it be if the query was 1000 lines long?

There is nothing wrong with this query. It is correct SQL, and implements complex logic. It is very good at what it does. The original author would have required significant knowledge of our database structure, and very solid SQL skills to write it. But, if everyone writes complex queries according to their own personal style, it becomes very hard to share knowledge or debug other people's work. A consistent style guide will significantly improve the engineering standards of the Data Analytics team, and will bring us more in line with industry best practices.

As you're comparing these queries, ask yourself the following questions:

- Which format makes the query easier to understand in less than 5 minutes?
- Which query will produce a nicer output table for the business user?
- Which format makes it easier to spot potential bugs or logic issues?
- Which format is easier to edit and maintain?

Original formatting [↗](#)

```
1  SELECT
2    *, IF(asset_released_at = process_created_at, 1, 0) sameDay, IF(process_state IN ('ASSET_DATA', 'ASSET_RELEASED', 'ASSET_TRANSFERRED', 'ASSET_REPOSSESSED', 'ASSET_RESTRUCTURED', 'ASSET_TRANSFERRING', 'ASSET_REPOSSESSING', 'ASSET_RESTRUCTURING', 'ASSET_TRANSFERRING'), 1, 0) assetState,
3  FROM (SELECT
4    l.external_id id, l.country, p.id id2, p.state state, p.created_at p_created_at, l.asset_released_at l_asset_released_at,
5    ifnull(cv.code_value, d.`Dealership Name`) "Dealership Name"
6  from `bigquerysets.raw_data_combined.db_onboarding_loan` l
7  left join `db_onboarding_process` p ON l.id = p.loan_id AND l.country = p.country
8  LEFT JOIN `db_mifostenant-default_dealerships` d ON d.loan_id = l.external_id AND d.country = l.country
9  Left join `bigquerysets.raw_data_combined.db_mifostenant-default_m_code_value` cv ON
10 cv.id = d.Dealership_Name AND
11 cv.country = d.country
12 where p.created_at > '2021-01-01') a
13 where ("Dealership Name" not in ('Repossessed', 'Restructured', 'Transferred') or
14 "Dealership Name" IS NULL)
15 and l.country = 'KE'
```

Style Guide Formatting

```
1 WITH onboarded_loans AS (  
2   SELECT  
3     onboarding_loan.external_id      AS loan_id,  
4     onboarding_loan.country         AS country,  
5     process.id                     AS process_id,  
6     process.state                   AS process_state,  
7     process.created_at              AS process_created_at,  
8     onboarding_loan.asset_released_at AS asset_released_at,  
9     IFNULL(  
10      code_value_dealership.code_value,  
11      dealership.`Dealership_Name`) AS dealership  
12 FROM `bigquerysets.raw_data_combined.db_onboarding_loan` AS onboarding_loan  
13 LEFT JOIN `bigquerysets.raw_data_combined.db_onboarding_process` AS process ON  
14   onboarding_loan.id = process.loan_id AND  
15   onboarding_loan.country = process.country  
16 LEFT JOIN `bigquerysets.raw_data_combined.db_mifostenant-default_dealerships` AS dealership  
17   dealership.loan_id = onboarding_loan.external_id AND  
18   dealership.country = onboarding_loan.country  
19 LEFT JOIN `bigquerysets.raw_data_combined.db_mifostenant-default_m_code_value` AS code_value  
20   code_value_dealership.id = dealership.Dealership_Name AND  
21   code_value_dealership.country = dealership.country  
WHERE  
  process.created_at > '2021-01-01'  
)  
  
SELECT  
  *,  
  IF(asset_released_at = process_created_at, 1, 0) AS is_released_same_day,  
  IF(process_state IN ('ASSET_DATA', 'AGREEMENT_SIGN'), 1, 0) AS is_pending  
FROM onboarded_loans  
WHERE  
  (dealership NOT IN (  
    'Repossessed',  
    'Restructured',  
    'Transferred') OR  
  dealership IS NULL) AND  
  country = 'KE'
```