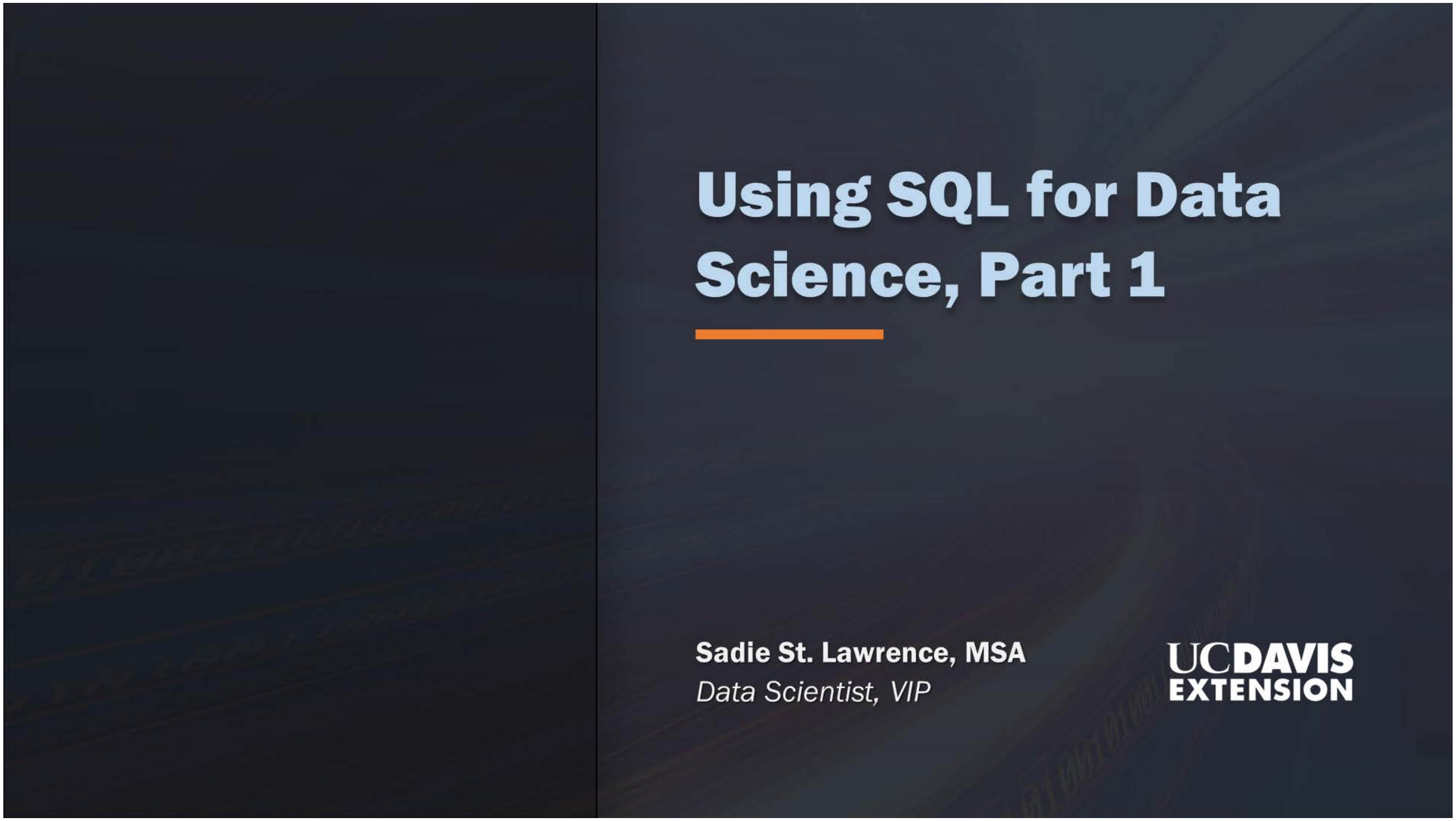
Slide 1: Using SQL for Data Science, Part 1

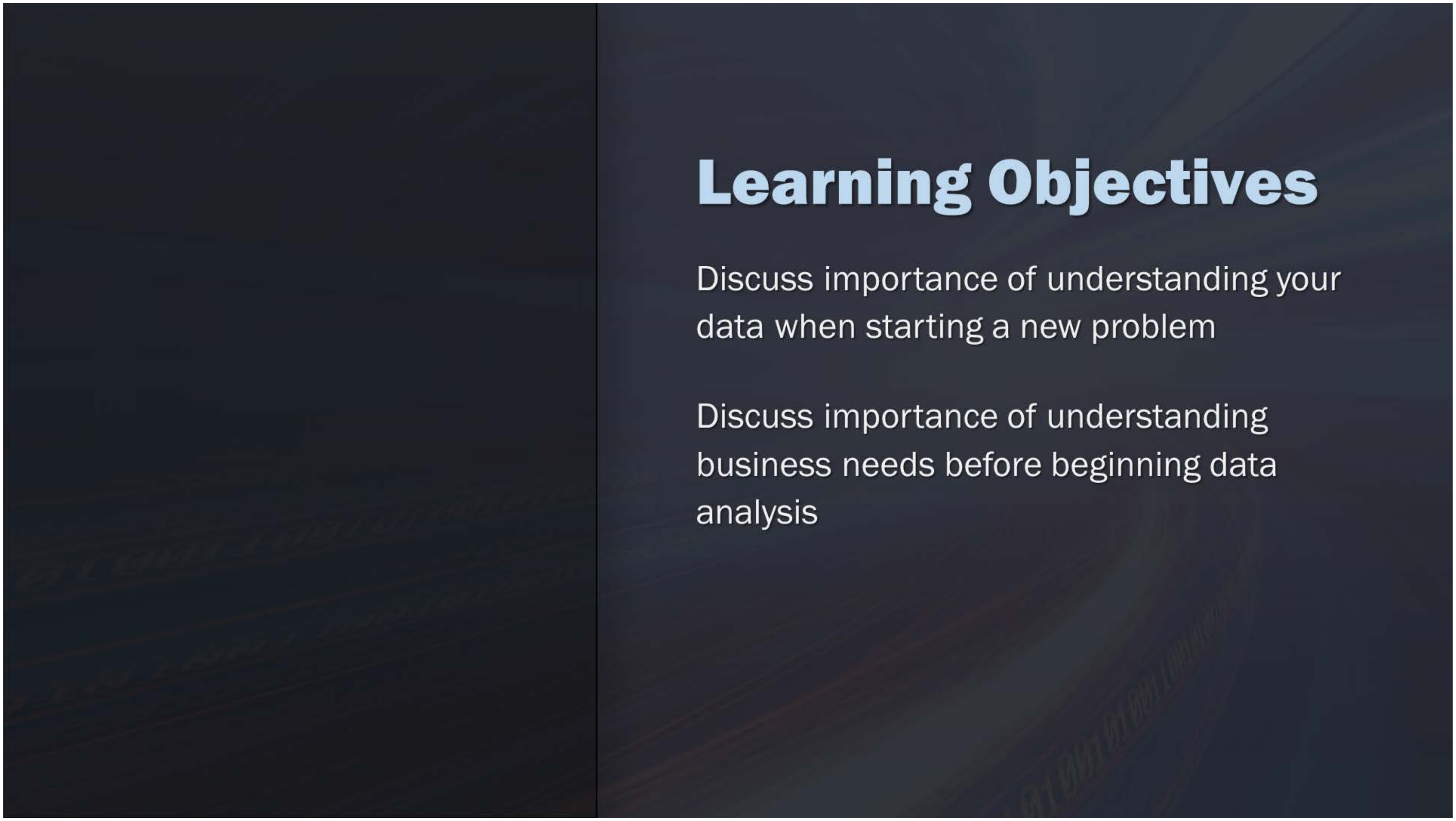


Using SQL for Data Science, Part 1

Sadie St. Lawrence, MSA
Data Scientist, VIP

UCDAVIS EXTENSION

Slide 2: Learning Objectives

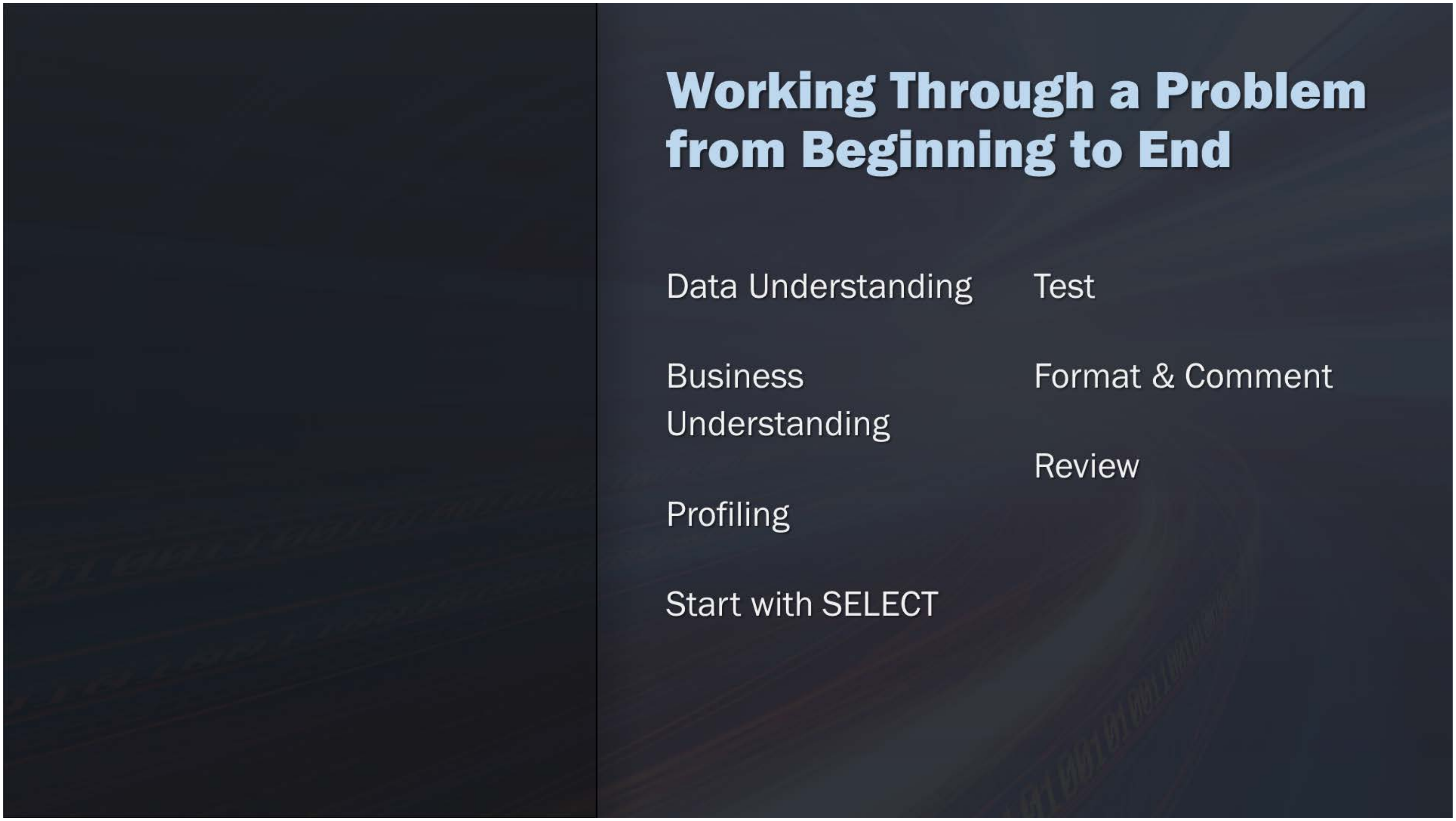


Learning Objectives

Discuss importance of understanding your data when starting a new problem

Discuss importance of understanding business needs before beginning data analysis

Slide 3: Working Through a Problem from Beginning to End



Working Through a Problem from Beginning to End

Data Understanding

Business Understanding

Profiling

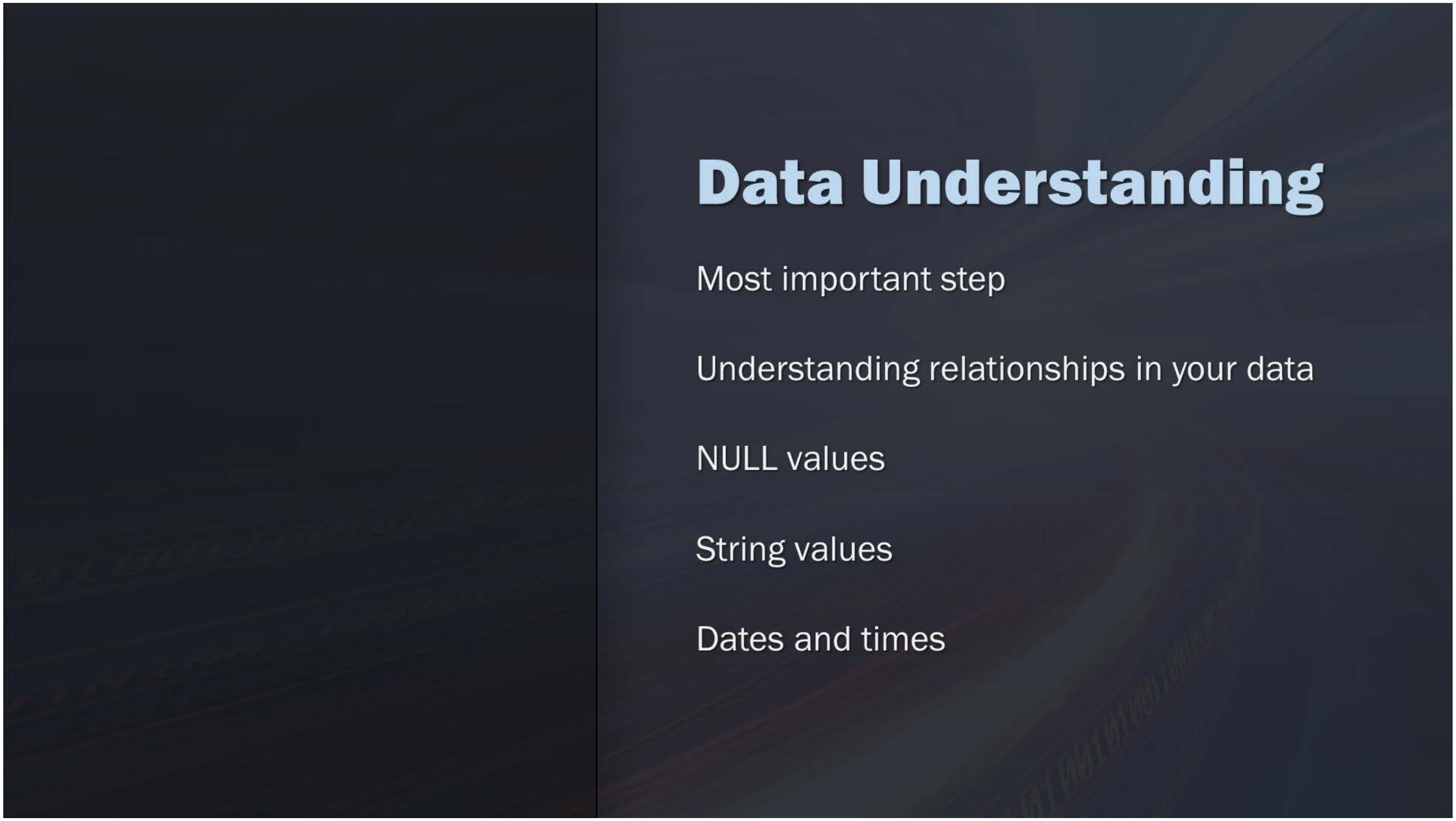
Start with SELECT

Test

Format & Comment

Review

Slide 4: Data Understanding



Data Understanding

Most important step

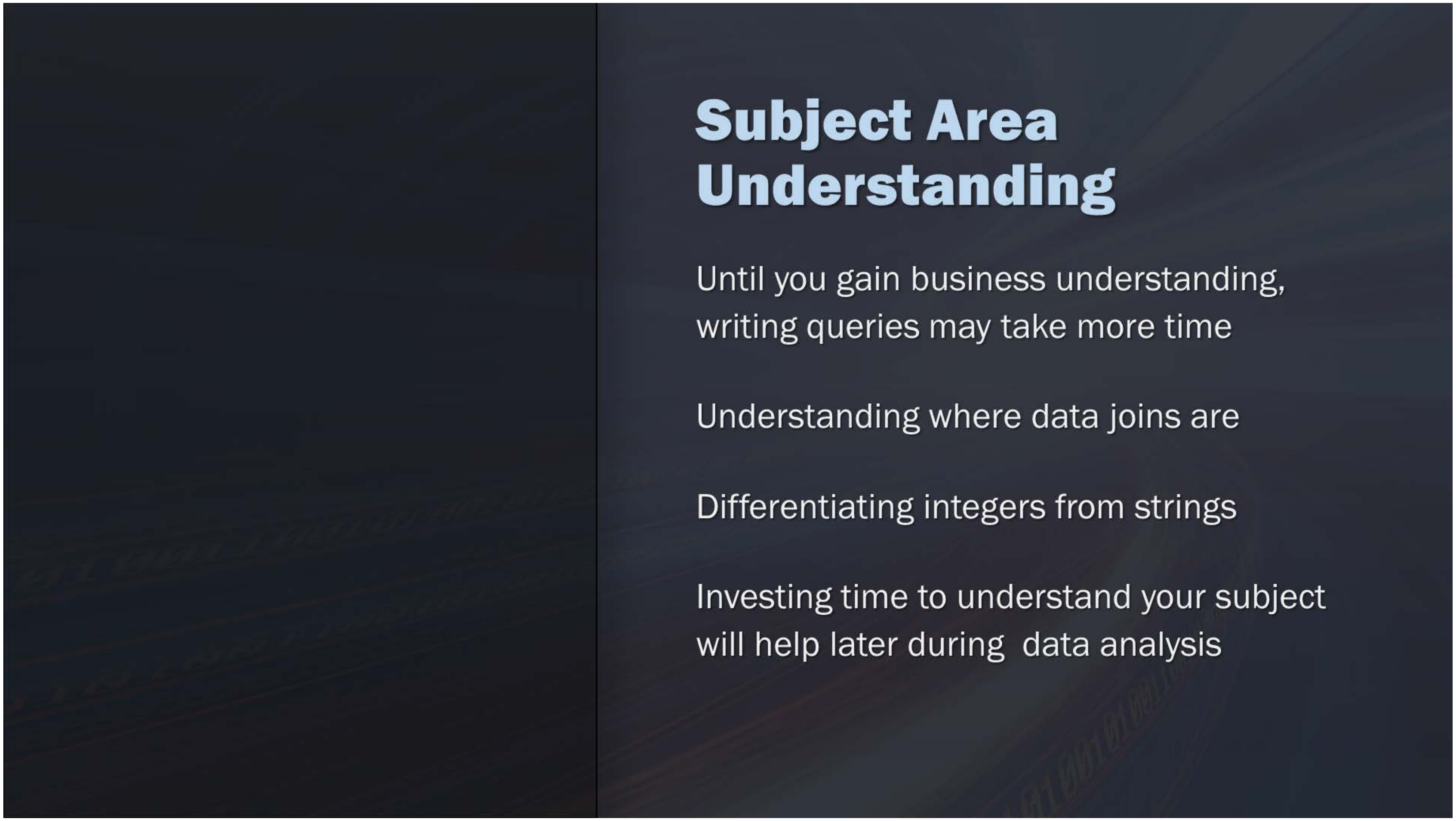
Understanding relationships in your data

NULL values

String values

Dates and times

Slide 5: Subject Area Understanding



Subject Area Understanding

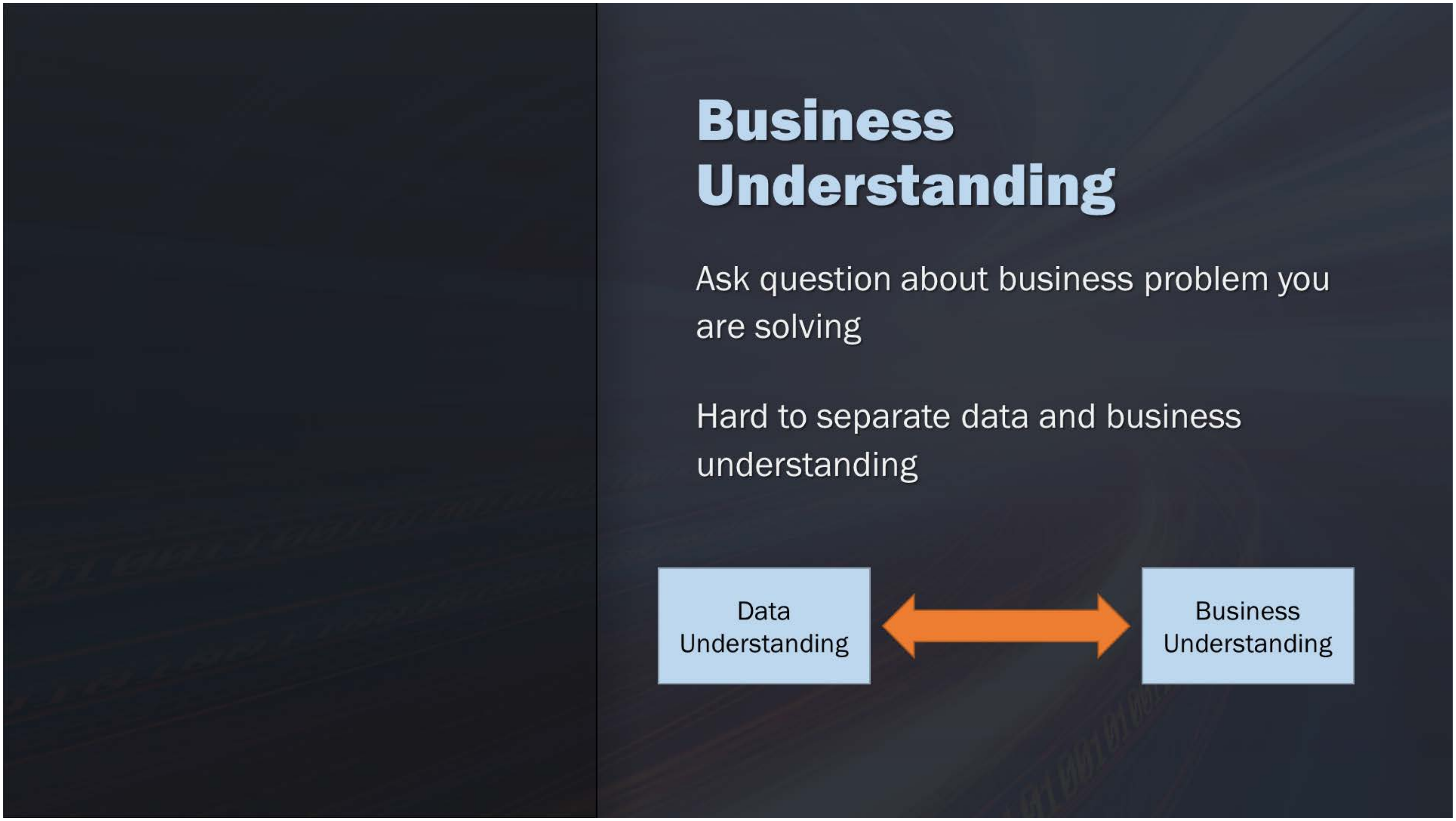
Until you gain business understanding, writing queries may take more time

Understanding where data joins are

Differentiating integers from strings

Investing time to understand your subject will help later during data analysis

Slide 6: Business Understanding



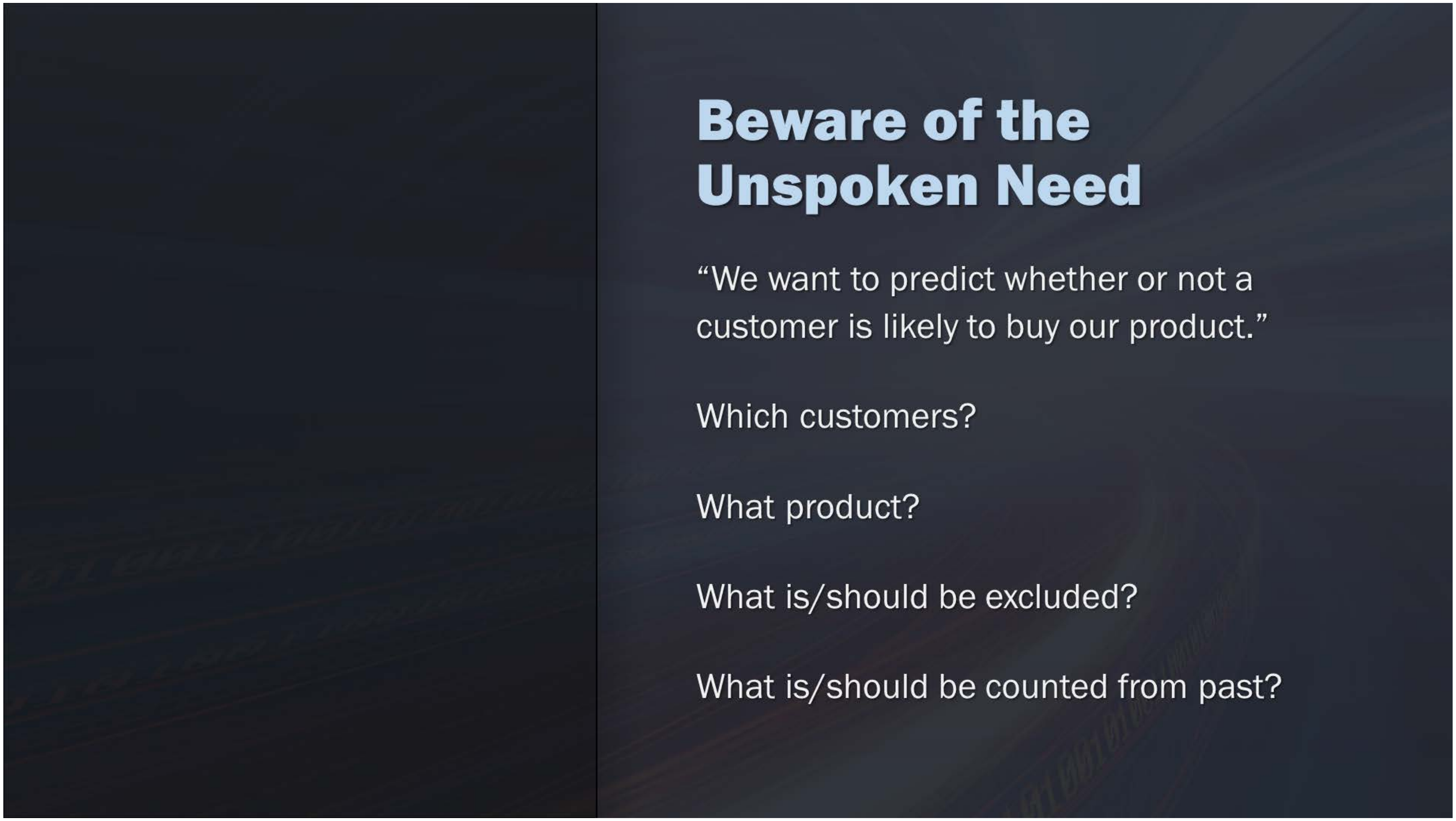
Business Understanding

Ask question about business problem you are solving

Hard to separate data and business understanding

Data Understanding ↔ Business Understanding

Slide 7: Beware of the Unspoken Need



Beware of the Unspoken Need

"We want to predict whether or not a customer is likely to buy our product."

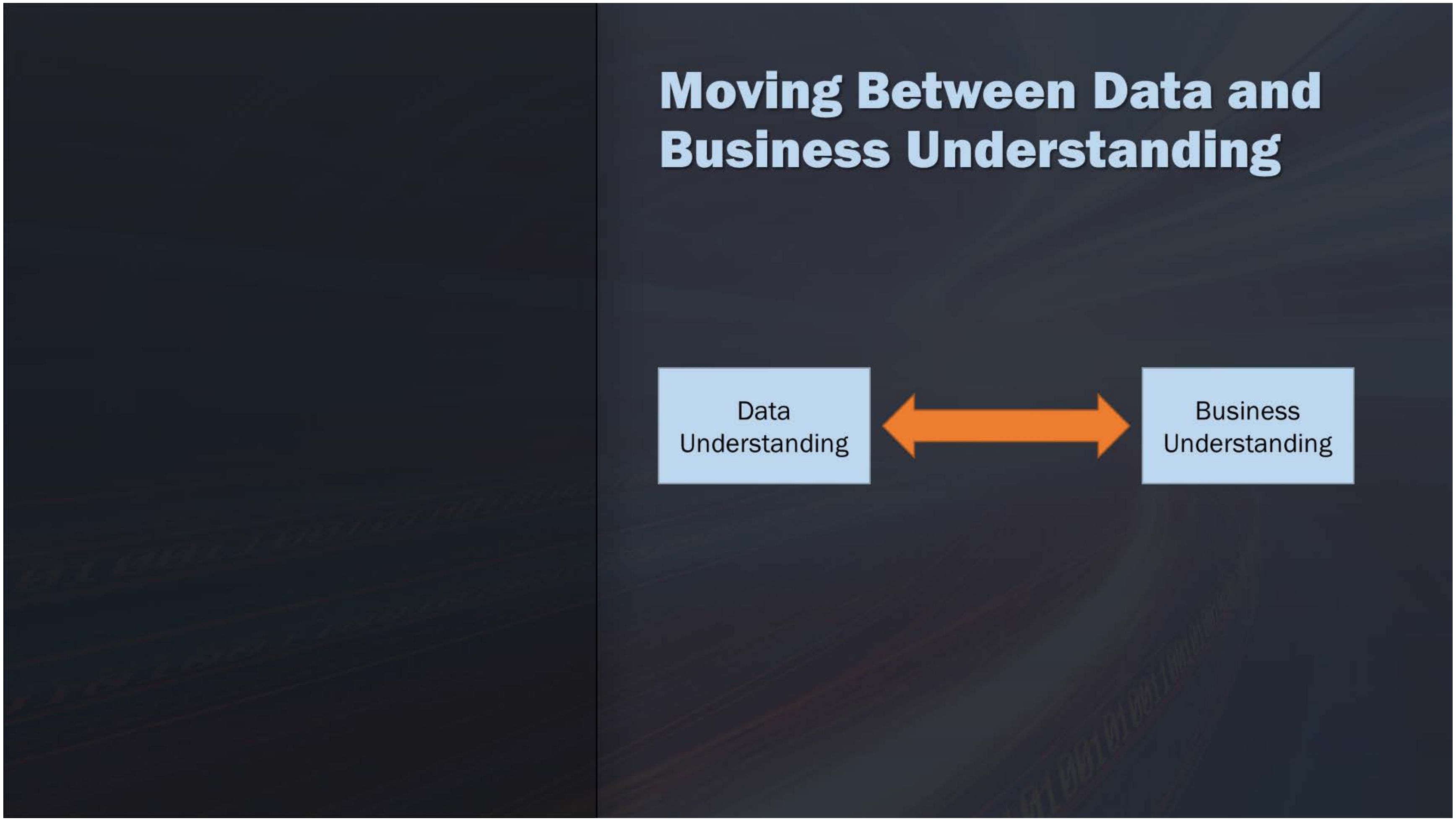
Which customers?

What product?

What is/should be excluded?

What is/should be counted from past?

Slide 8: Moving Between Data and Business Understanding

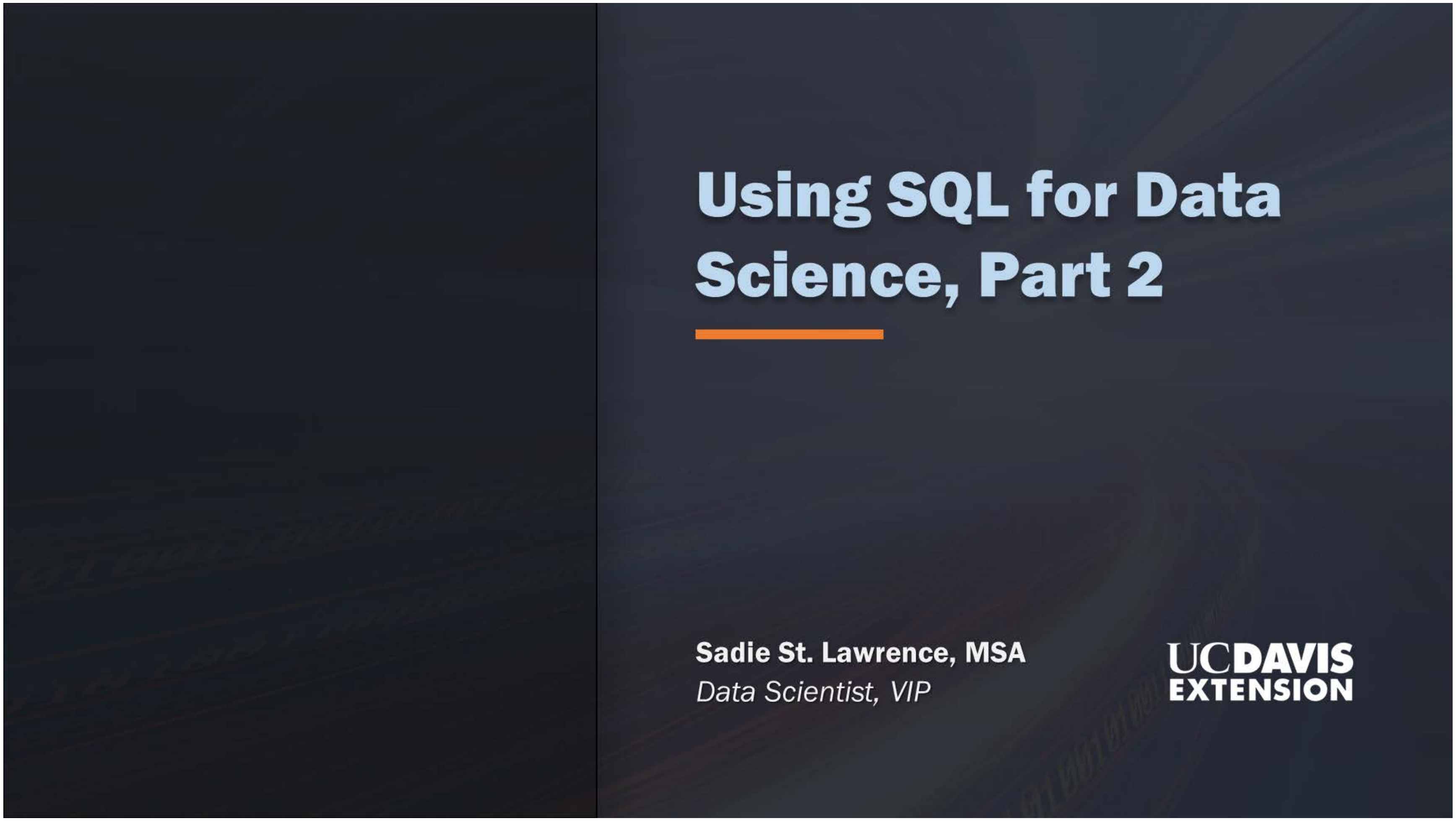


Moving Between Data and Business Understanding

Data Understanding

Business Understanding

Slide 1: Using SQL for Data Science, Part 2



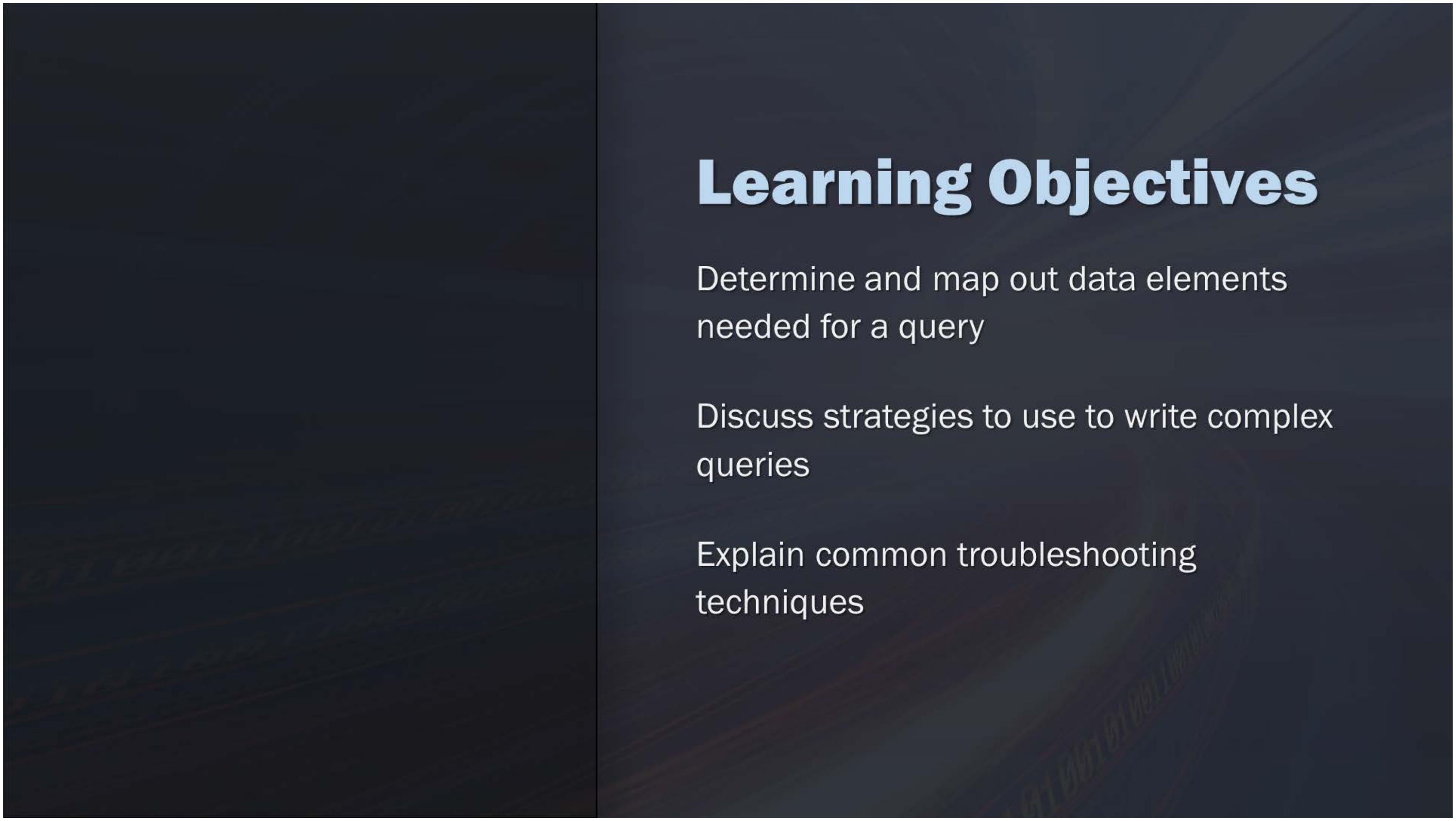
Using SQL for Data Science, Part 2

Sadie St. Lawrence, MSA

Data Scientist, VIP

UCDAVIS EXTENSION

Slide 2: Learning Objectives



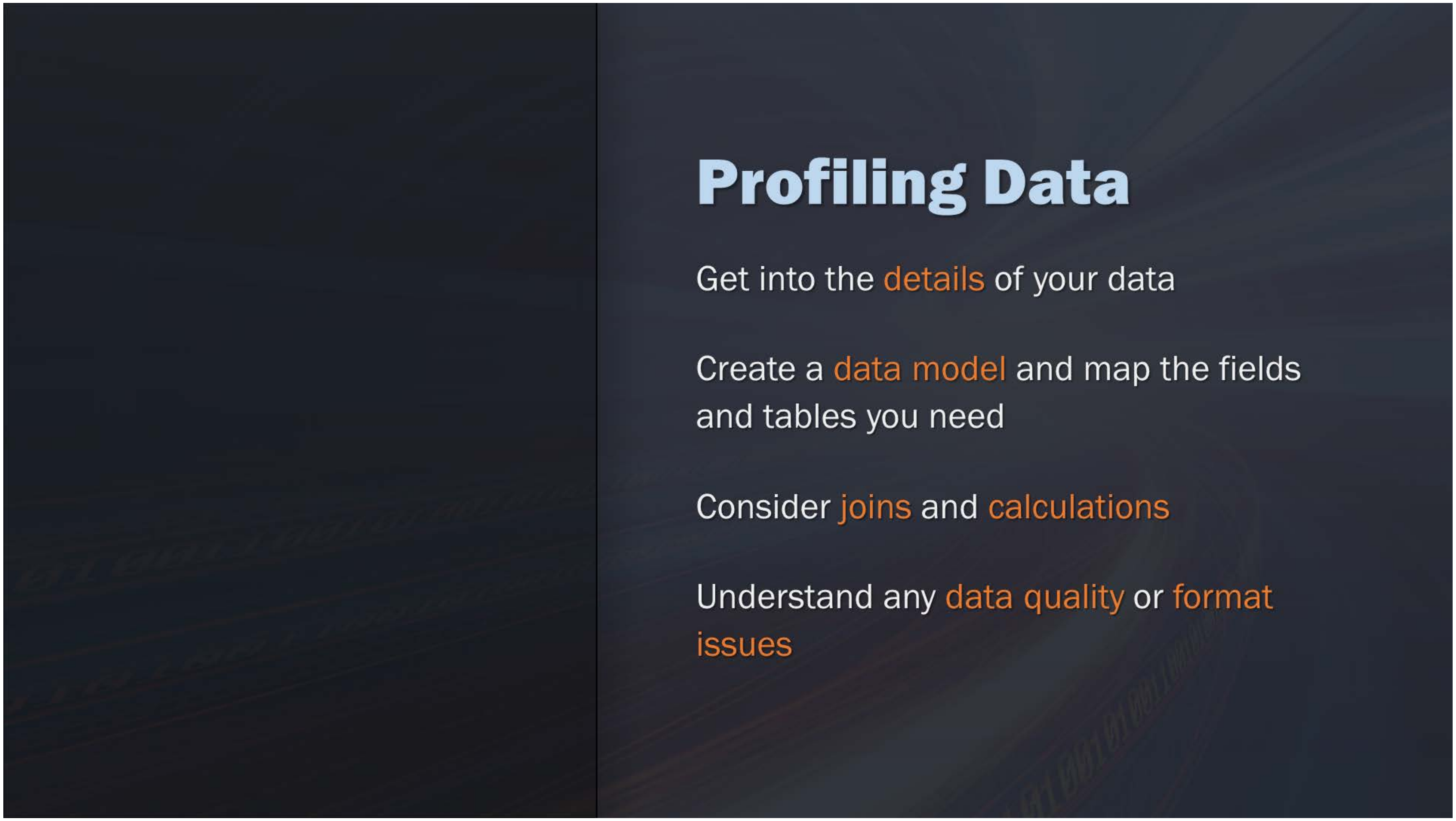
Learning Objectives

Determine and map out data elements needed for a query

Discuss strategies to use to write complex queries

Explain common troubleshooting techniques

Slide 3: Profiling Data



Profiling Data

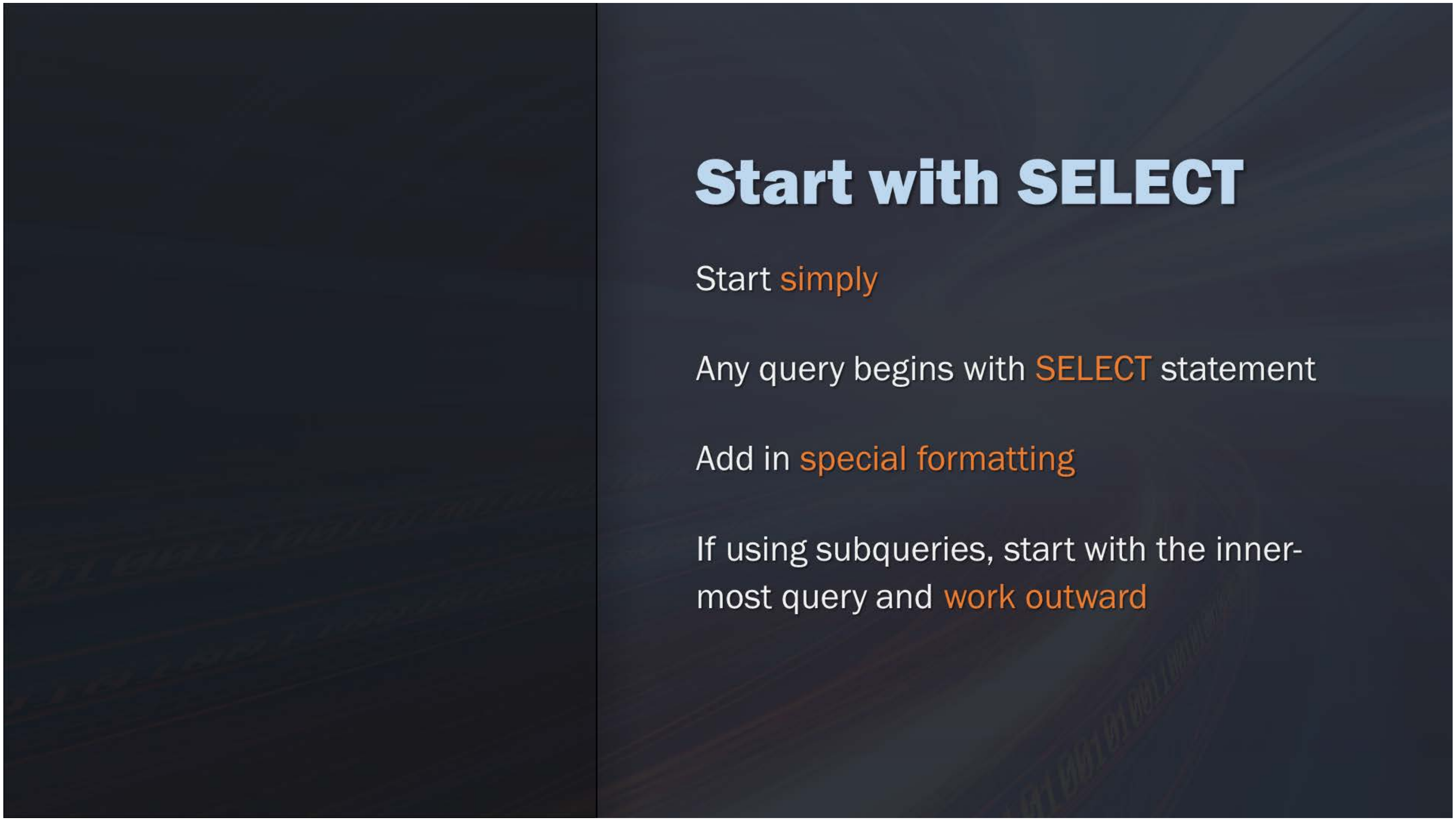
Get into the details of your data

Create a data model and map the fields and tables you need

Consider joins and calculations

Understand any data quality or format issues

Slide 4: Start with SELECT



Start with SELECT

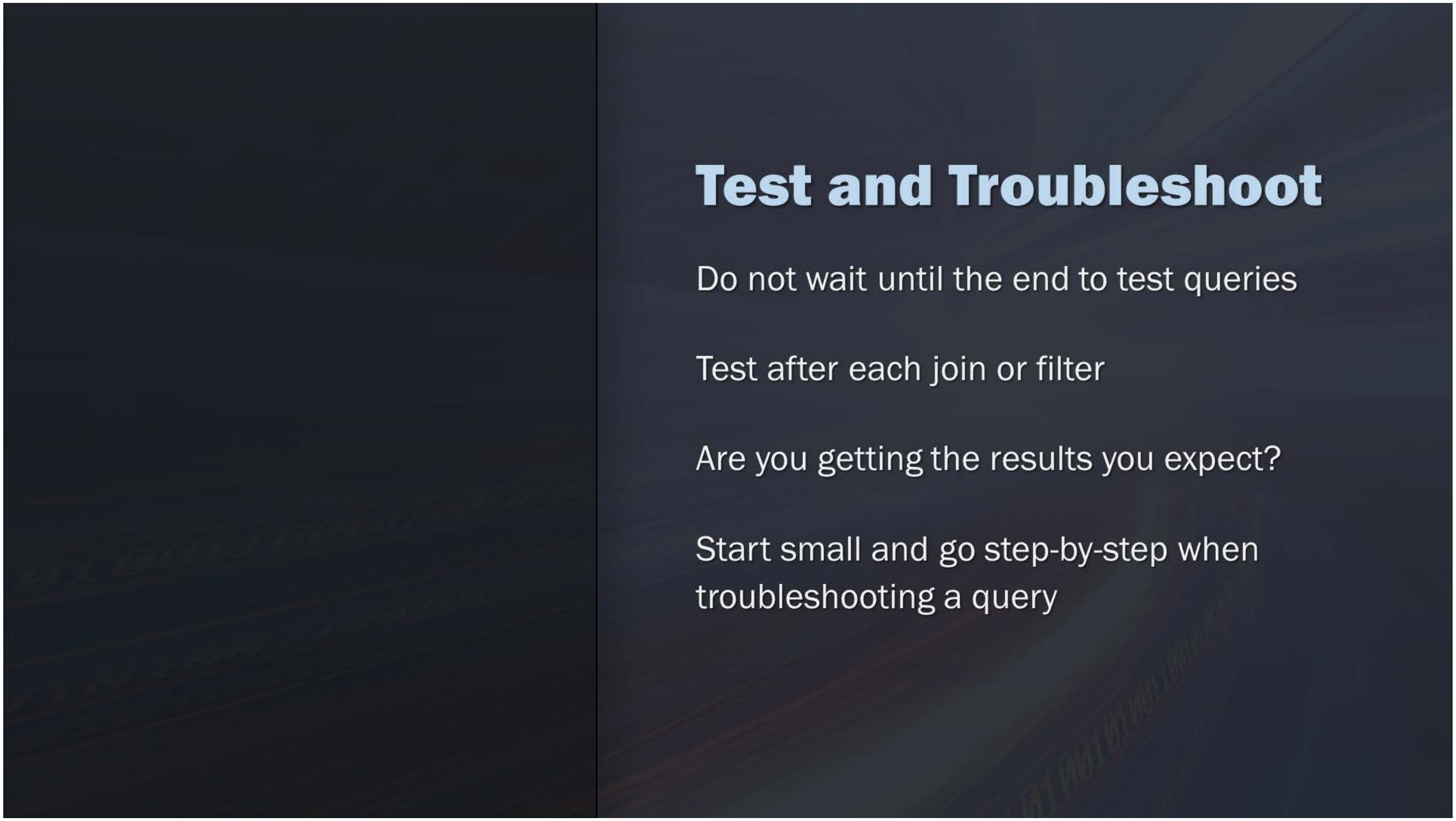
Start simply

Any query begins with SELECT statement

Add in special formatting

If using subqueries, start with the inner-most query and work outward

Slide 5: Test and Troubleshoot



Test and Troubleshoot

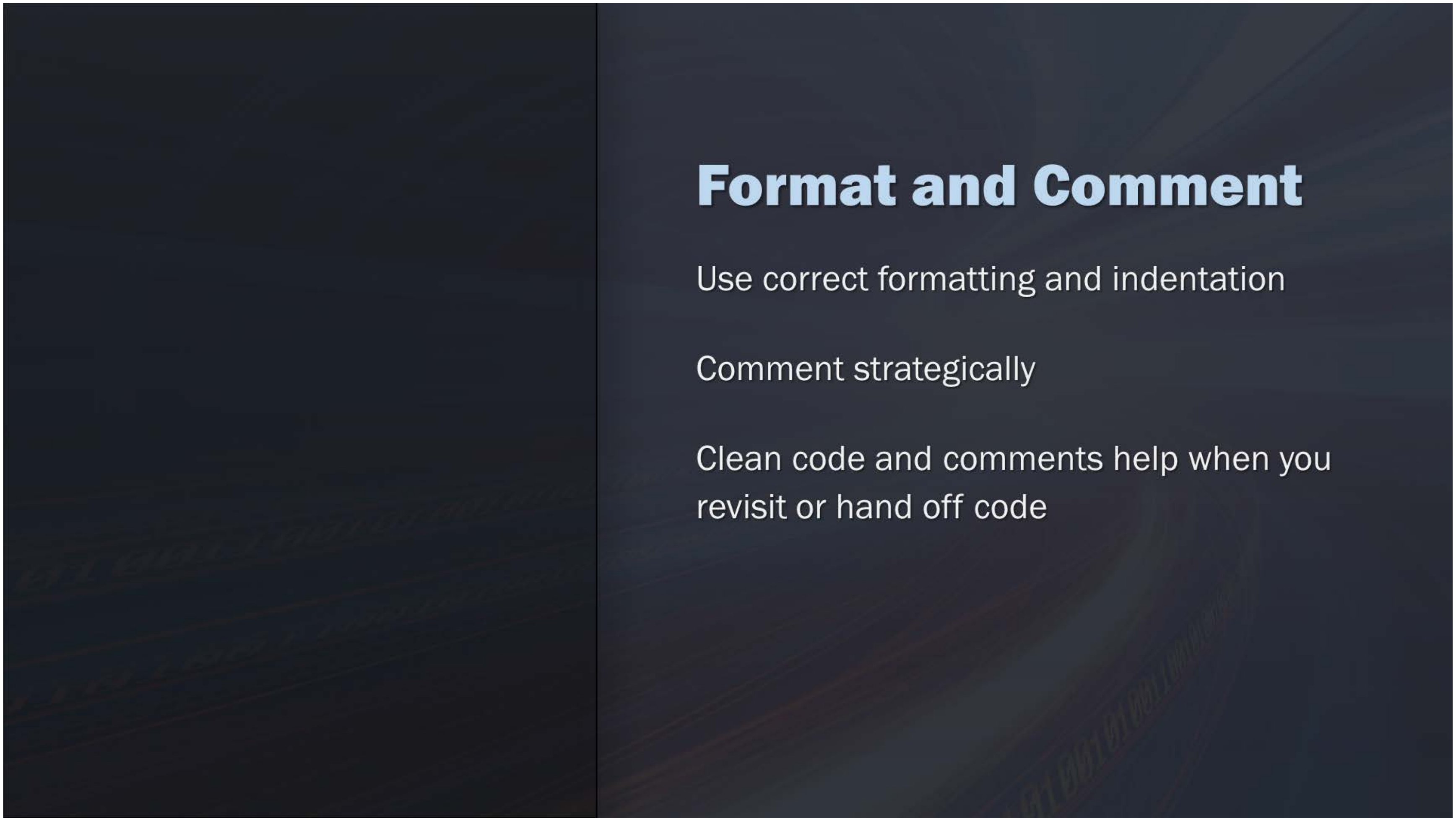
Do not wait until the end to test queries

Test after each join or filter

Are you getting the results you expect?

Start small and go step-by-step when troubleshooting a query

Slide 6: Format and Comment



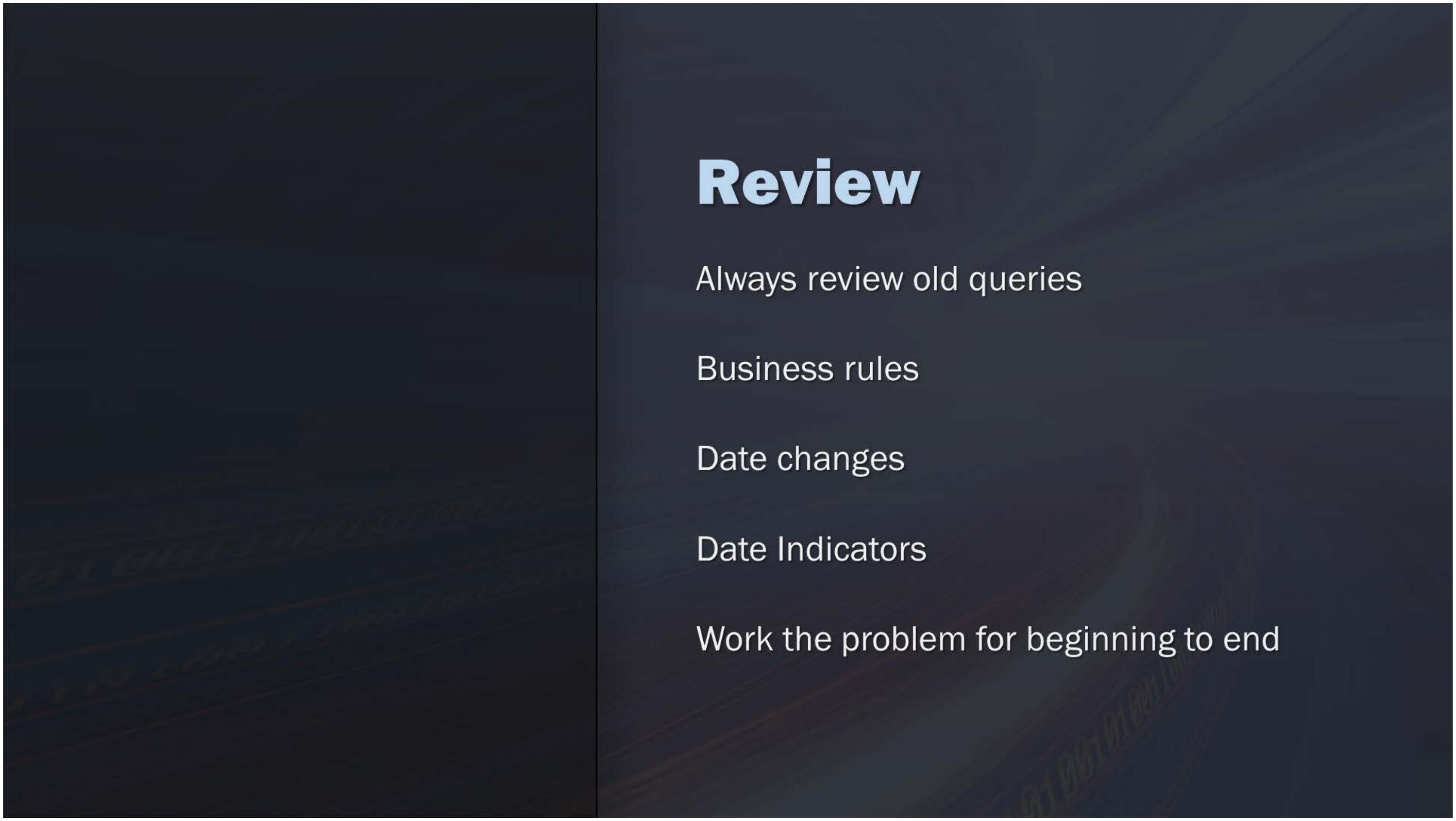
Format and Comment

Use correct formatting and indentation

Comment strategically

Clean code and comments help when you revisit or hand off code

Slide 7: Review



Review

Always review old queries

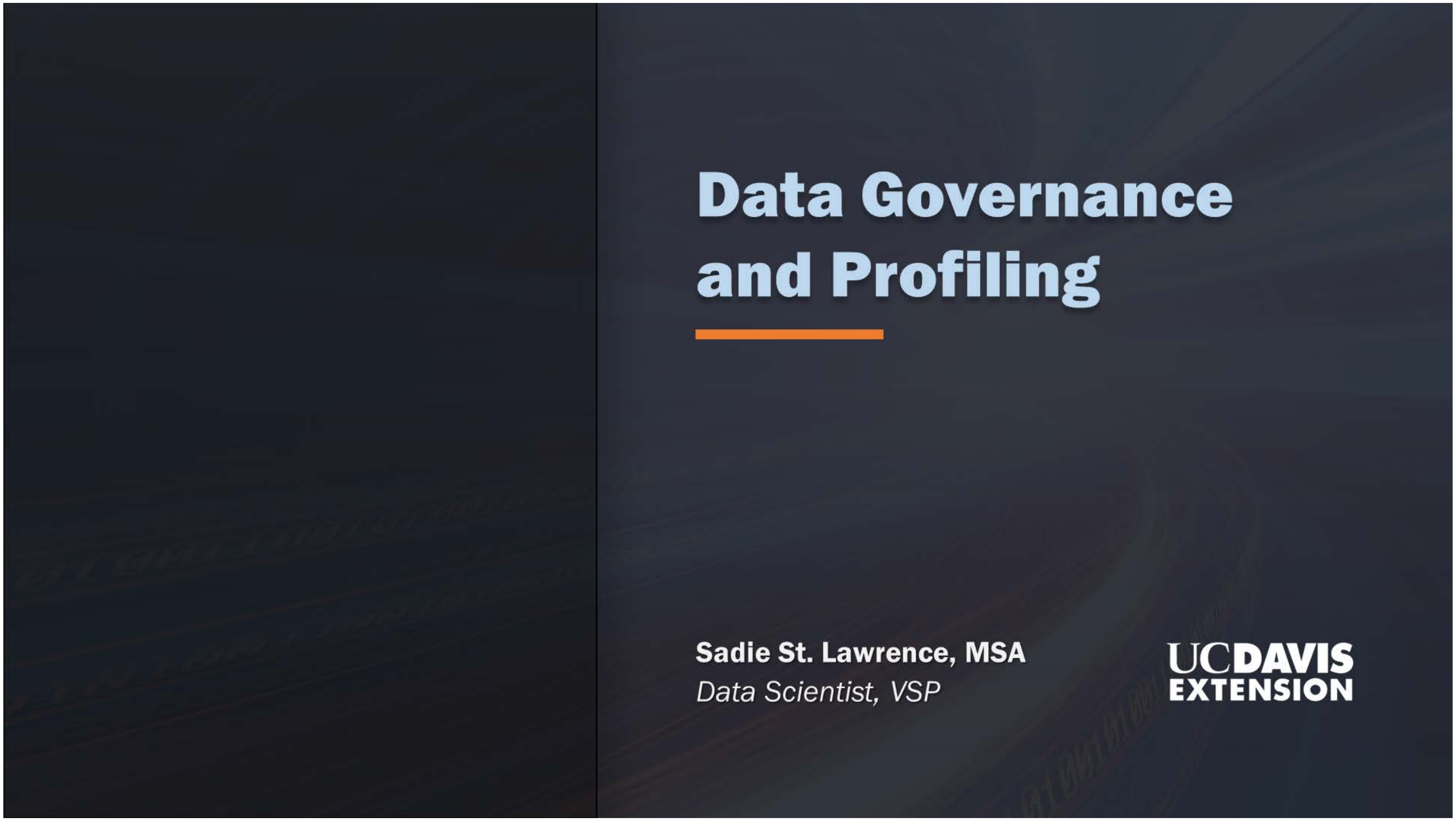
Business rules

Date changes

Date Indicators

Work the problem for beginning to end

Slide 1: Data Governance and Profiling



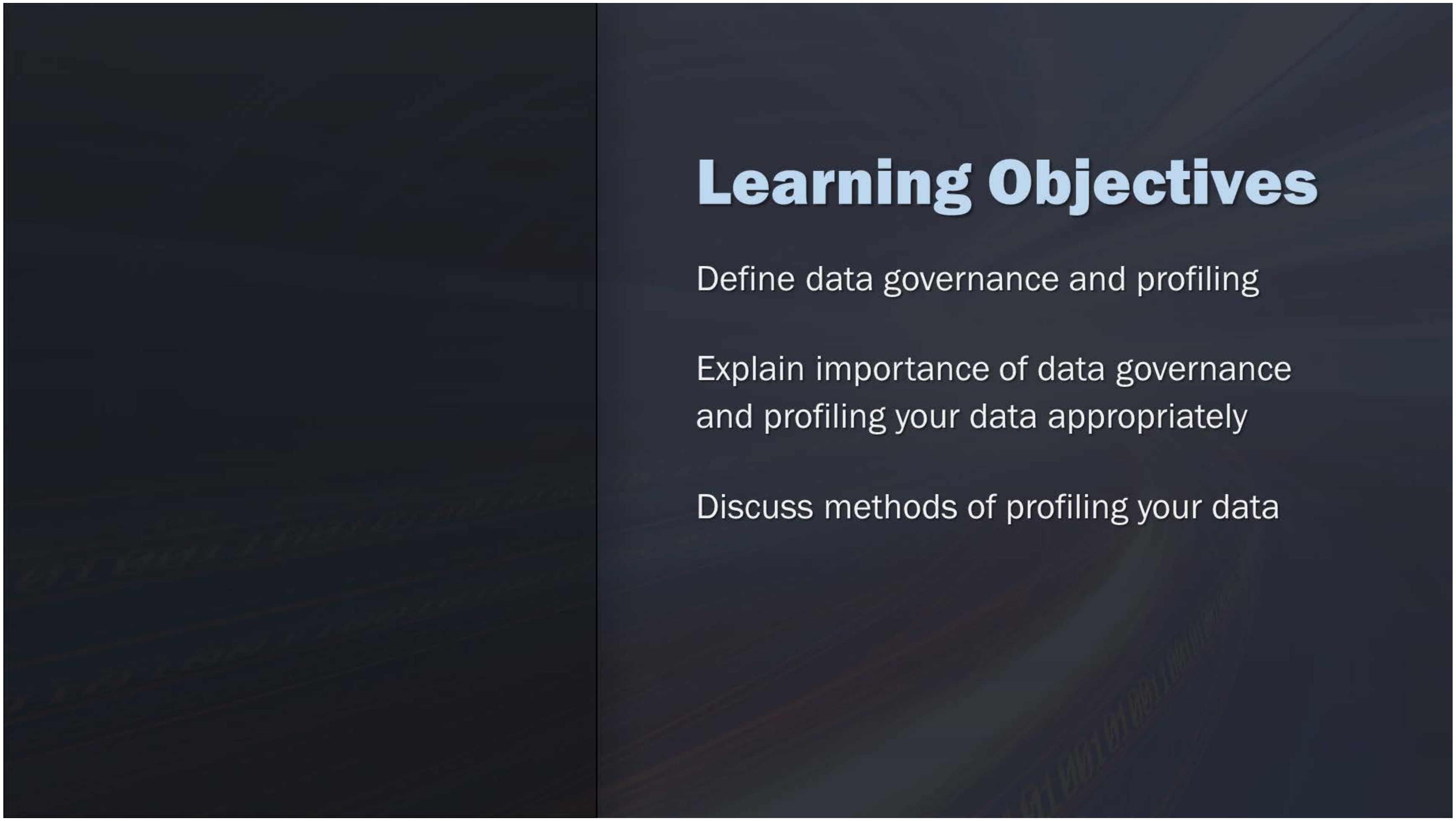
Data Governance and Profiling

Sadie St. Lawrence, MSA

Data Scientist, VSP

UC DAVIS EXTENSION

Slide 2: Learning Objectives



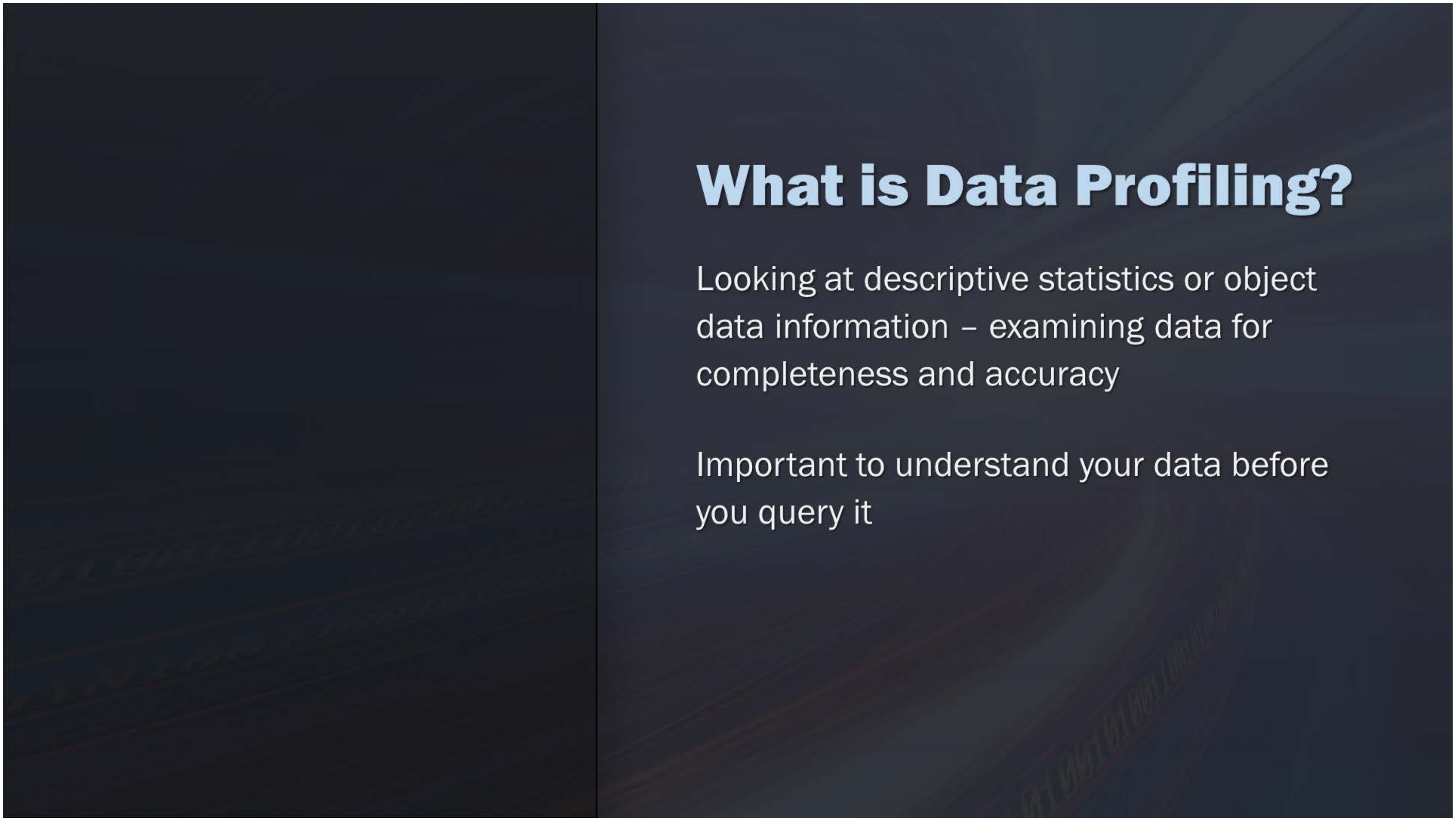
Learning Objectives

Define data governance and profiling

Explain importance of data governance and profiling your data appropriately

Discuss methods of profiling your data

Slide 3: What is Data Profiling

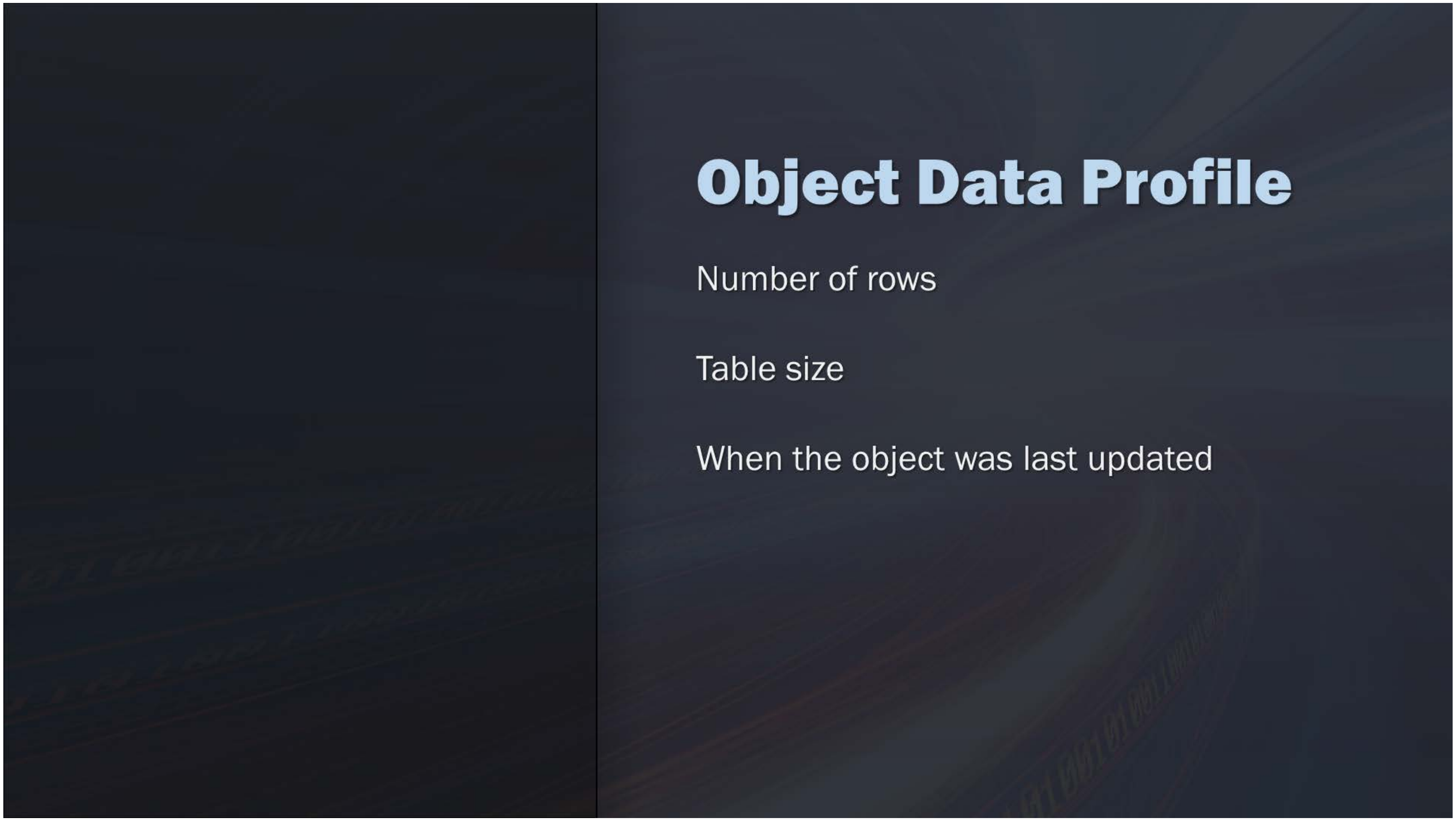


What is Data Profiling?

Looking at descriptive statistics or object data information – examining data for completeness and accuracy

Important to understand your data before you query it

Slide 4: Object Data Profile



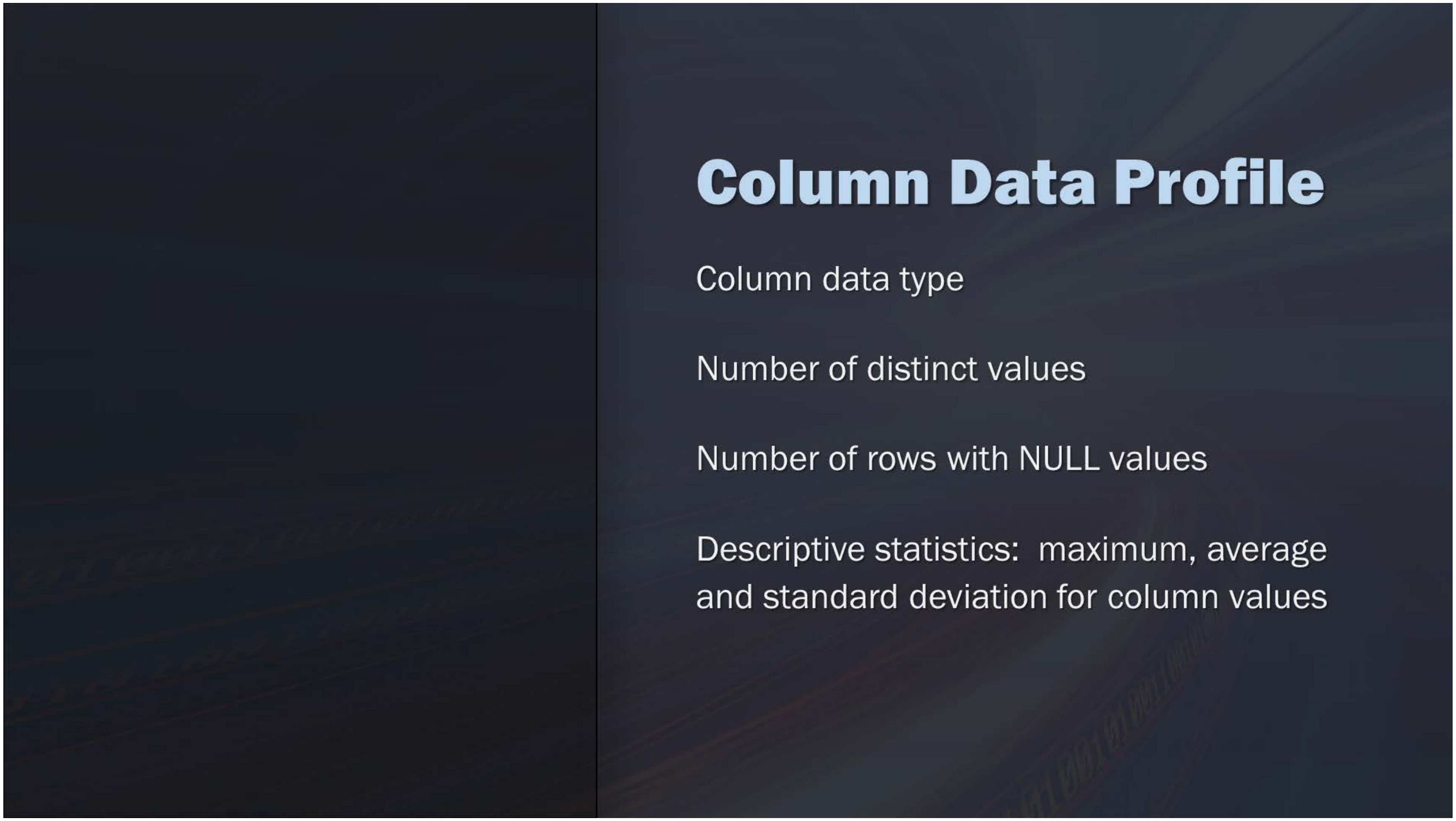
Object Data Profile

Number of rows

Table size

When the object was last updated

Slide 5: Column Data Profile



Column Data Profile

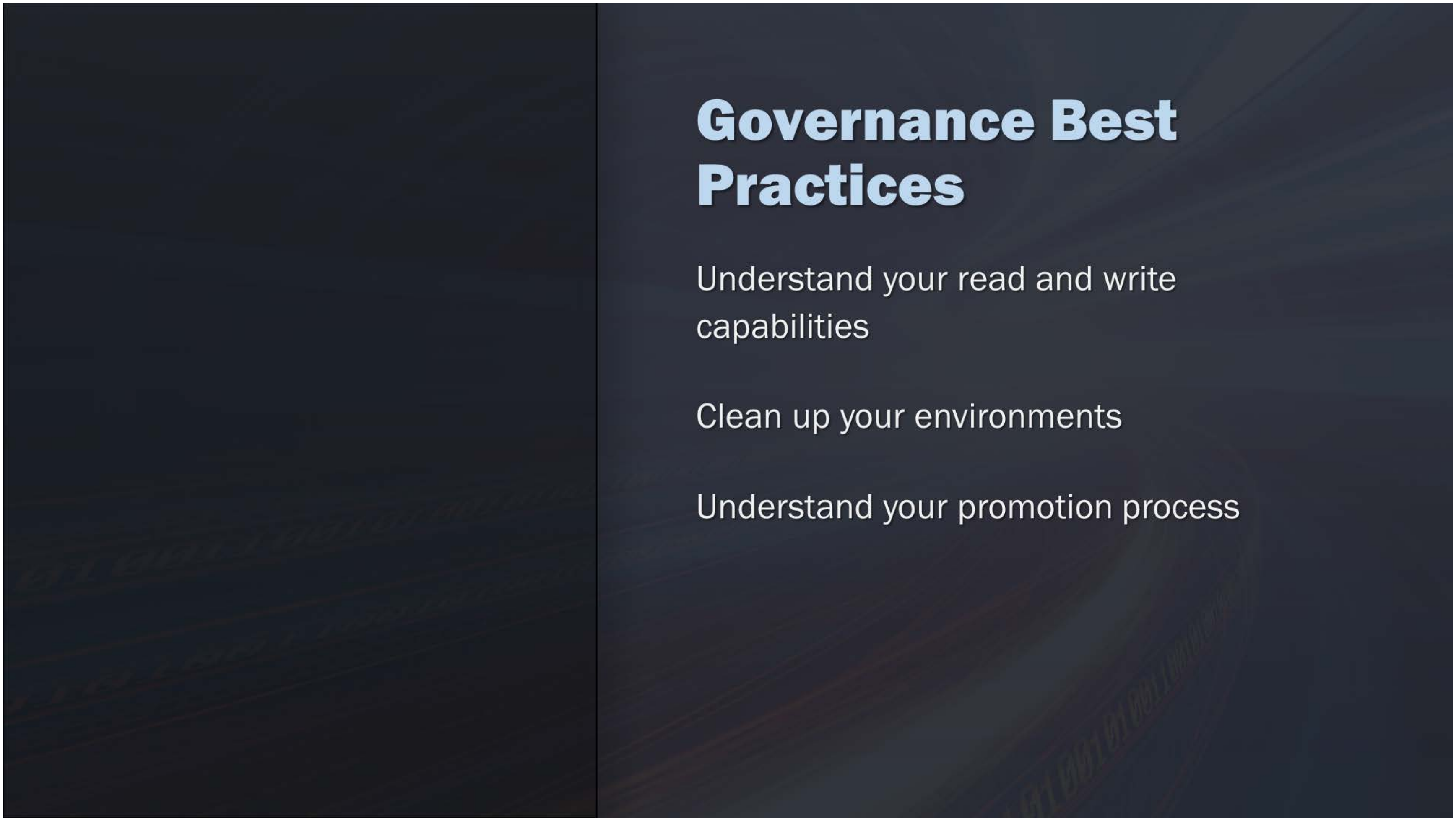
Column data type

Number of distinct values

Number of rows with NULL values

Descriptive statistics: maximum, average and standard deviation for column values

Slide 6: Governance Best Practices



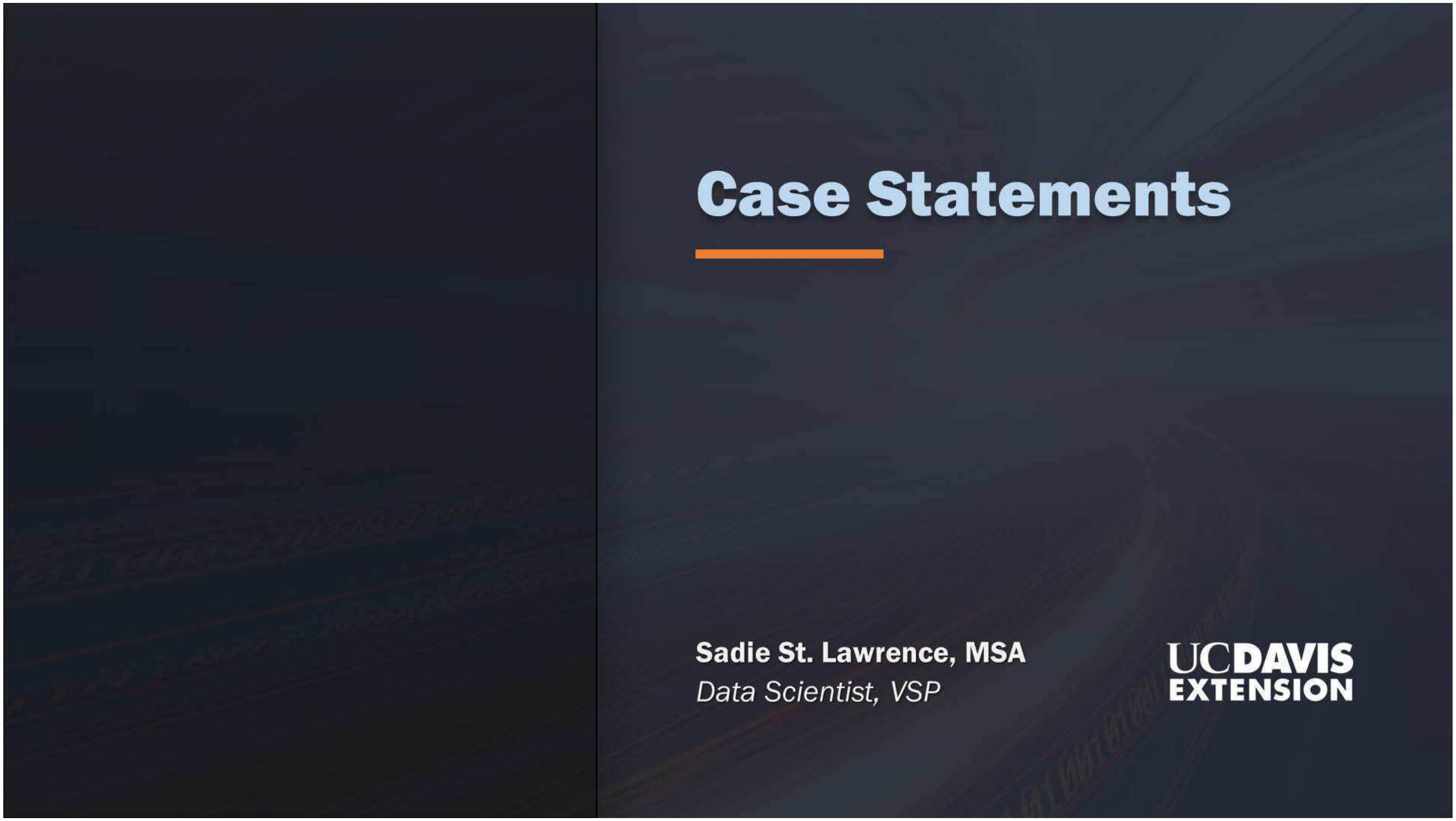
Governance Best Practices

Understand your read and write capabilities

Clean up your environments

Understand your promotion process

Slide 1: Case Statements

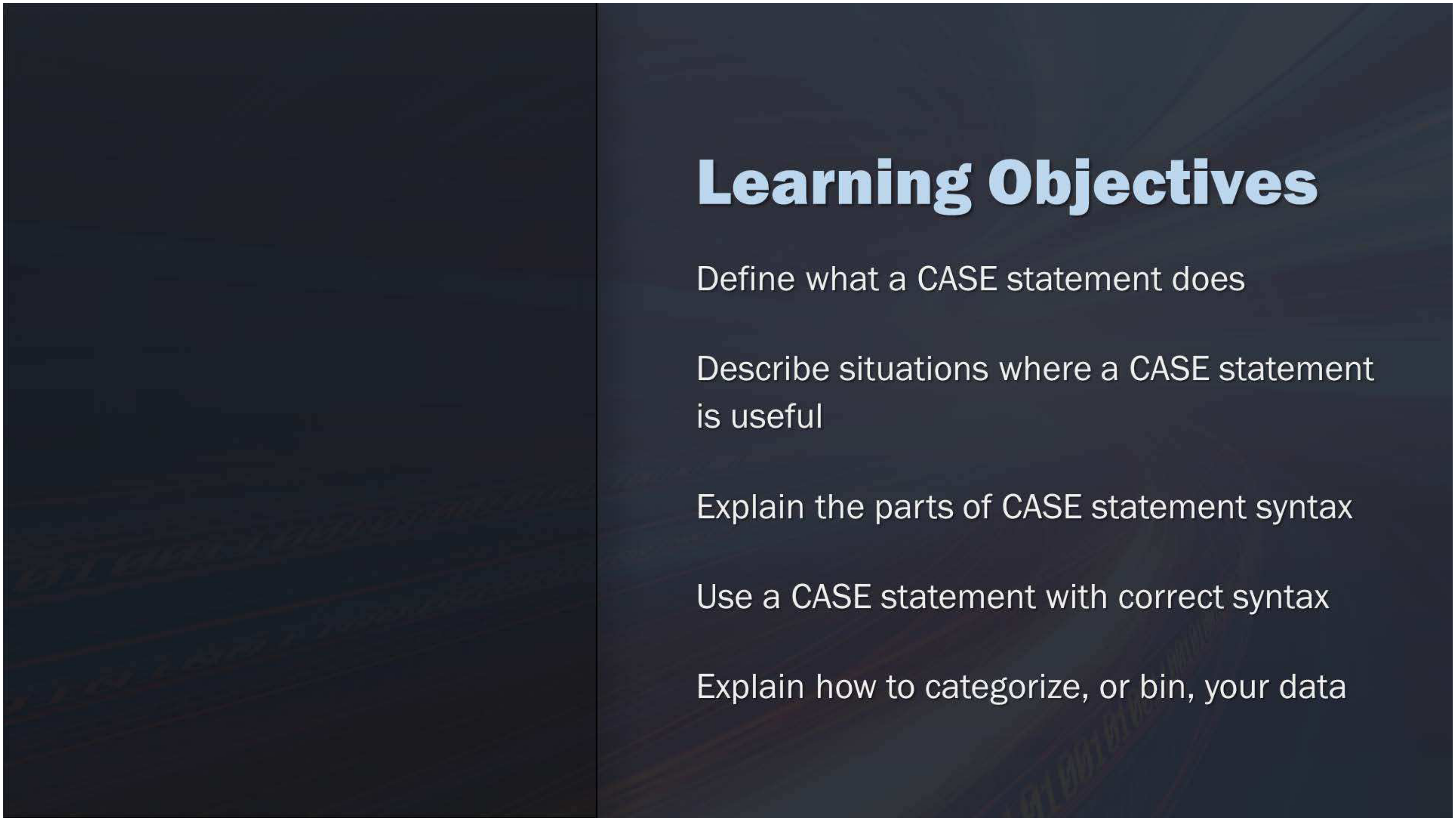


Case Statements

Sadie St. Lawrence, MSA
Data Scientist, VSP

UCDAVIS
EXTENSION

Slide 2: Learning Objectives



Learning Objectives

Define what a CASE statement does

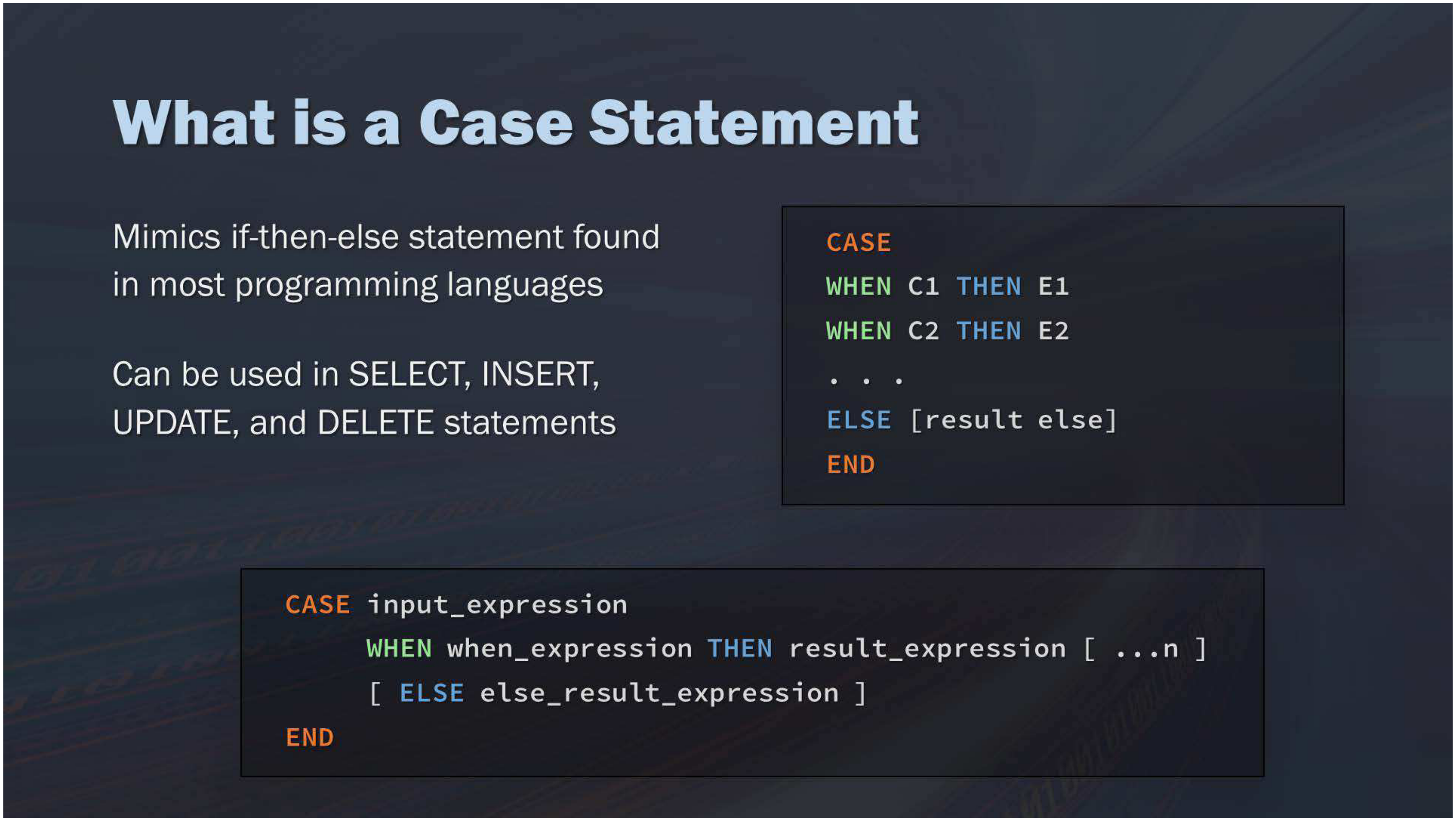
Describe situations where a CASE statement is useful

Explain the parts of CASE statement syntax

Use a CASE statement with correct syntax

Explain how to categorize, or bin, your data

Slide 3: What is a Case Statement



What is a Case Statement

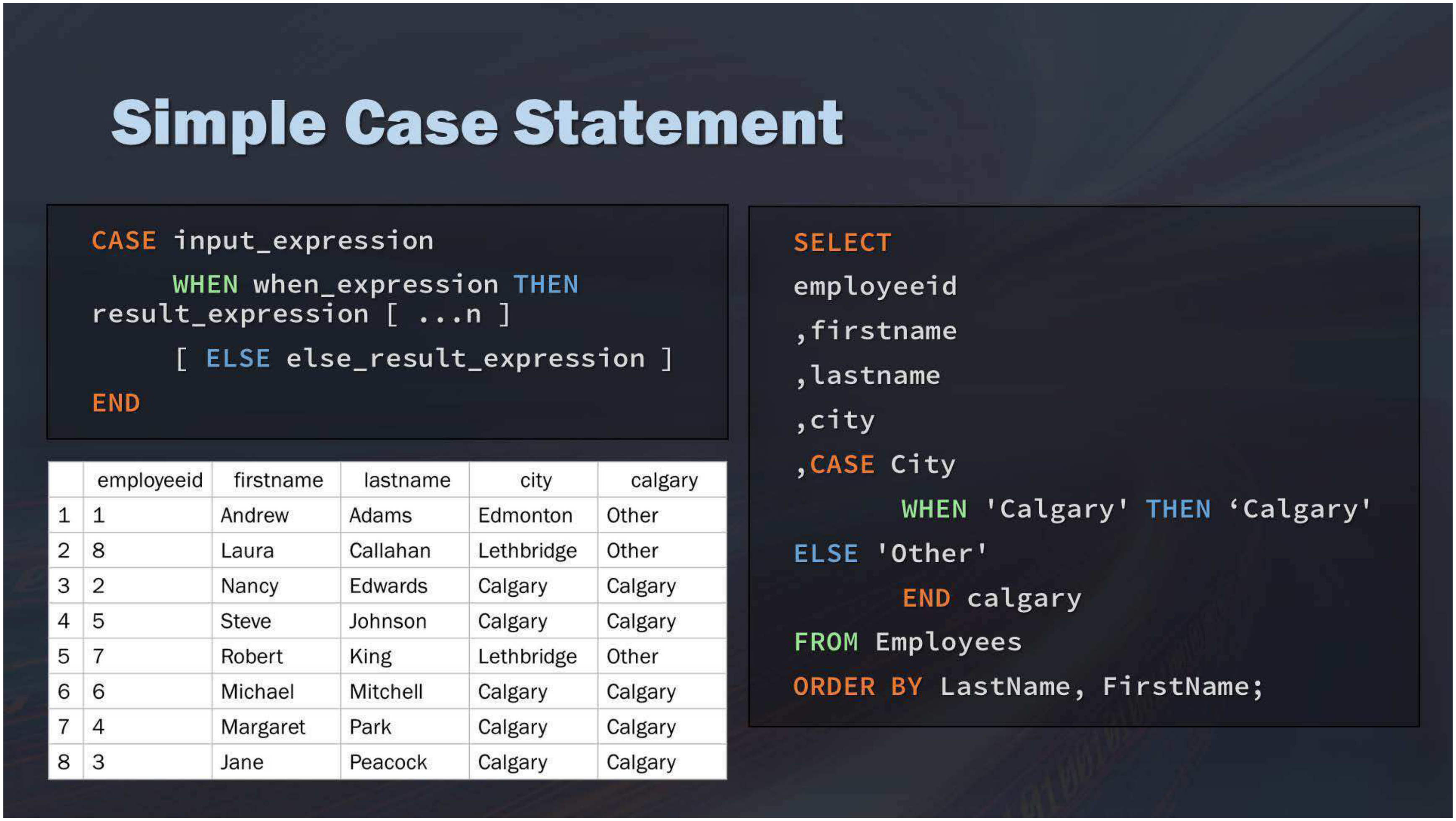
Mimics if-then-else statement found in most programming languages

Can be used in SELECT, INSERT, UPDATE, and DELETE statements

```
CASE
WHEN C1 THEN E1
WHEN C2 THEN E2
. . .
ELSE [result else]
END
```

```
CASE input_expression
     WHEN when_expression THEN result_expression [ ...n ]
     [ ELSE else_result_expression ]
END
```

Slide 4: Simple Case Statement



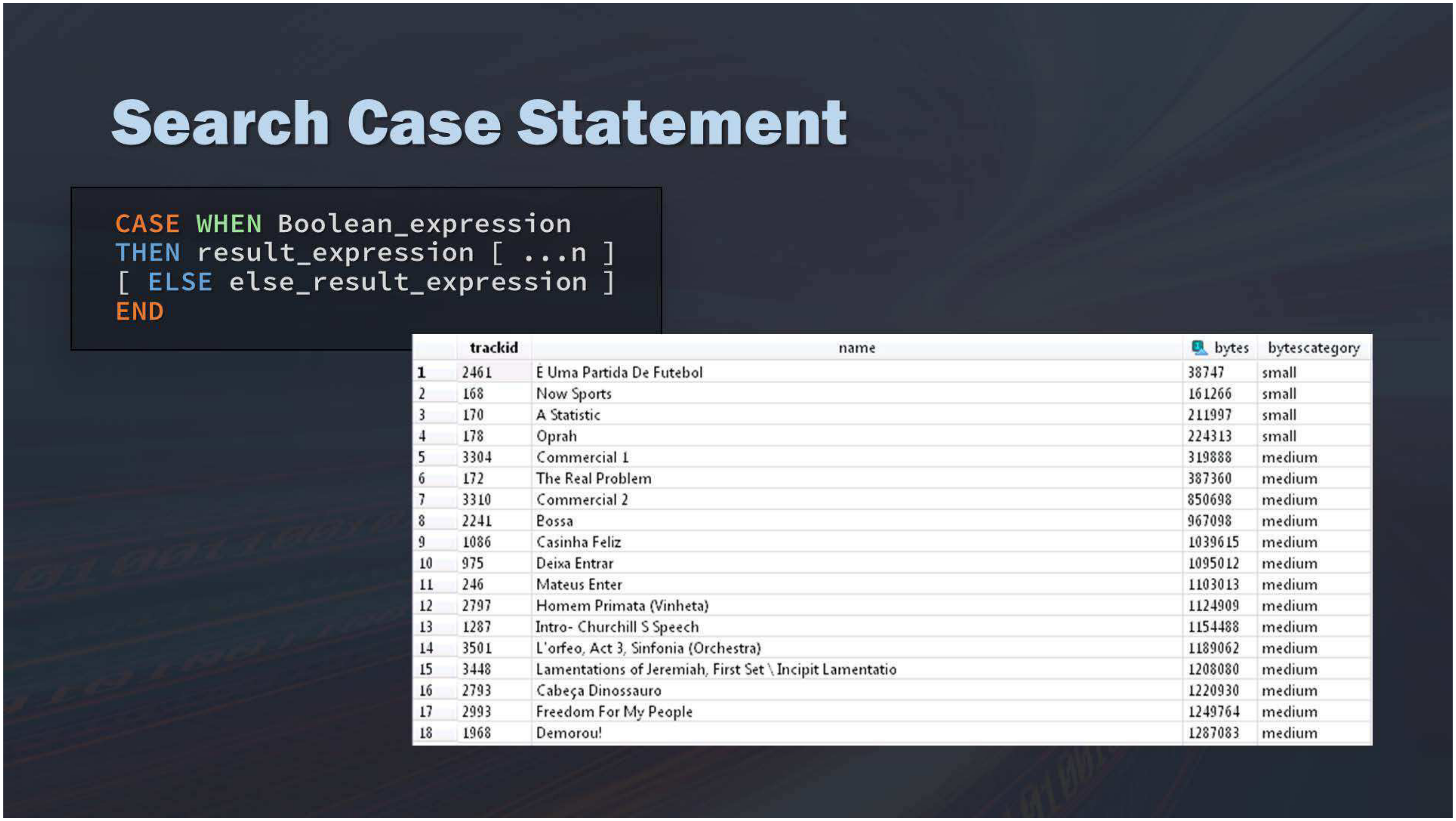
Simple Case Statement

```
CASE input_expression
     WHEN when_expression THEN
result_expression [ ...n ]
     [ ELSE else_result_expression ]
END
```

	employeeid	firstname	lastname	city	calgary
1	1	Andrew	Adams	Edmonton	Other
2	8	Laura	Callahan	Lethbridge	Other
3	2	Nancy	Edwards	Calgary	Calgary
4	5	Steve	Johnson	Calgary	Calgary
5	7	Robert	King	Lethbridge	Other
6	6	Michael	Mitchell	Calgary	Calgary
7	4	Margaret	Park	Calgary	Calgary
8	3	Jane	Peacock	Calgary	Calgary

```
SELECT
employeeid
,firstname
,lastname
,city
,CASE City
      WHEN 'Calgary' THEN ‘Calgary'
ELSE 'Other'
      END calgary
FROM Employees
ORDER BY LastName, FirstName;
```

Slide 5: Search Case Statement

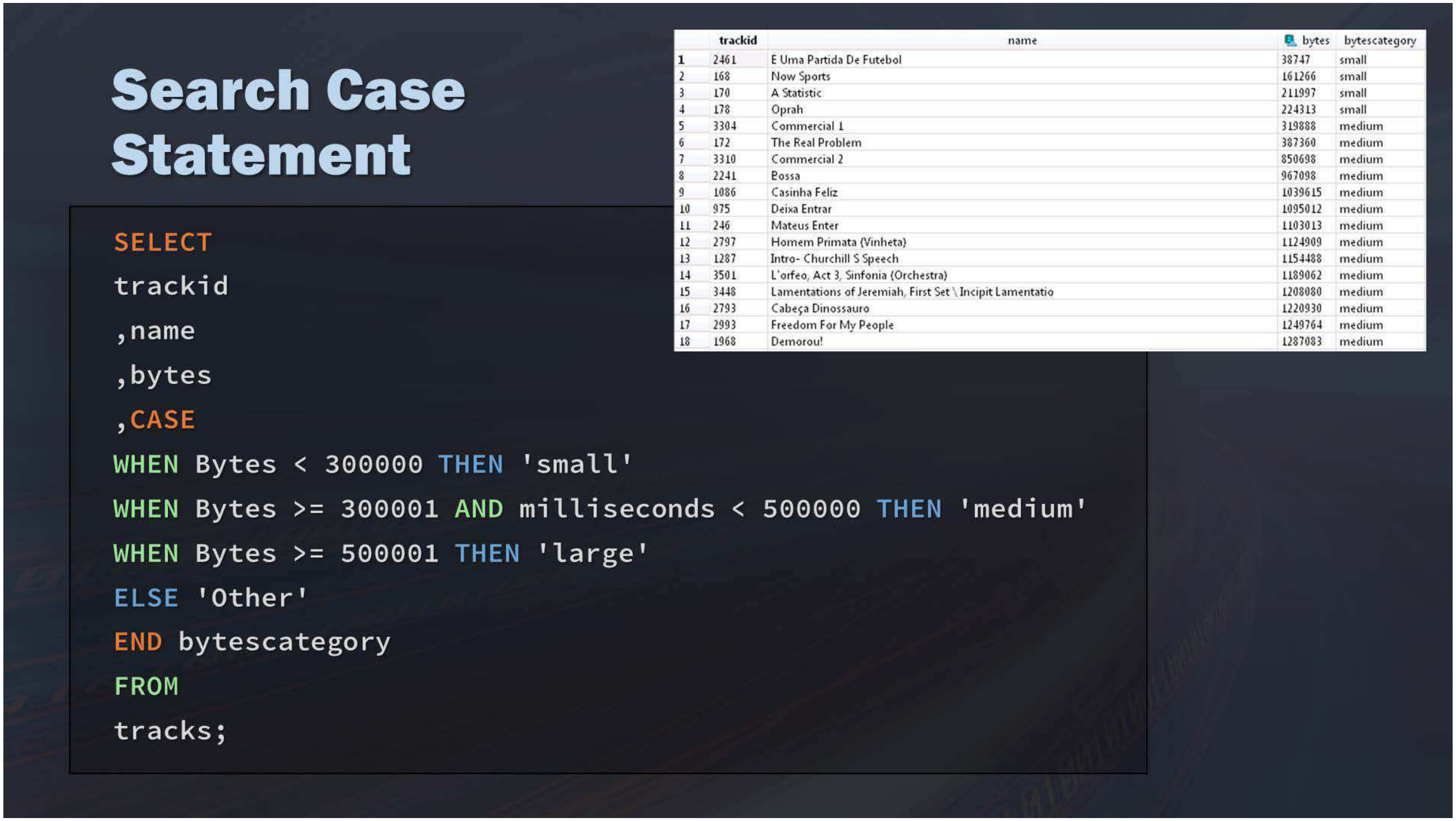


Search Case Statement

```
CASE WHEN Boolean_expression
THEN result_expression [ ...n ]
[ ELSE else_result_expression ]
END
```

	trackid	name	bytes	bytescategory
1	2461	E Uma Partida De Futebol	38747	small
2	168	Now Sports	161266	small
3	170	A Statistic	211997	small
4	178	Oprah	224313	small
5	3304	Commercial 1	319888	medium
6	172	The Real Problem	387360	medium
7	3310	Commercial 2	850698	medium
8	2241	Bossa	967098	medium
9	1086	Casinha Feliz	1039615	medium
10	975	Deixa Entrar	1095012	medium
11	246	Mateus Enter	1103013	medium
12	2797	Homem Primata (Vinheta)	1124909	medium
13	1287	Intro- Churchill S Speech	1154488	medium
14	3501	L'orfeo, Act 3, Sinfonia (Orchestra)	1189062	medium
15	3448	Lamentations of Jeremiah, First Set \ Incipit Lamentatio	1208080	medium
16	2793	Cabeça Dinossauro	1220930	medium
17	2993	Freedom For My People	1249764	medium
18	1968	Demorou!	1287083	medium

Slide 6: Search Case Statement

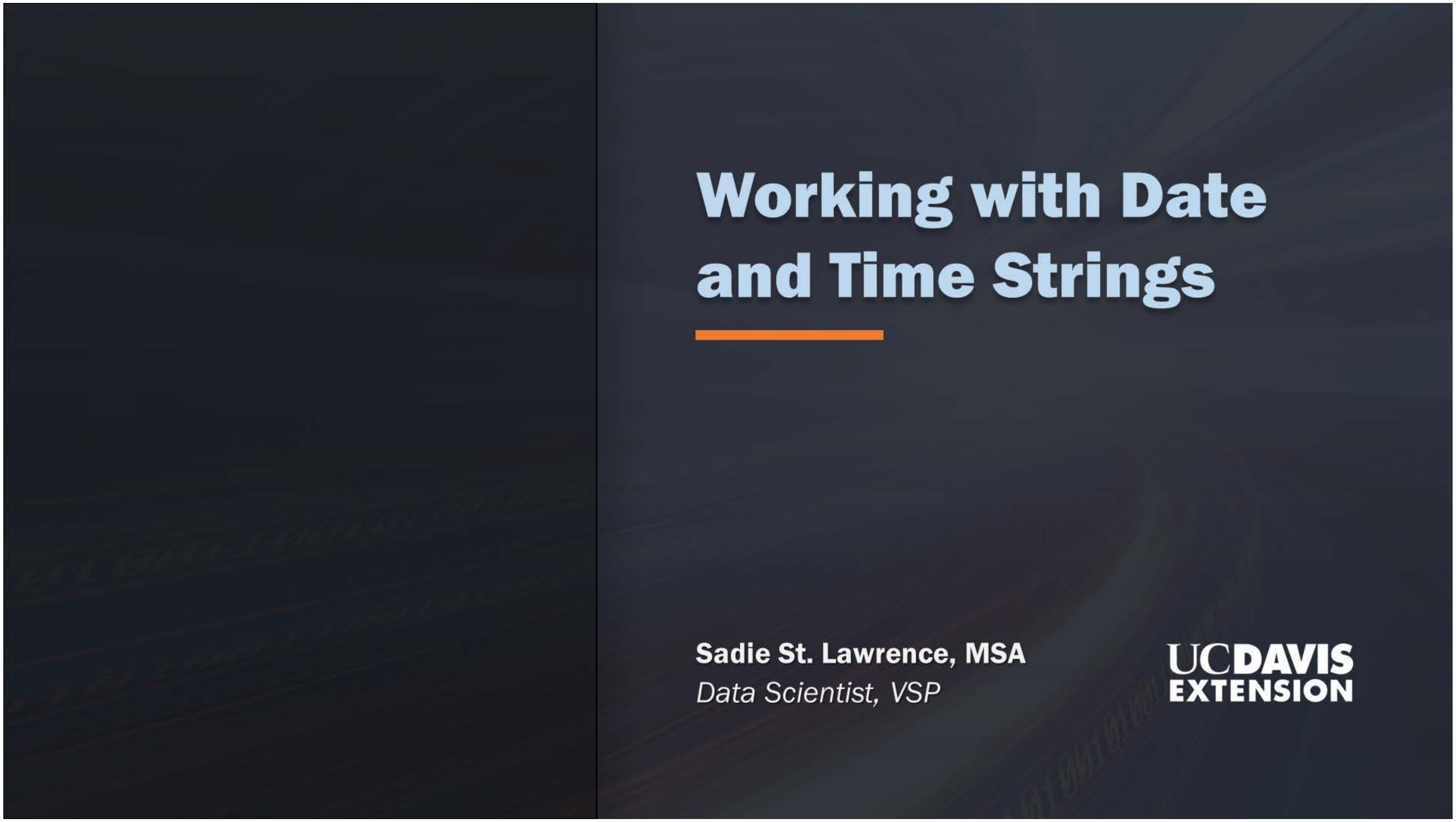


Search Case Statement

```
SELECT
trackid
,name
,bytes
,CASE
WHEN Bytes < 300000 THEN 'small'
WHEN Bytes >= 300001 AND milliseconds < 500000 THEN 'medium'
WHEN Bytes >= 500001 THEN 'large'
ELSE 'Other'
END bytescategory
FROM
tracks;
```

	trackid	name	bytes	bytescategory
1	2461	E Uma Partida De Futebol	38747	small
2	168	Now Sports	161266	small
3	170	A Statistic	211997	small
4	178	Oprah	224313	small
5	3304	Commercial 1	319888	medium
6	172	The Real Problem	387360	medium
7	3310	Commercial 2	850698	medium
8	2241	Bossa	967098	medium
9	1086	Casinha Feliz	1039615	medium
10	975	Deixa Entrar	1095012	medium
11	246	Mateus Enter	1103013	medium
12	2797	Homem Primata (Vinheta)	1124909	medium
13	1287	Intro- Churchill S Speech	1154488	medium
14	3501	L'orfeo, Act 3, Sinfonia (Orchestra)	1189062	medium
15	3448	Lamentations of Jeremiah, First Set \ Incipit Lamentatio	1208080	medium
16	2793	Cabeça Dinossauro	1220930	medium
17	2993	Freedom For My People	1249764	medium
18	1968	Demorou!	1287083	medium

Slide 1: Working with Date and Time Strings

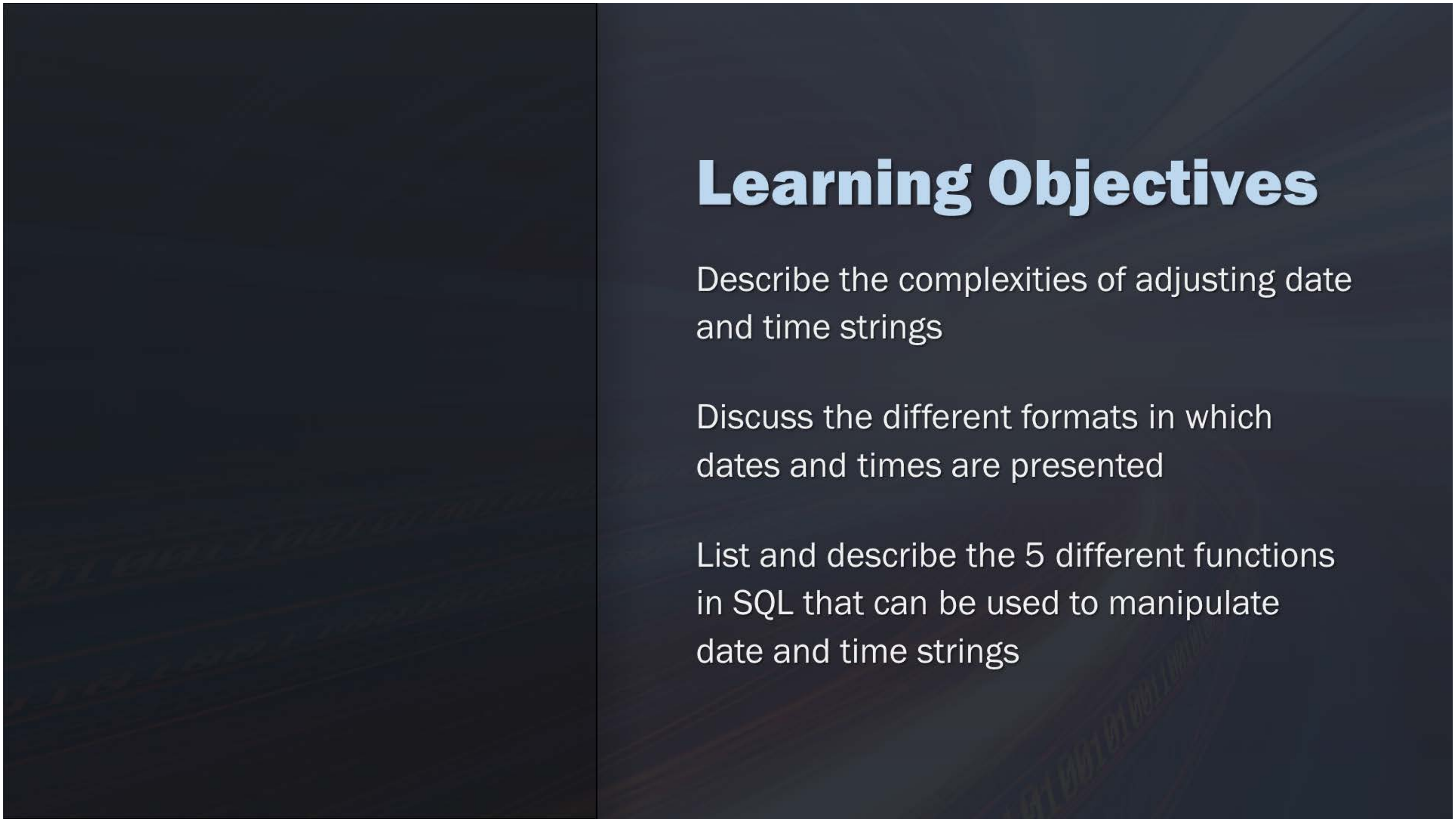


Working with Date and Time Strings

Sadie St. Lawrence, MSA
Data Scientist, VSP

UCDAVIS
EXTENSION

Slide 2: Learning Objectives



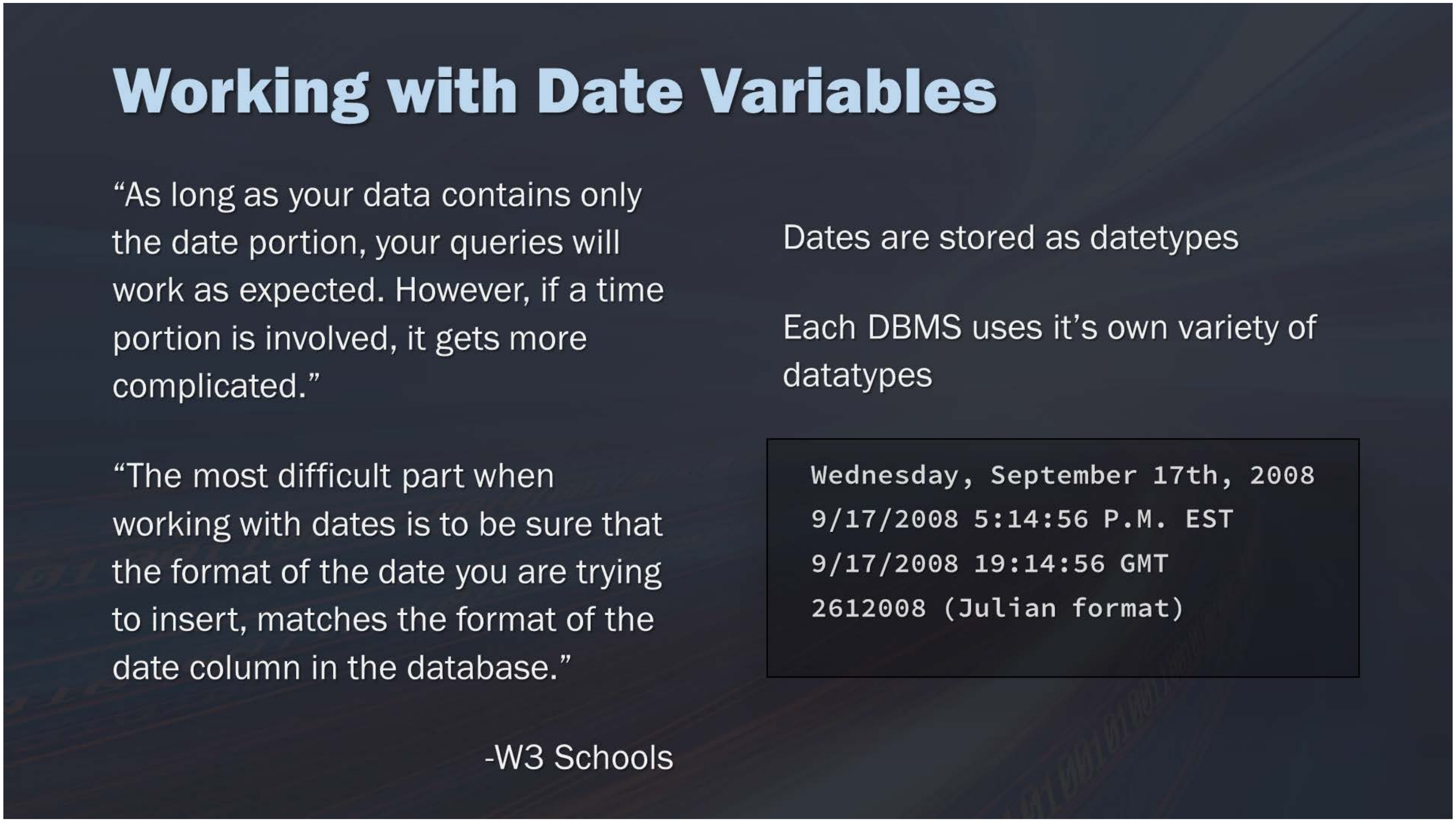
Learning Objectives

Describe the complexities of adjusting date and time strings

Discuss the different formats in which dates and times are presented

List and describe the 5 different functions in SQL that can be used to manipulate date and time strings

Slide 3: Working with Date Variables



Working with Date Variables

“As long as your data contains only the date portion, your queries will work as expected. However, if a time portion is involved, it gets more complicated.”

“The most difficult part when working with dates is to be sure that the format of the date you are trying to insert, matches the format of the date column in the database.”

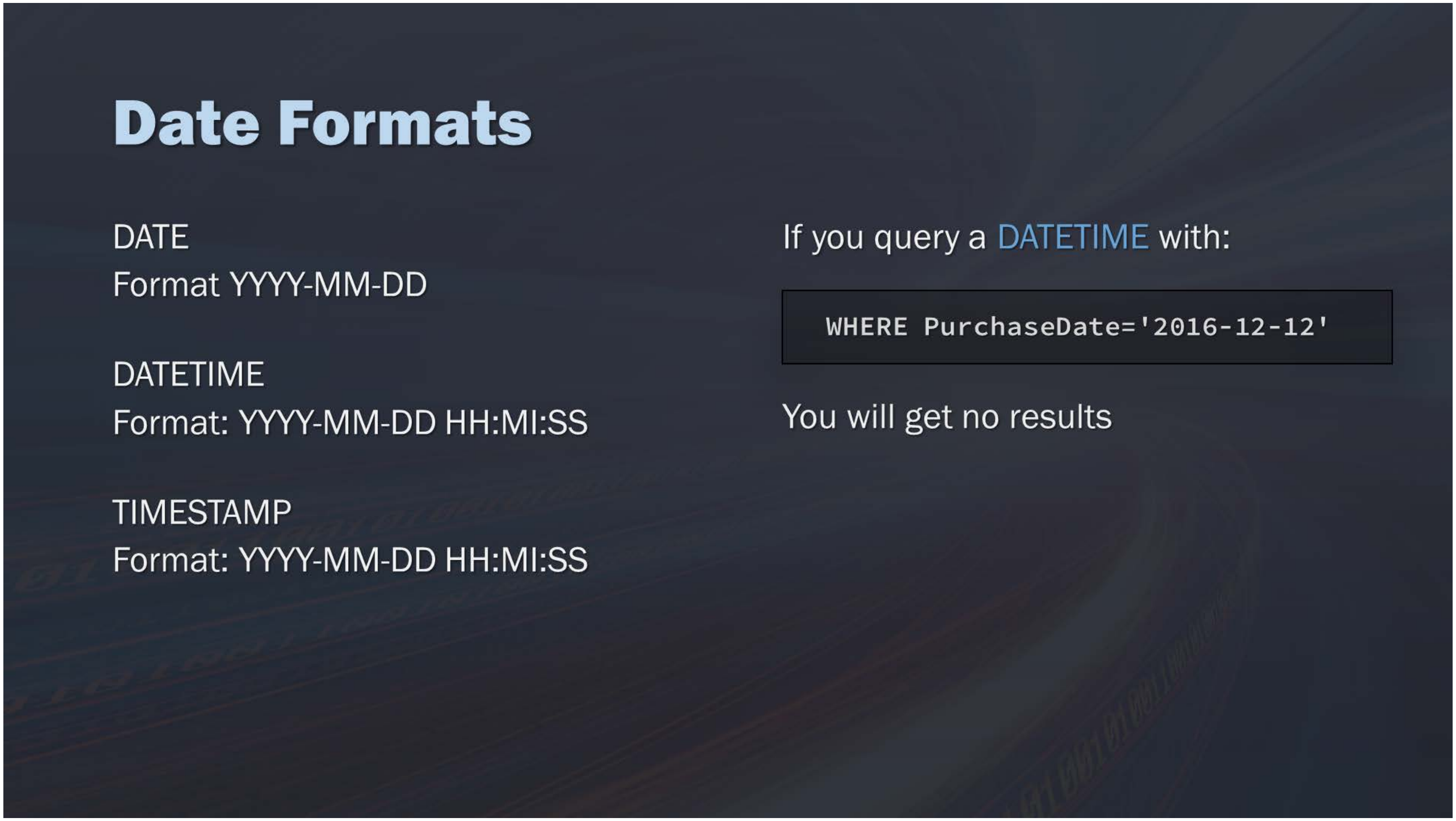
-W3 Schools

Dates are stored as datetypes

Each DBMS uses it’s own variety of datatypes

```
Wednesday, September 17th, 2008
9/17/2008 5:14:56 P.M. EST
9/17/2008 19:14:56 GMT
2612008 (Julian format)
```

Slide 4: Date Formats



Date Formats

DATE
Format YYYY-MM-DD

DATETIME
Format: YYYY-MM-DD HH:MI:SS

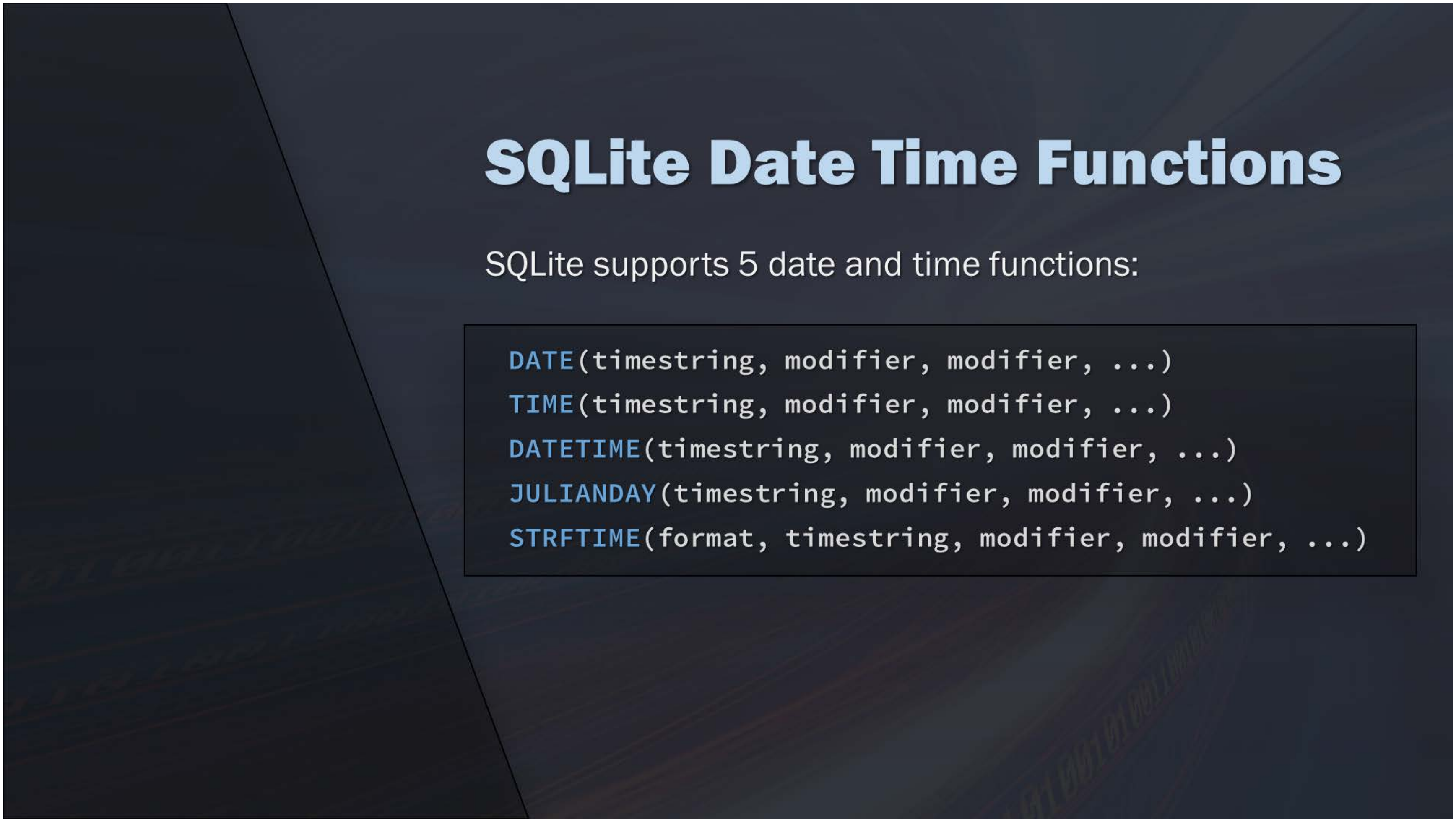
TIMESTAMP
Format: YYYY-MM-DD HH:MI:SS

If you query a DATETIME with:

```
WHERE PurchaseDate='2016-12-12'
```

You will get no results

Slide 5: SQLite Date Time Functions

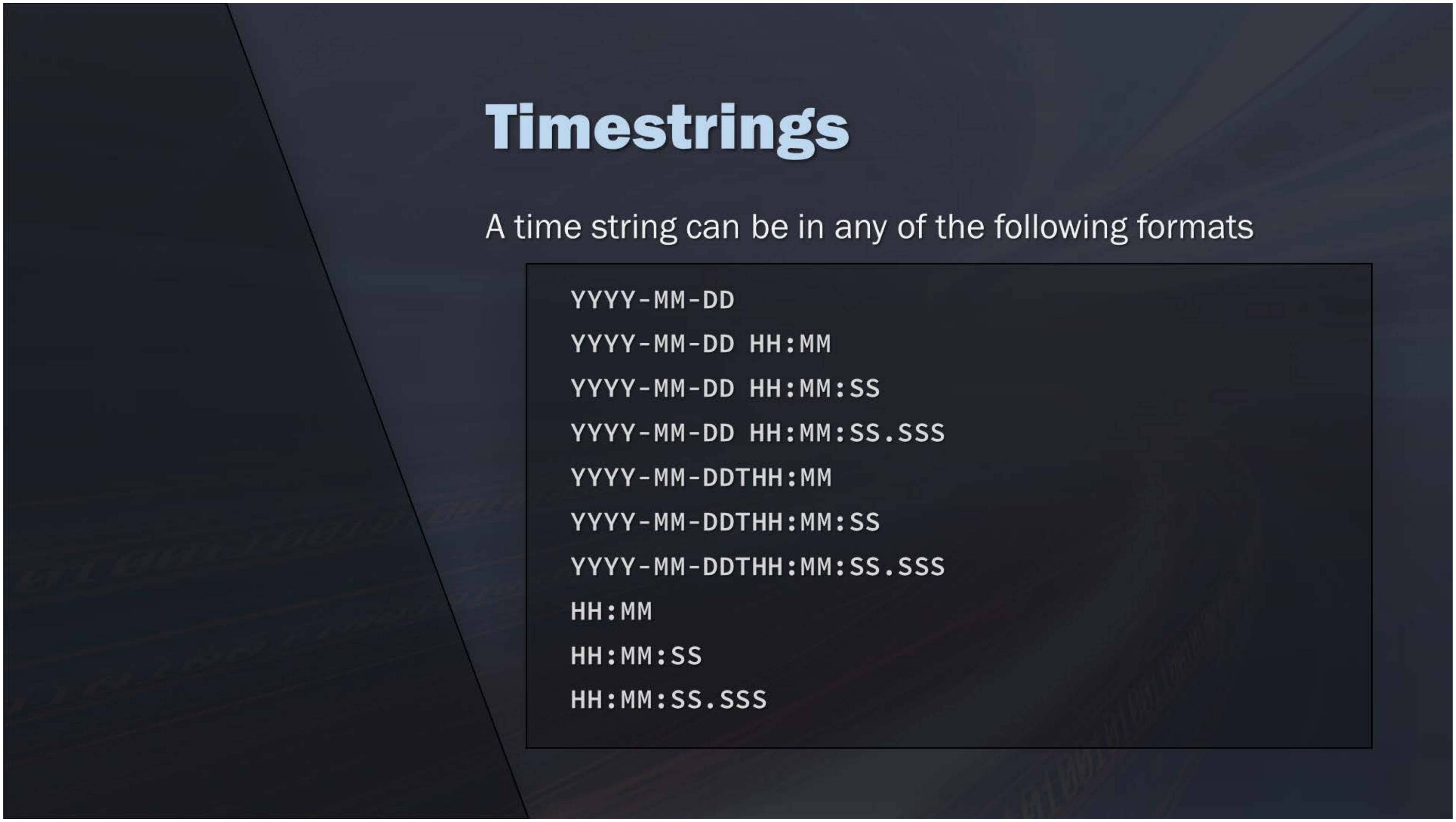


SQLite Date Time Functions

SQLite supports 5 date and time functions:

```
DATE(timestring, modifier, modifier, ...)
TIME(timestring, modifier, modifier, ...)
DATETIME(timestring, modifier, modifier, ...)
JULIANDAY(timestring, modifier, modifier, ...)
STRFTIME(format, timestring, modifier, modifier, ...)
```

Slide 6: Timestrings

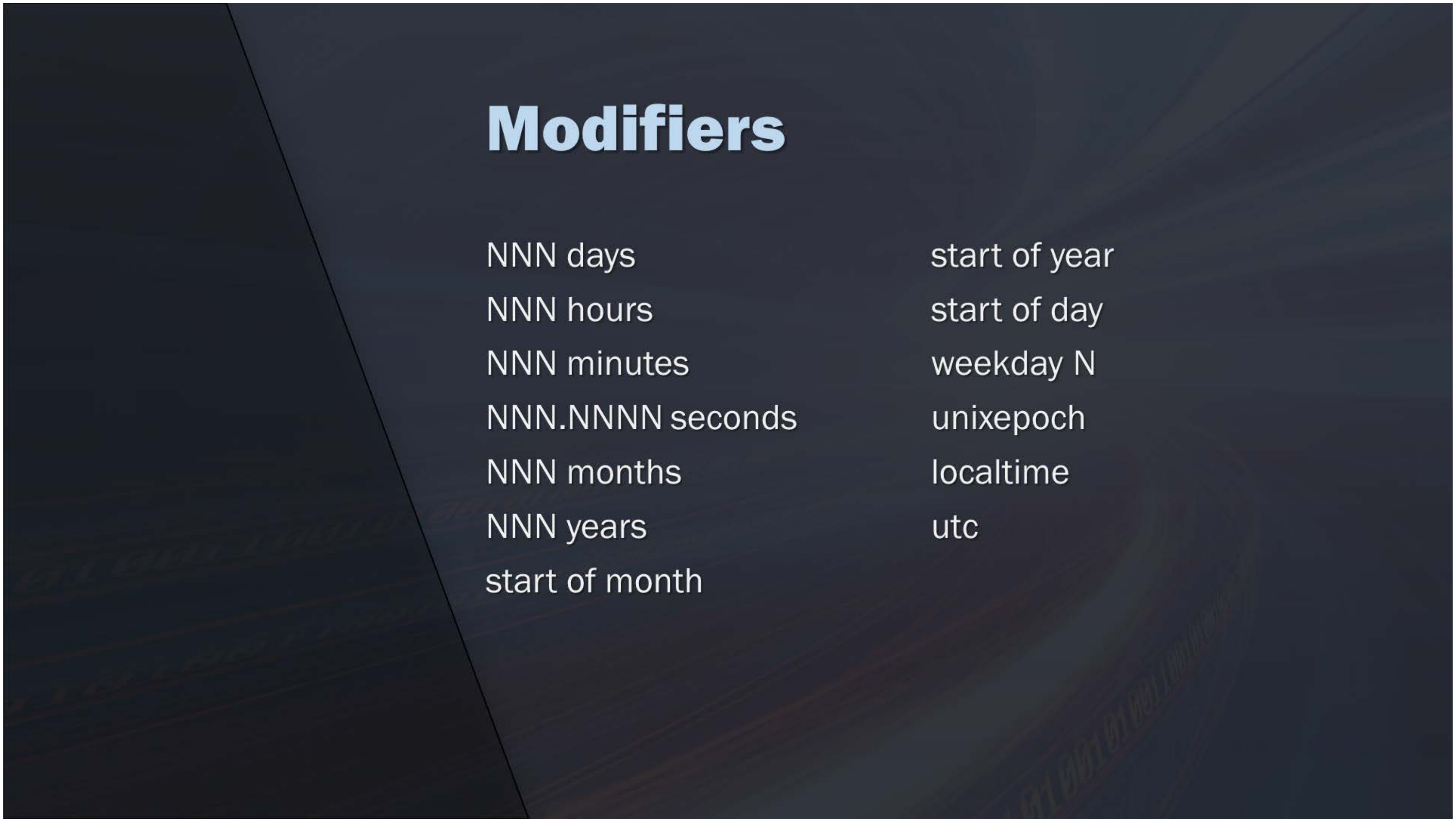


Timestrings

A time string can be in any of the following formats

```
YYYY-MM-DD
YYYY-MM-DD HH:MM
YYYY-MM-DD HH:MM:SS
YYYY-MM-DD HH:MM:SS.SSS
YYYY-MM-DDTHH:MM
YYYY-MM-DDTHH:MM:SS
YYYY-MM-DDTHH:MM:SS.SSS
HH:MM
HH:MM:SS
HH:MM:SS.SSS
```

Slide 7: Modifiers



Modifiers

NNN days

NNN hours

NNN minutes

NNN.NNNN seconds

NNN months

NNN years

start of month

start of year

start of day

weekday N

unixepoch

localtime

utc

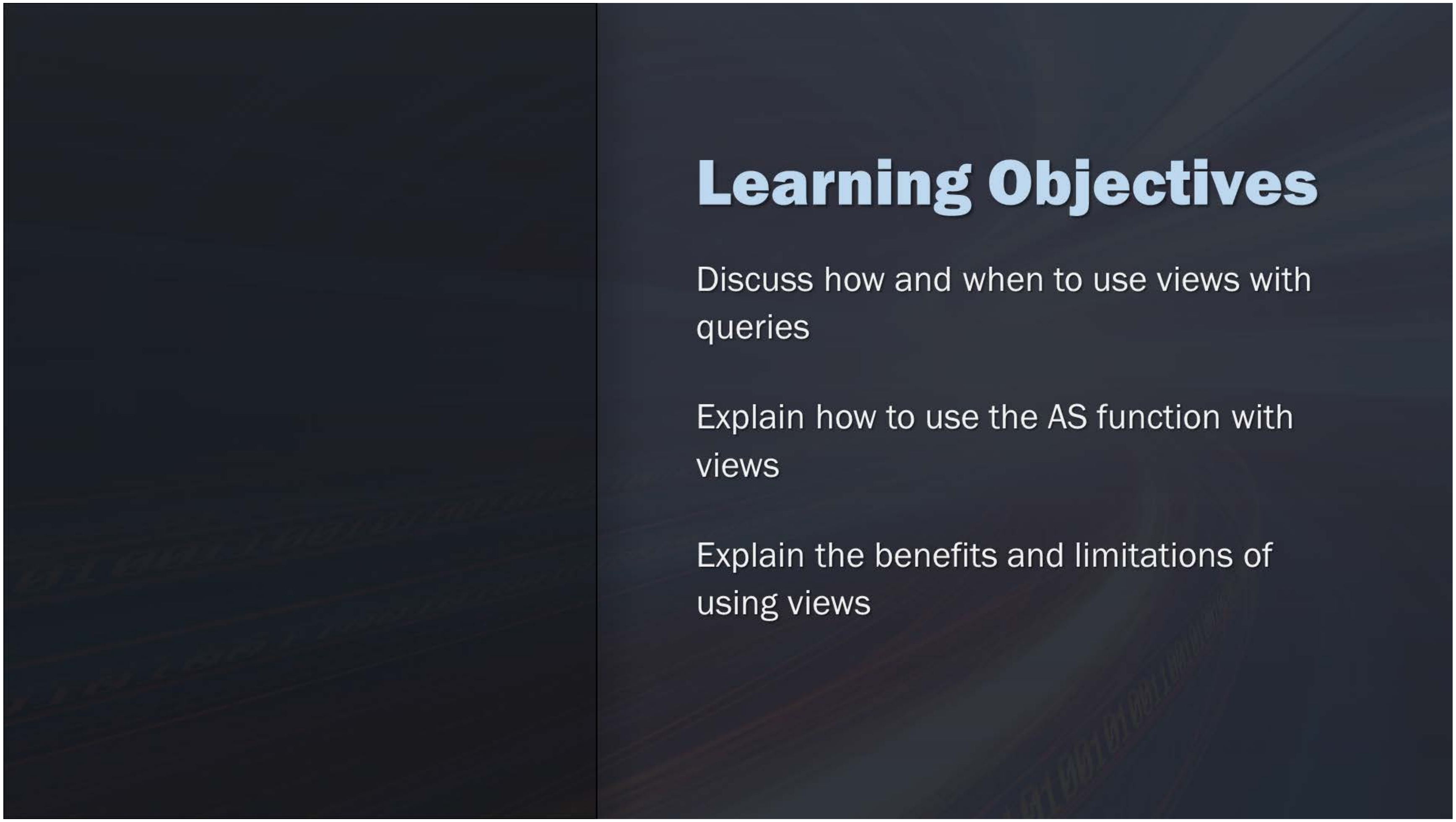
Slide 1: Views

Views

Sadie St. Lawrence, MSA
Data Scientist, VSP

UCDAVIS
EXTENSION

Slide 2: Learning Objectives



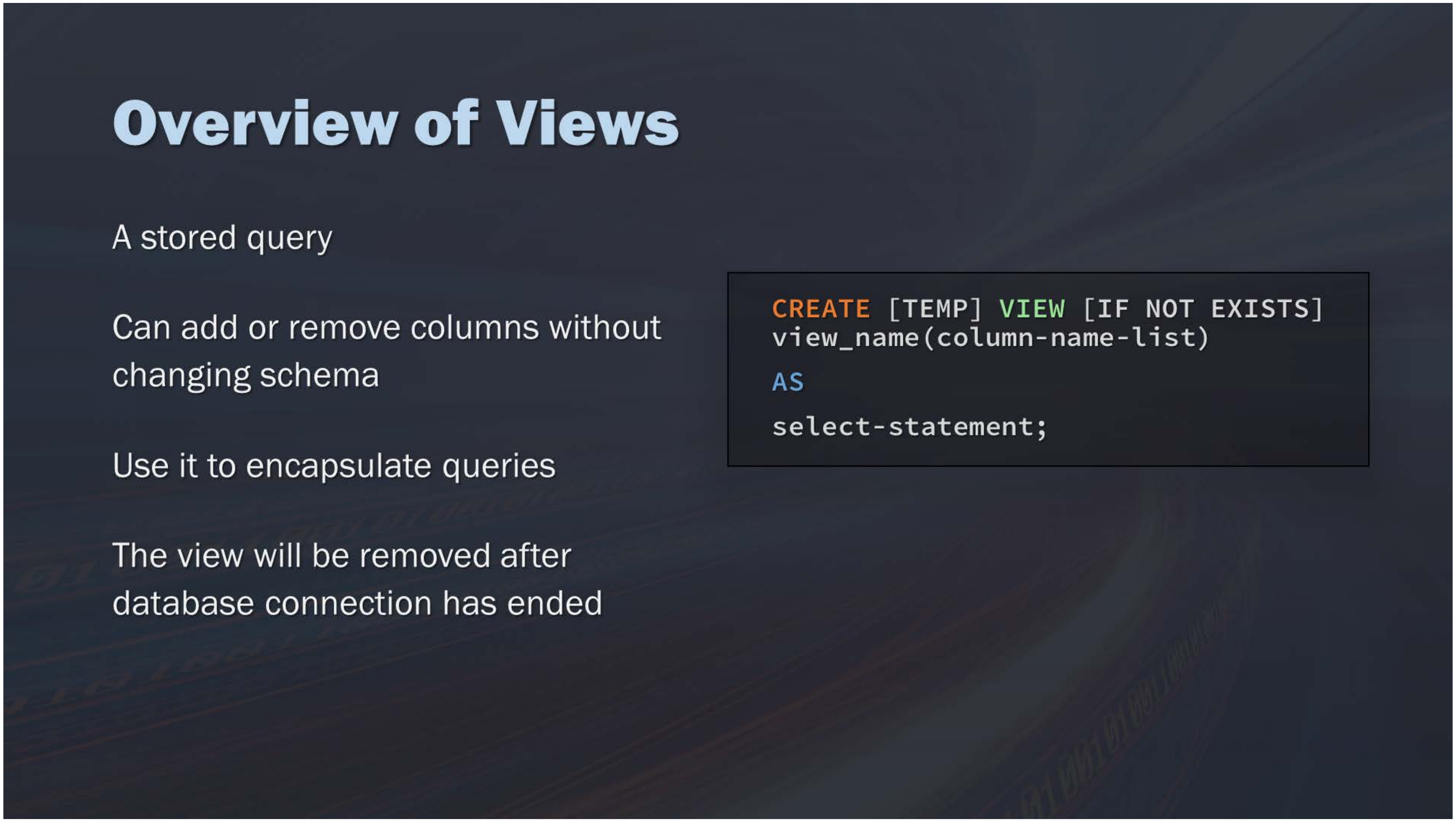
Learning Objectives

Discuss how and when to use views with queries

Explain how to use the AS function with views

Explain the benefits and limitations of using views

Slide 3: Overview of Views



Overview of Views

A stored query

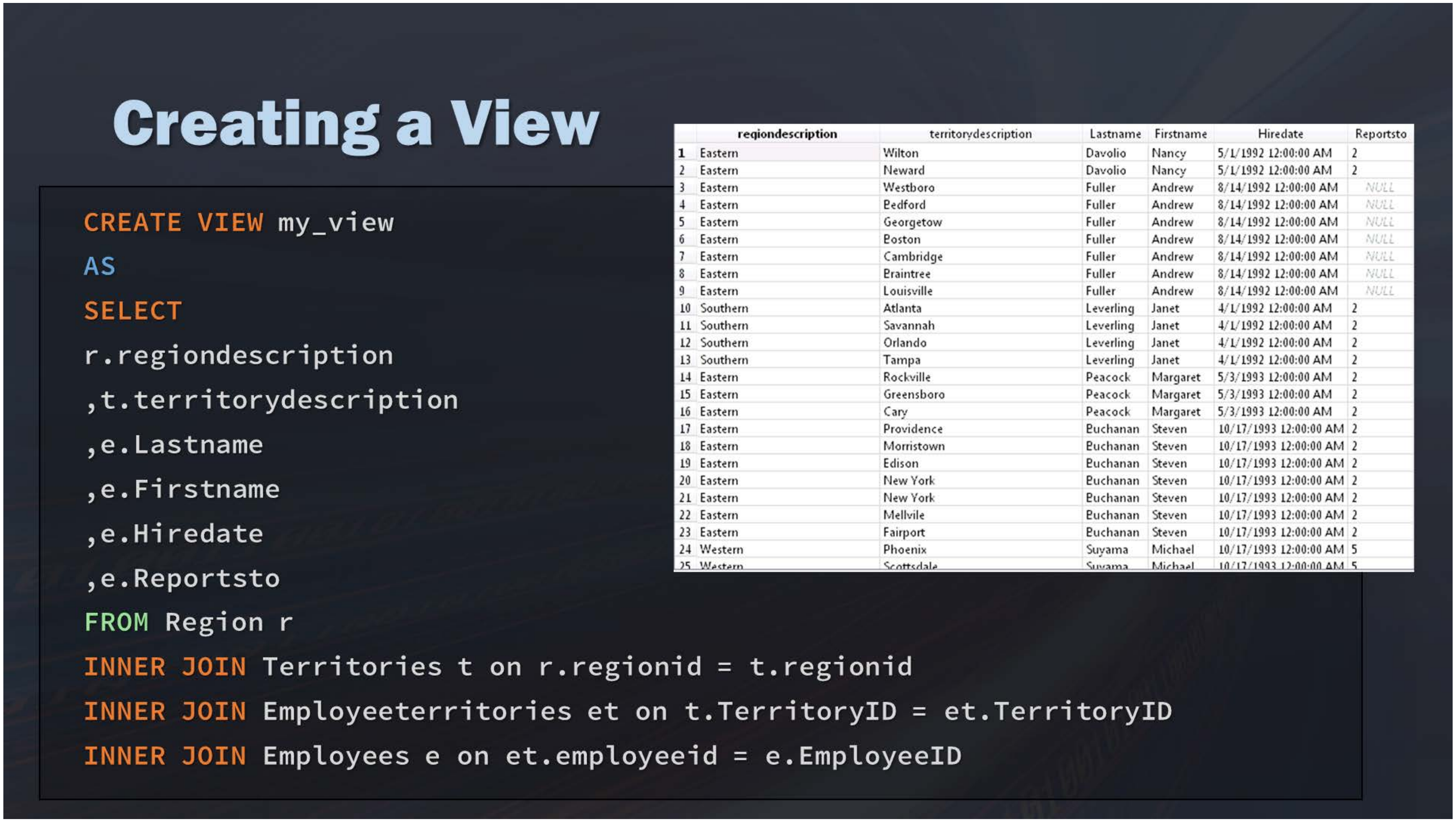
Can add or remove columns without changing schema

Use it to encapsulate queries

The view will be removed after database connection has ended

```
CREATE [TEMP] VIEW [IF NOT EXISTS]
view_name(column-name-list)
AS
select-statement;
```

Slide 4: Creating a View

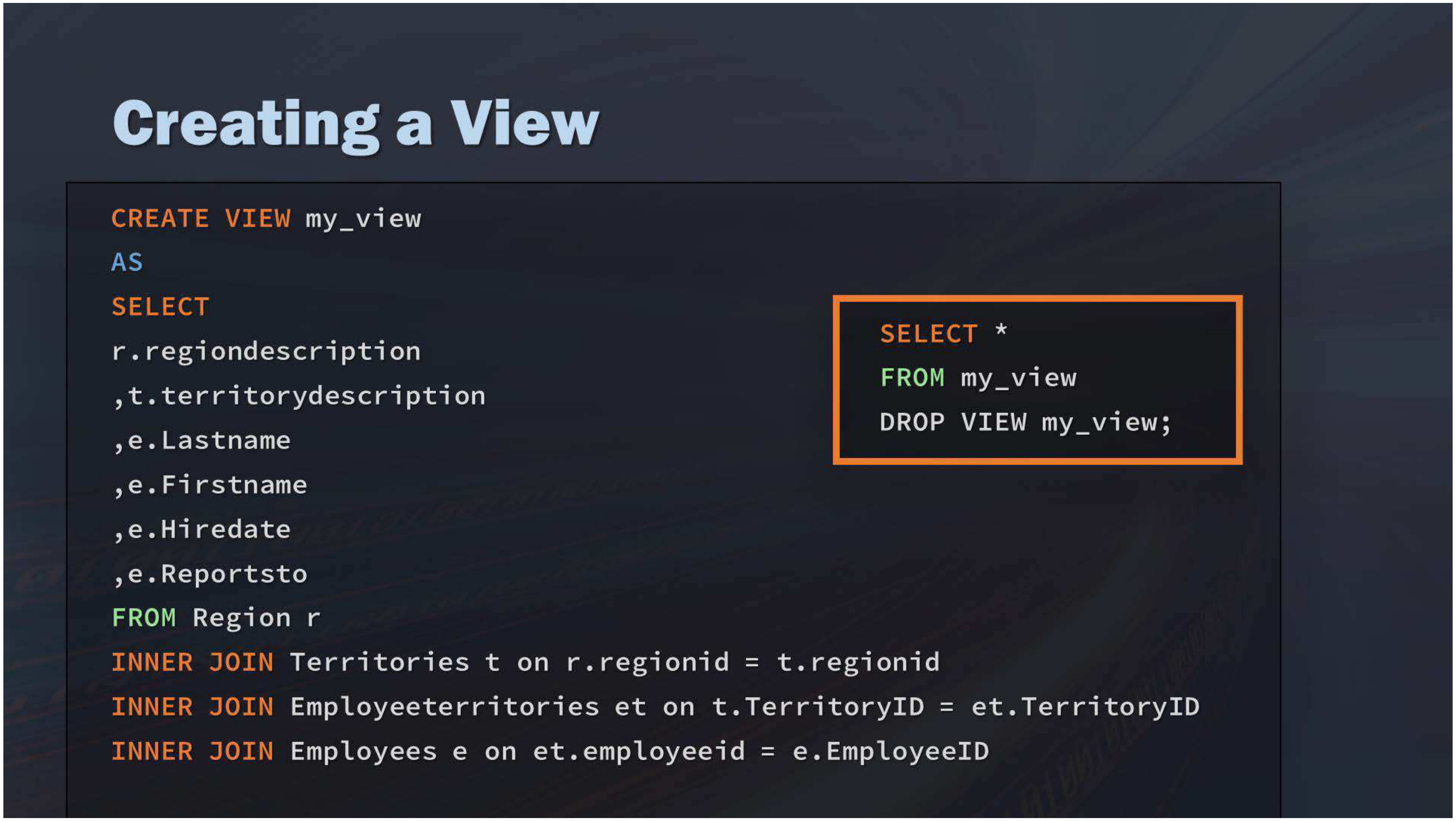


Creating a View

```
CREATE VIEW my_view
AS
SELECT
r.regiondescription
,t.territorydescription
,e.Lastname
,e.Firstname
,e.Hiredate
,e.Reportsto
FROM Region r
INNER JOIN Territories t on r.regionid = t.regionid
INNER JOIN Employeeterritories et on t.TerritoryID = et.TerritoryID
INNER JOIN Employees e on et.employeeid = e.EmployeeID
```

	regiondescription	territorydescription	Lastname	Firstname	Hiredate	Reportsto
1	Eastern	Wilton	Davolio	Nancy	5/1/1992 12:00:00 AM	2
2	Eastern	Neward	Davolio	Nancy	5/1/1992 12:00:00 AM	2
3	Eastern	Westboro	Fuller	Andrew	8/14/1992 12:00:00 AM	NULL
4	Eastern	Bedford	Fuller	Andrew	8/14/1992 12:00:00 AM	NULL
5	Eastern	Georgetow	Fuller	Andrew	8/14/1992 12:00:00 AM	NULL
6	Eastern	Boston	Fuller	Andrew	8/14/1992 12:00:00 AM	NULL
7	Eastern	Cambridge	Fuller	Andrew	8/14/1992 12:00:00 AM	NULL
8	Eastern	Braintree	Fuller	Andrew	8/14/1992 12:00:00 AM	NULL
9	Eastern	Louisville	Fuller	Andrew	8/14/1992 12:00:00 AM	NULL
10	Southern	Atlanta	Leverling	Janet	4/1/1992 12:00:00 AM	2
11	Southern	Savannah	Leverling	Janet	4/1/1992 12:00:00 AM	2
12	Southern	Orlando	Leverling	Janet	4/1/1992 12:00:00 AM	2
13	Southern	Tampa	Leverling	Janet	4/1/1992 12:00:00 AM	2
14	Eastern	Rockville	Peacock	Margaret	5/3/1993 12:00:00 AM	2
15	Eastern	Greensboro	Peacock	Margaret	5/3/1993 12:00:00 AM	2
16	Eastern	Cary	Peacock	Margaret	5/3/1993 12:00:00 AM	2
17	Eastern	Providence	Buchanan	Steven	10/17/1993 12:00:00 AM	2
18	Eastern	Morristown	Buchanan	Steven	10/17/1993 12:00:00 AM	2
19	Eastern	Edison	Buchanan	Steven	10/17/1993 12:00:00 AM	2
20	Eastern	New York	Buchanan	Steven	10/17/1993 12:00:00 AM	2
21	Eastern	New York	Buchanan	Steven	10/17/1993 12:00:00 AM	2
22	Eastern	Mellvile	Buchanan	Steven	10/17/1993 12:00:00 AM	2
23	Eastern	Fairport	Buchanan	Steven	10/17/1993 12:00:00 AM	2
24	Western	Phoenix	Suyama	Michael	10/17/1993 12:00:00 AM	5
25	Western	Scottsdale	Suyama	Michael	10/17/1993 12:00:00 AM	5

Slide 5: Creating a View

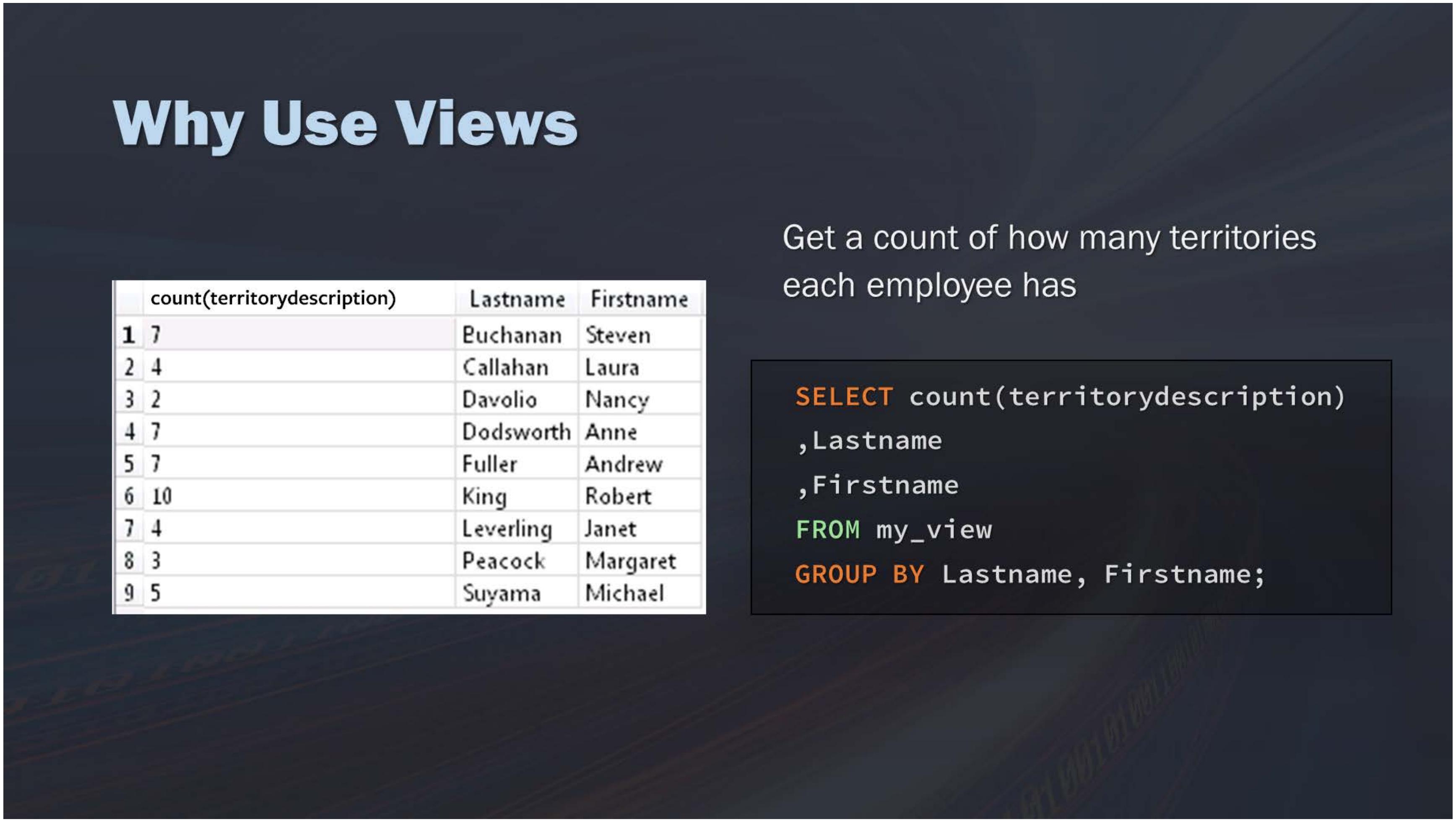


Creating a View

```
CREATE VIEW my_view
AS
SELECT
r.regiondescription
,t.territorydescription
,e.Lastname
,e.Firstname
,e.Hiredate
,e.Reportsto
FROM Region r
INNER JOIN Territories t on r.regionid = t.regionid
INNER JOIN Employeeterritories et on t.TerritoryID = et.TerritoryID
INNER JOIN Employees e on et.employeeid = e.EmployeeID
```

```
SELECT *
FROM my_view
DROP VIEW my_view;
```

Slide 6: Why Use Views



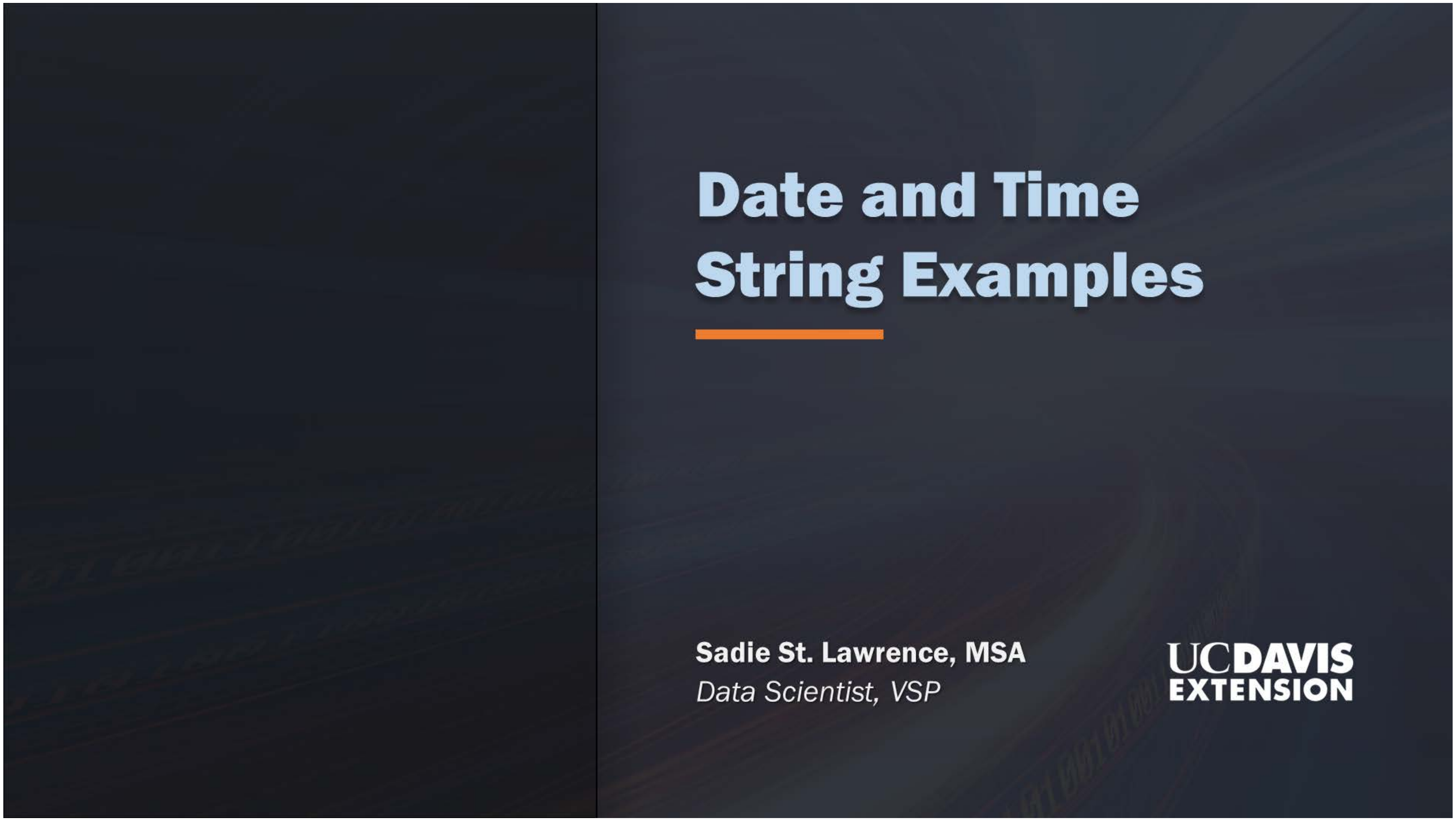
Why Use Views

Get a count of how many territories each employee has

	count(territorydescription)	Lastname	Firstname
1	7	Buchanan	Steven
2	4	Callahan	Laura
3	2	Davolio	Nancy
4	7	Dodsworth	Anne
5	7	Fuller	Andrew
6	10	King	Robert
7	4	Leverling	Janet
8	3	Peacock	Margaret
9	5	Suyama	Michael

```sql
SELECT count(territorydescription)
,Lastname
,Firstname
FROM my_view
GROUP BY Lastname, Firstname;
```

Slide 1: Date and Time String Examples

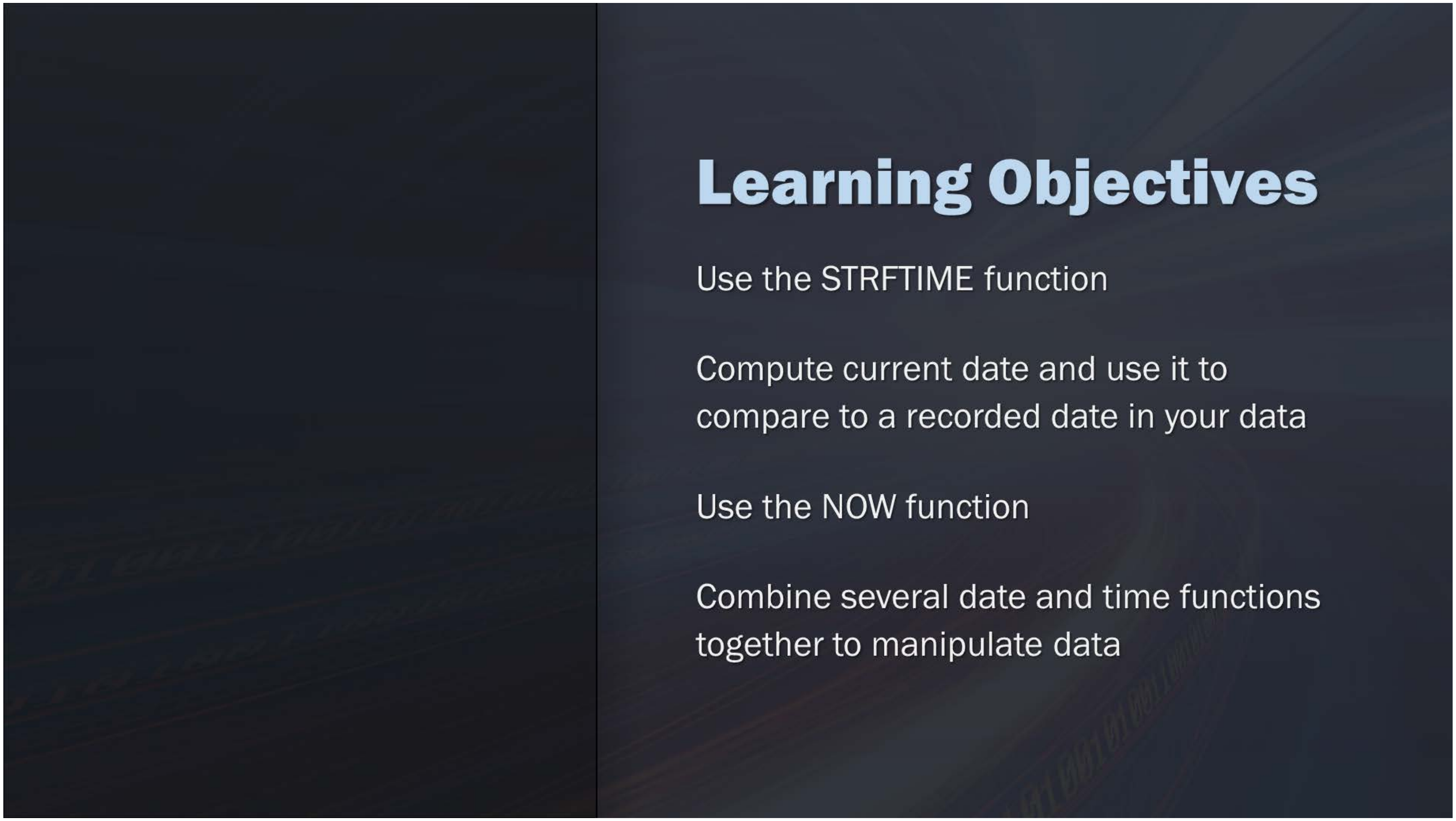


Date and Time String Examples

Sadie St. Lawrence, MSA
Data Scientist, VSP

UC DAVIS EXTENSION

Slide 2: Learning Objectives



Learning Objectives

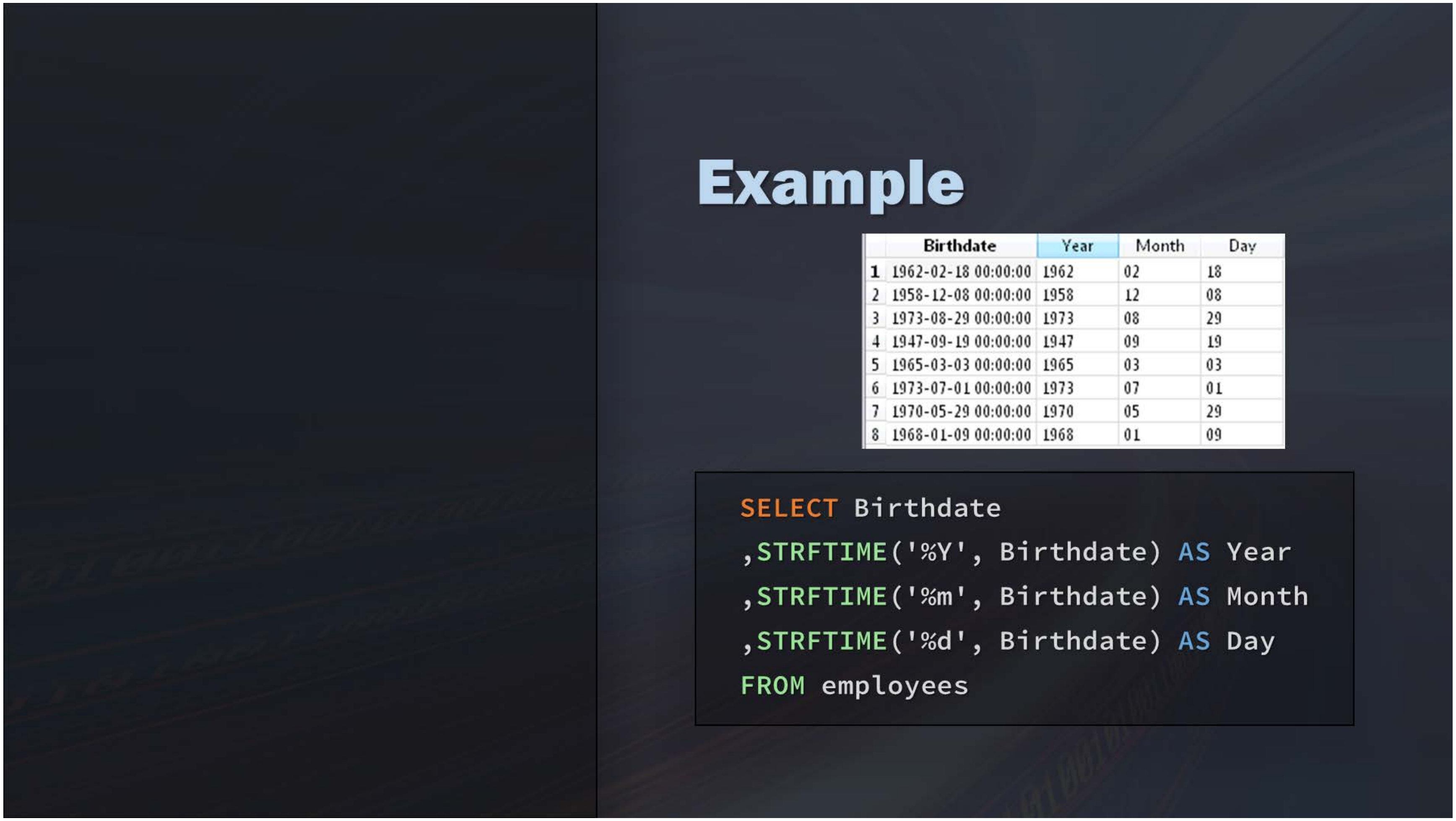
Use the STRFTIME function

Compute current date and use it to compare to a recorded date in your data

Use the NOW function

Combine several date and time functions together to manipulate data

Slide 3: Example

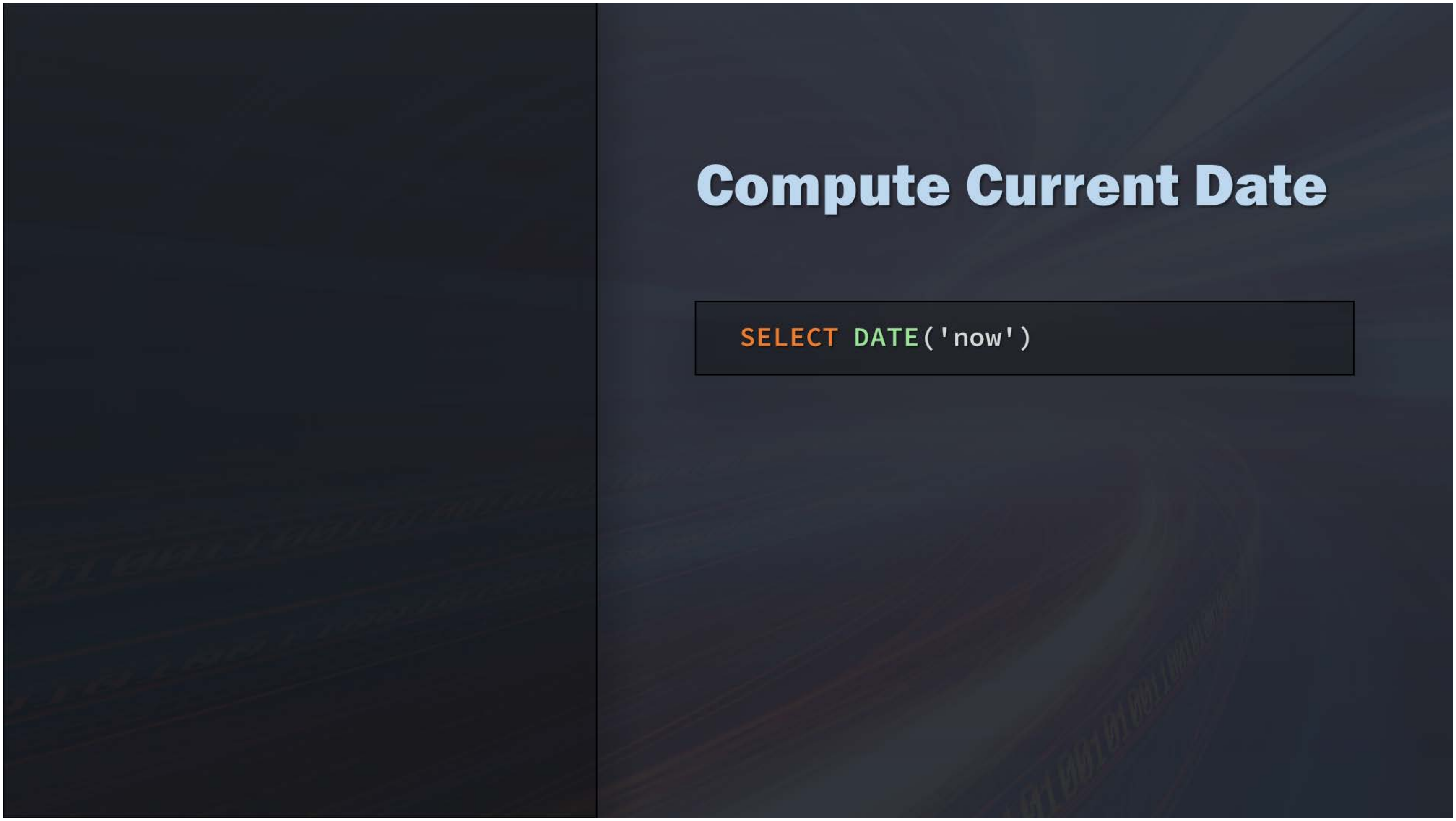


Example

	Birthdate	Year	Month	Day
1	1962-02-18 00:00:00	1962	02	18
2	1958-12-08 00:00:00	1958	12	08
3	1973-08-29 00:00:00	1973	08	29
4	1947-09-19 00:00:00	1947	09	19
5	1965-03-03 00:00:00	1965	03	03
6	1973-07-01 00:00:00	1973	07	01
7	1970-05-29 00:00:00	1970	05	29
8	1968-01-09 00:00:00	1968	01	09

```sql
SELECT Birthdate
,STRFTIME('%Y', Birthdate) AS Year
,STRFTIME('%m', Birthdate) AS Month
,STRFTIME('%d', Birthdate) AS Day
FROM employees
```

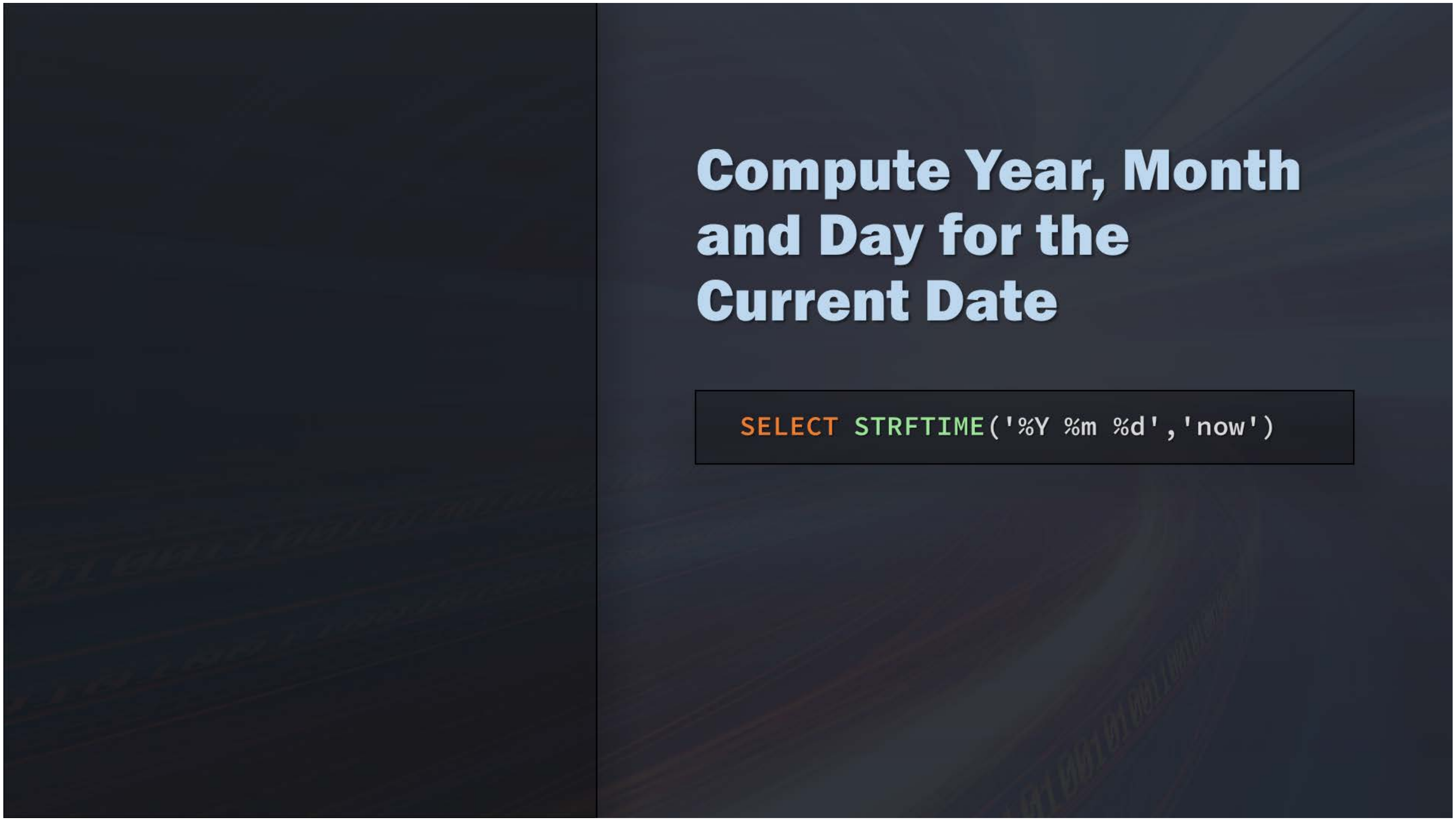
Slide 4: Compute Current Date



Compute Current Date

```
SELECT DATE('now')
```

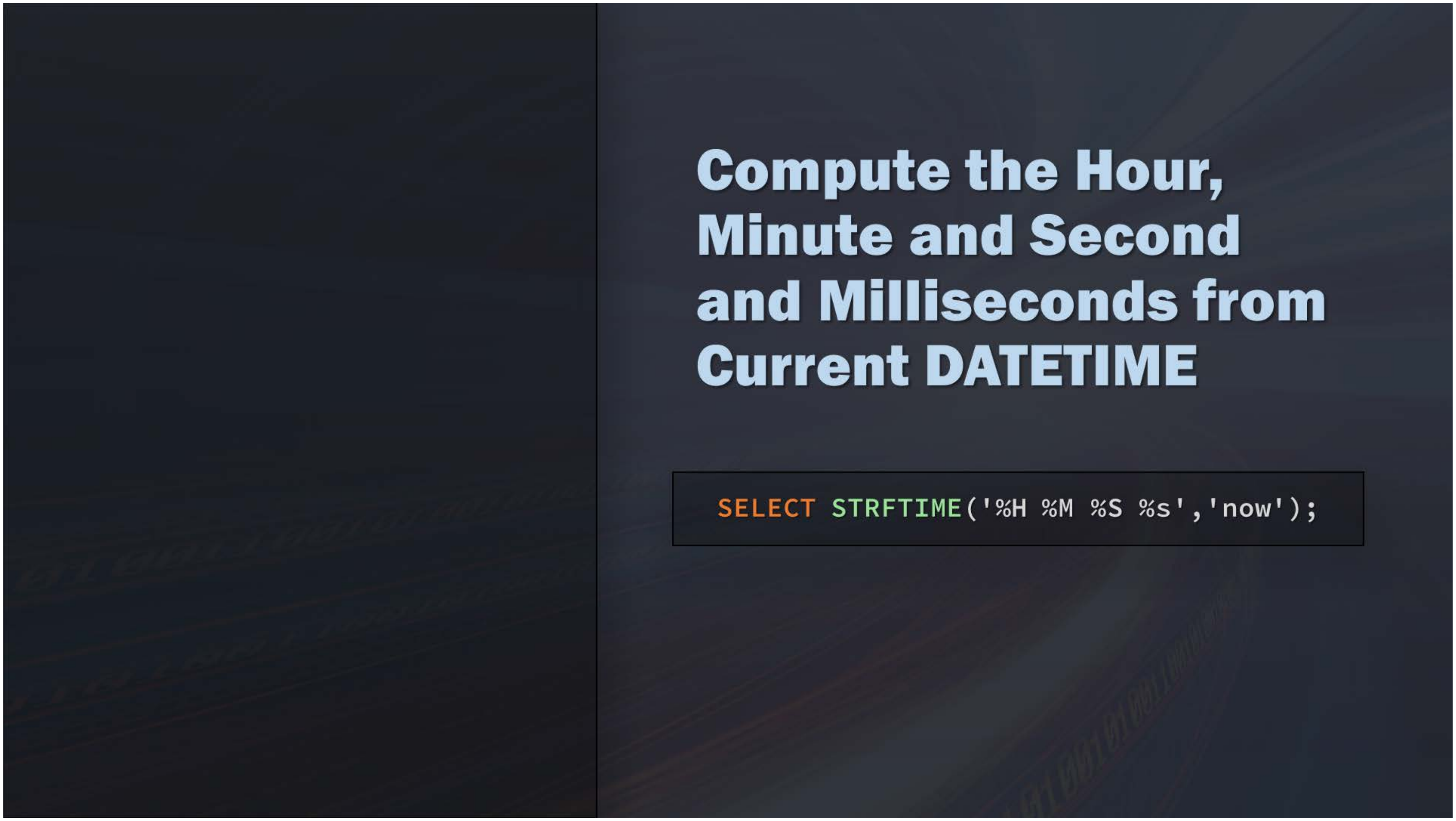
Slide 5: Compute Year, Month, and Date for the Current Date



Compute Year, Month and Day for the Current Date

```
SELECT STRFTIME('%Y %m %d','now')
```

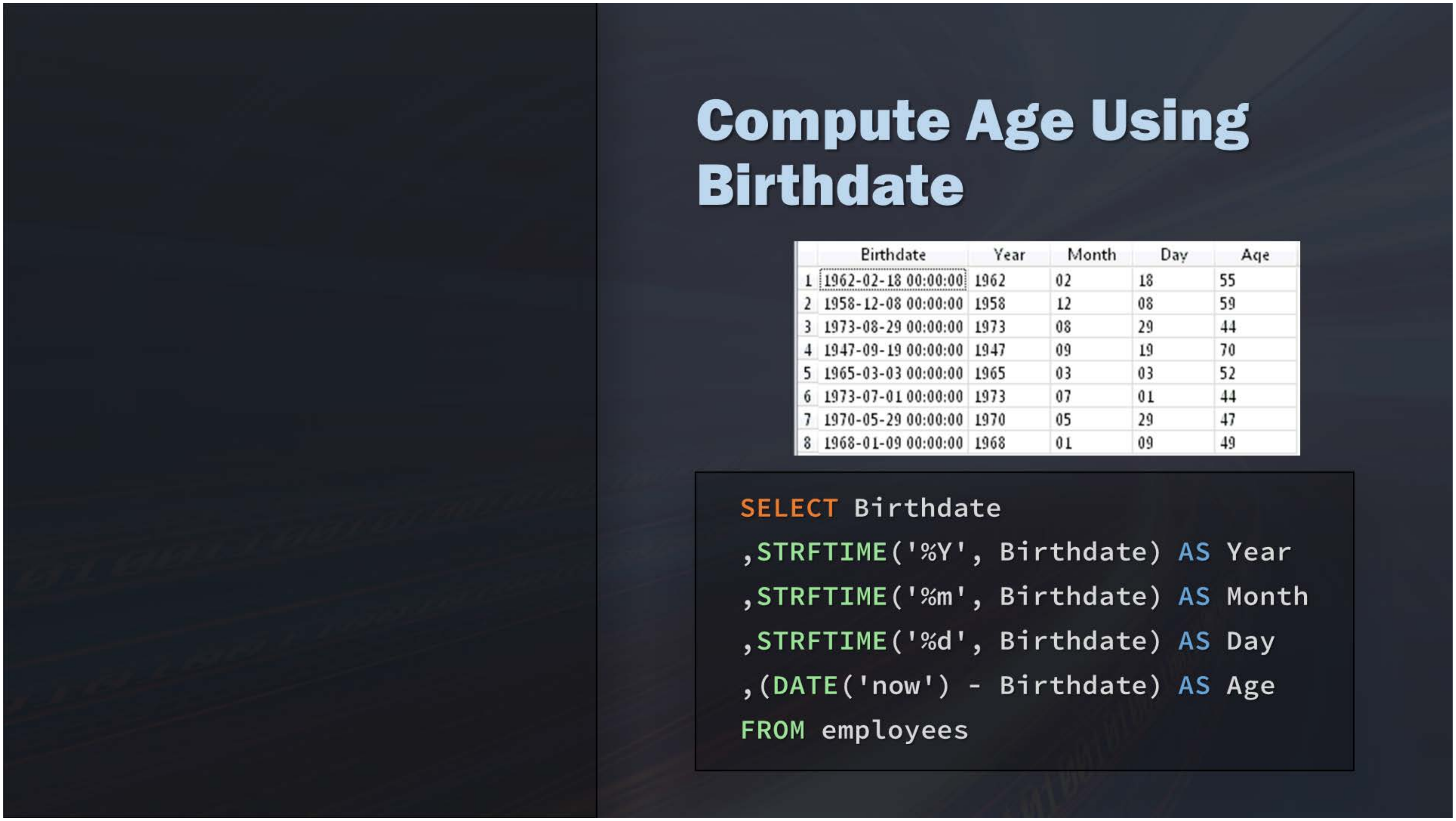
Slide 6: Compute the Hour, Minute and Second and Milliseconds from Current DATETIME



Compute the Hour, Minute and Second and Milliseconds from Current DATETIME

```
SELECT STRFTIME('%H %M %S %s','now');
```

Slide 7: Compute Age Using Birthdate



Compute Age Using Birthdate

	Birthdate	Year	Month	Day	Age
1	1962-02-18 00:00:00	1962	02	18	55
2	1958-12-08 00:00:00	1958	12	08	59
3	1973-08-29 00:00:00	1973	08	29	44
4	1947-09-19 00:00:00	1947	09	19	70
5	1965-03-03 00:00:00	1965	03	03	52
6	1973-07-01 00:00:00	1973	07	01	44
7	1970-05-29 00:00:00	1970	05	29	47
8	1968-01-09 00:00:00	1968	01	09	49

```sql
SELECT Birthdate
,STRFTIME('%Y', Birthdate) AS Year
,STRFTIME('%m', Birthdate) AS Month
,STRFTIME('%d', Birthdate) AS Day
,(DATE('now') - Birthdate) AS Age
FROM employees
```