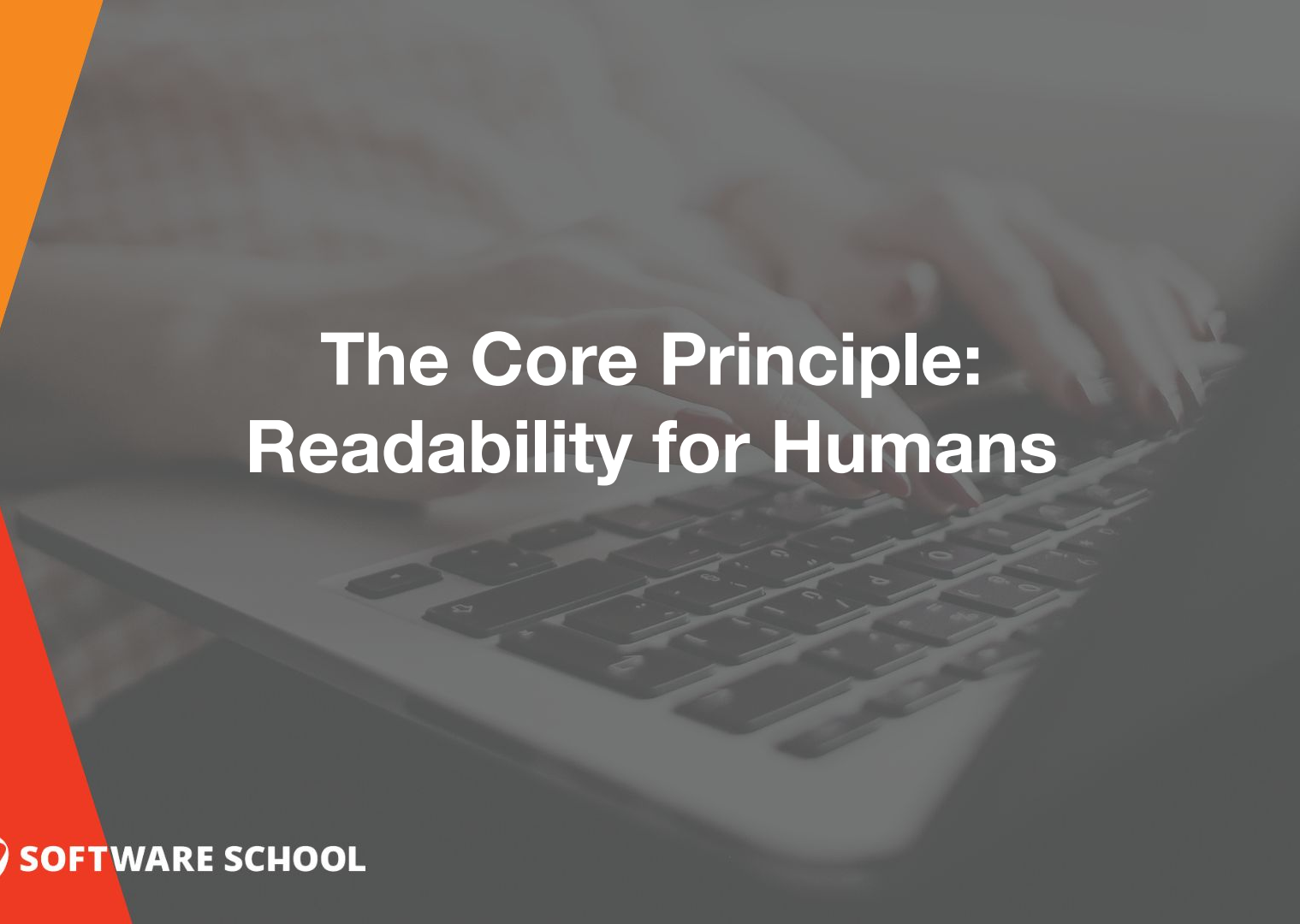
Best Practices for Writing and Formatting Code

Optimizing SQL for Analytics in the Cloud

Objectives

1. Differentiate between poorly and well-formatting SQL, and articulate the business value of readability
2. Apply industry-standard conventions for casing and indenting
3. Write robust, production-ready queries by explicitly listing columns, defining JOIN types, and using clear aliases
4. Improve code maintainability by incorporating strategic and useful comments
5. Conform to team-specific formatting and documentation standards

The Core Principle: Readability for Humans



NASHVILLE SOFTWARE SCHOOL

Why Formatting Matters

SQL doesn't *require* specific formatting (line breaks, indents, casing), but **humans do.**

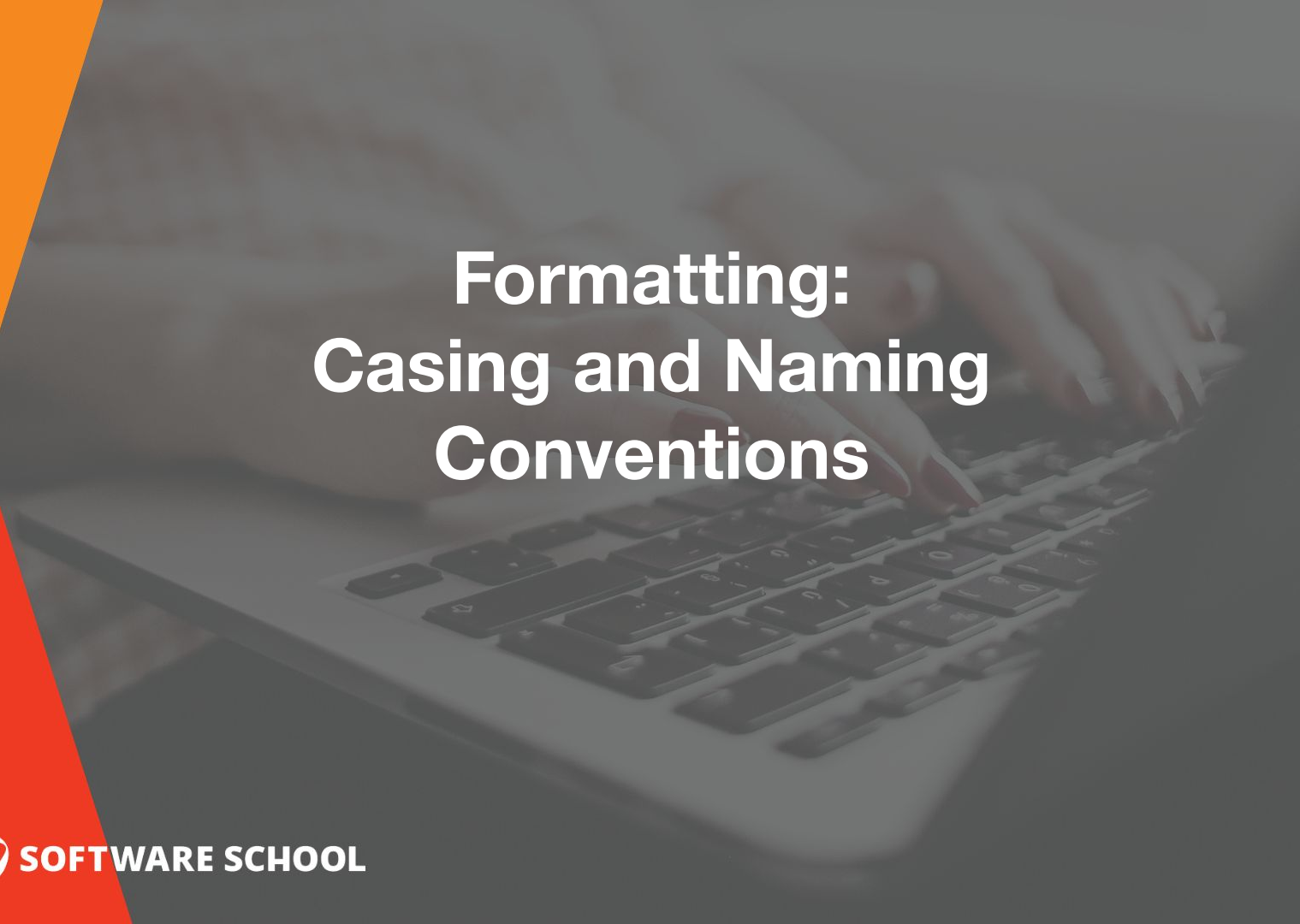
```
select
person_id,gender_concept_id,year_of_birth,month_of_birth,day_of_birth,race_concept_id,ethnicity_concept_id,location_id from person where year_of_birth > 1950 and month_of_birth = 1 and day_of_birth = 1 and race_concept_id = 8527
```

Why Formatting Matters

You will be judged professionally on the readability and maintainability of your code.

```
SELECT person_id,
        gender_concept_id,
        year_of_birth,
        month_of_birth,
        day_of_birth,
        race_concept_id,
        ethnicity_concept_id,
        location_id
FROM person
WHERE year_of_birth > 1950
        AND month_of_birth = 1
        AND day_of_birth = 1
        AND race_concept_id = 8527
;
```

Formatting: Casing and Naming Conventions



NASHVILLE SOFTWARE SCHOOL

Uppercase Keywords

Keywords should be in **UPPERCASE**

- Most common keywords: SELECT, FROM, WHERE, JOIN

This makes them visually pop and quickly outlines the query's structure.

```
SELECT person_id,
        gender_concept_id,
        year_of_birth,
        month_of_birth,
        day_of_birth,
        race_concept_id,
        ethnicity_concept_id,
        location_id
FROM person
WHERE year_of_birth > 1950
        AND month_of_birth = 1
        AND day_of_birth = 1
        AND race_concept_id = 8527
;
```

Lowercase, Snake_Case Names

Use **lowercase, snake_case** for database objects

- Examples of database objects: tables, columns, views

This is a widely accepted standard that improves readability.

Example:

- year_of_birth vs yearOfBirth
- You will often see the the second (called Camel Case) when data is coming from an API and dropped into the database in its raw form

Descriptive Aliases

Alias columns and tables with names that are descriptive and concise.

Avoid ambiguous or non-standard abbreviations.

What would you change in the given example?

```
SELECT person_id AS p_id,
        gender_concept_id AS gc_id,
        year_of_birth AS yob,
        month_of_birth AS mob,
        day_of_birth AS dob,
        race_concept_id AS rc_id,
        ethnicity_concept_id AS ec_id,
        location_id as l_id
FROM person AS p
WHERE year_of_birth > 1950
        AND month_of_birth = 1
        AND day_of_birth = 1
        AND race_concept_id = 8527
;
```

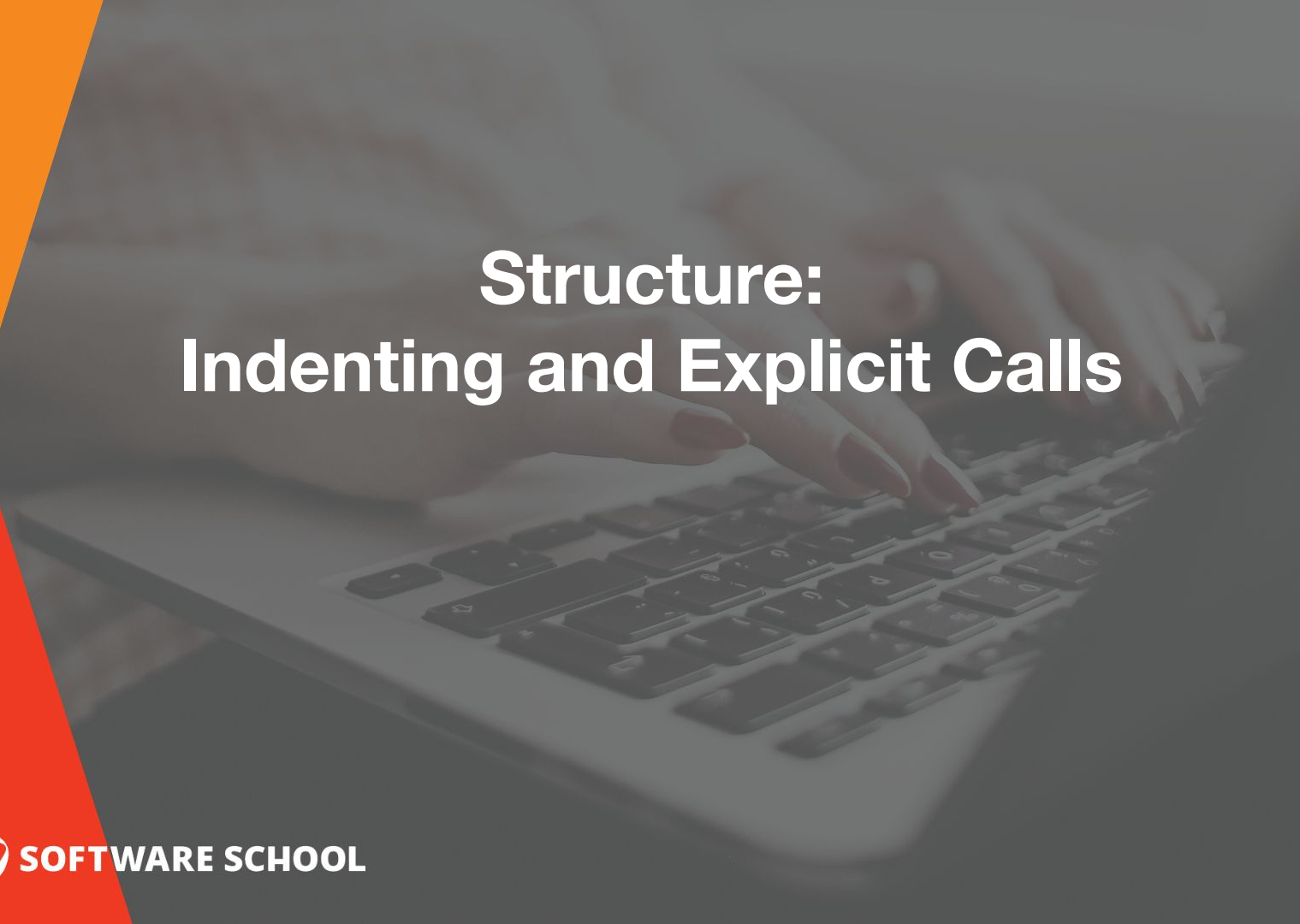
The AS Keyword

Always include the **AS** keyword when aliasing columns or tables.

It is optional in many dialects, but including it clearly signals the aliased name.

```
SELECT person_id pid,
        gender_concept_id gcid,
        year_of_birth yob,
        month_of_birth mob,
        day_of_birth dob,
        race_concept_id rcid,
        ethnicity_concept_id ecid,
        location_id lid
FROM person p
WHERE year_of_birth > 1950
        AND month_of_birth = 1
        AND day_of_birth = 1
        AND race_concept_id = 8527
;
```

Structure:
Indenting and Explicit Calls



NASHVILLE SOFTWARE SCHOOL

Consistent Indenting

Use **consistent indenting** to visually represent the structure and dependencies of your code

- Can be either spaces (typically 2-4) or tabs

```
WITH drug_exposure AS (
        SELECT
                person_id,
                drug_concept_id,
                MIN(drug_exposure_start_date)
AS first_drug_exposure_date
        FROM drug_exposure
        GROUP BY
                person_id,
                drug_concept_id
)

SELECT
        p.person_id,
        year_of_birth,
        drug_concept_id,
        first_drug_exposure_date
FROM person AS p
LEFT JOIN drug_exposure AS d
        ON p.person_id = d.person_id
ORDER BY first_drug_exposure_date
LIMIT 1000
;
```

Avoid SELECT *

Never use **SELECT** * in production code.

Always **explicitly list columns**.

Why?

- **Data bloat**: processes unnecessary data
- **Fragility**: breaks downstream processes (eg ETL, dashboards) when new columns are added or column order changes
- **Clarity**: readers don't know what columns are returned without checking the table schema

Explicit JOINs

Always explicitly define your JOIN type and use the JOIN clause.

Why?

- It clearly communicates the relationship and prevents accidental cross joins

```
SELECT p.*, d.drug_concept_id, d.drug_exposure_start_date
FROM person AS p, drug_exposure AS d
WHERE p.person_id = d.person_id
AND p.gender_concept_id = 8507
```

Explicit JOINs - Examples

```
SELECT p.*, d.drug_concept_id,
d.drug_exposure_start_date
FROM person p, drug_exposure d
WHERE p.person_id = d.person_id
AND p.gender_concept_id = 8507
```

```
SELECT
    p.person_id,
    p.year_of_birth,
    de.drug_concept_id,
    de.drug_exposure_start_date AS
exposure_date
FROM
    person AS p
INNER JOIN
    drug_exposure AS de
    ON p.person_id = de.person_id
WHERE
    p.gender_concept_id = 8507 -- 8507
corresponds to Male
;
```

Maintenance: Comments and Conventions

NASHVILLE SOFTWARE SCHOOL

Useful, Specific Comments

Be kind to your future self!

Include brief, helpful comments to outline the purpose of the query or specific parts of complex logic

```
/*
Purpose: Retrieve the age and most recent prescription date
for male patients (gender_concept_id 8507) currently living in 1999.
*/
WITH
    -- latest_drug_date: Identifies the single most recent drug
exposure date for every patient.
    latest_drug_date AS (
        SELECT
            person_id,
            MAX(drug_exposure_start_date) AS
most_recent_exposure_date -- most recent date
        FROM
            drug_exposure
        GROUP BY
            person_id
    )

-- join patient demographics with their calculated latest drug date.
SELECT
    p.person_id,
    p.year_of_birth,
    (1999 - p.year_of_birth) AS calculated_age_in_1999,
    ldd.most_recent_exposure_date
FROM
    person AS p
INNER JOIN
    latest_drug_date AS ldd
    ON p.person_id = ldd.person_id
WHERE
    p.gender_concept_id = 8507 -- 8507 is the OMOP Standard
Concept ID for "Male"
    AND p.year_of_birth IS NOT NULL
ORDER BY
    calculated_age_in_1999 DESC
;
```

Follow Team Conventions

If your team or company has a documented set of **code conventions**, conform to them.

Consistency trumps individual preference.

Why?

- This ensures the codebase remains cohesive and easier for the entire team to navigate.

Additional Resources

- SQL Style Guide - Simon Holywell
- SQL Style Guide - Mozilla Data Documentation
- SQL Formatting Best Practices