3) Building SAS Studio Flows in SAS Viya (VSTU1C)

Objective

- Understand how to use SAS Studio
- Be able to run and create flow in SAS Studio

Outline

1. Introduction (Lesson1)
    1. Introduction to SAS Studio
    2. SAS Studio Flows
2. Accessing Data (Lesson2)
    1. Understand SAS Data
    2. Accessing Data through Libraries
    3. Importing Data
3. Transforming and Analyzing Data (Lesson3)
    1. Create Simple Queries
    2. Using SAS Studio Analyst Steps
    3. Creating Results with Tasks

Lesson 1: Getting Started

1.1 Introducing SAS Studio

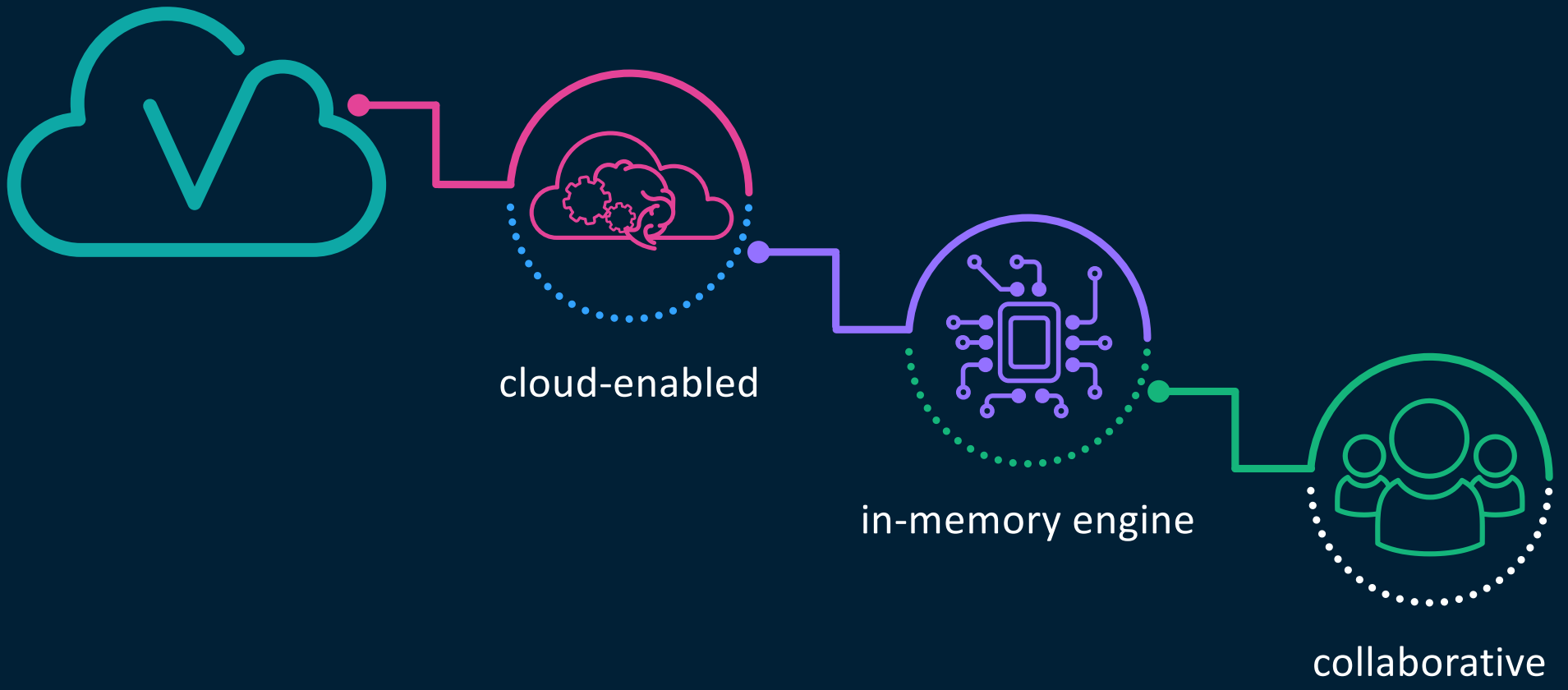
1.2 SAS Studio Flows

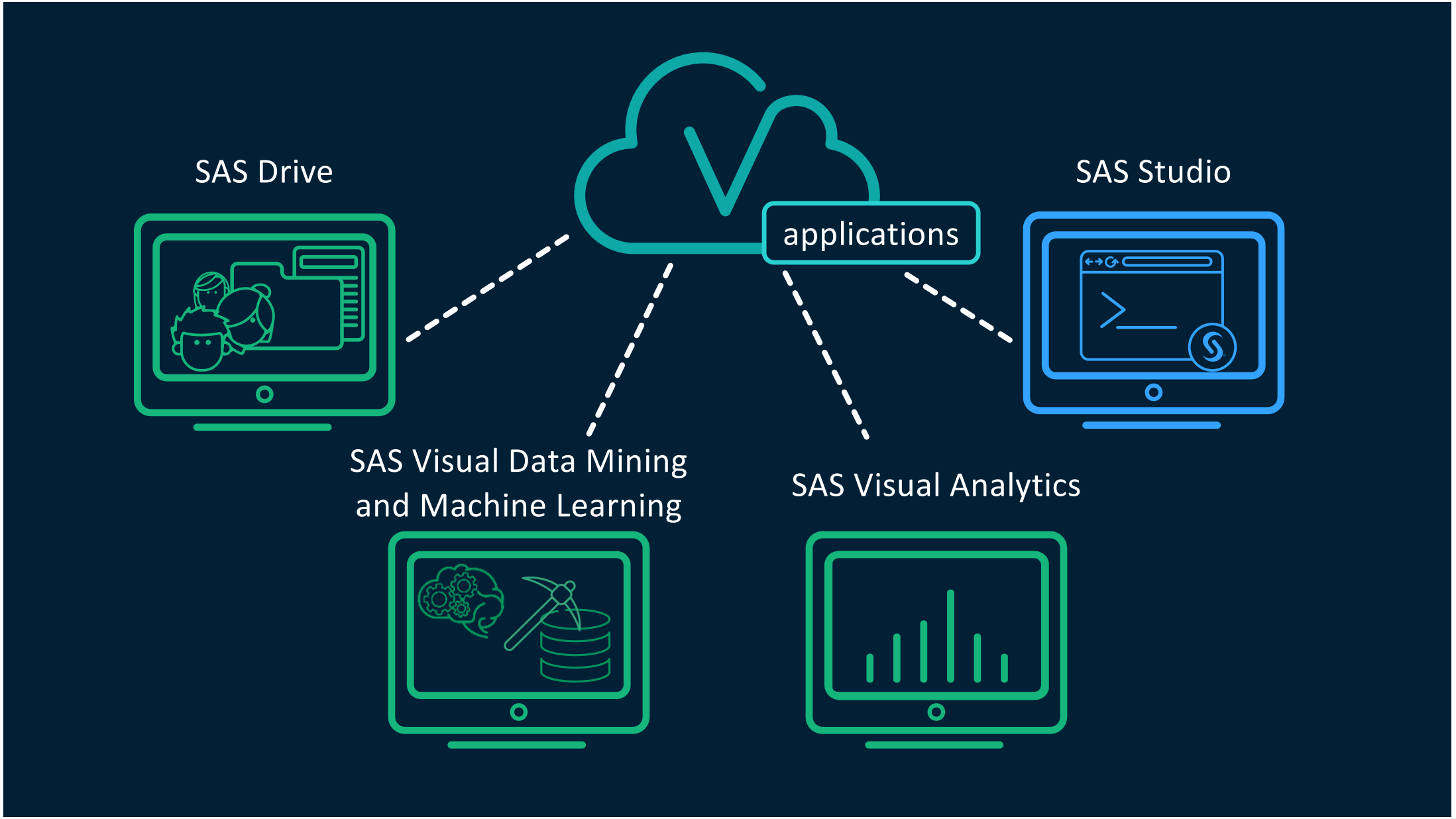
Lesson 1: Getting Started

1.1 Introducing SAS Studio

1.2 SAS Studio Flows

What Is SAS Viya?




SAS Drive
applications
SAS Studio
SAS Visual Data Mining
and Machine Learning
SAS Visual Analytics

What Is SAS Studio?

point-and-click

Program Editor

flows

tasks

snippets

syntax Help
and autocomplete

da

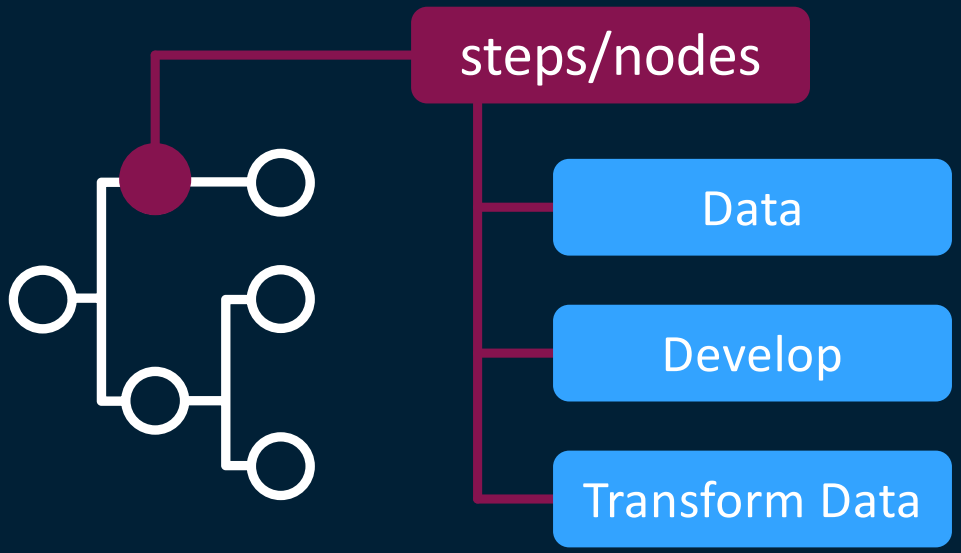
data

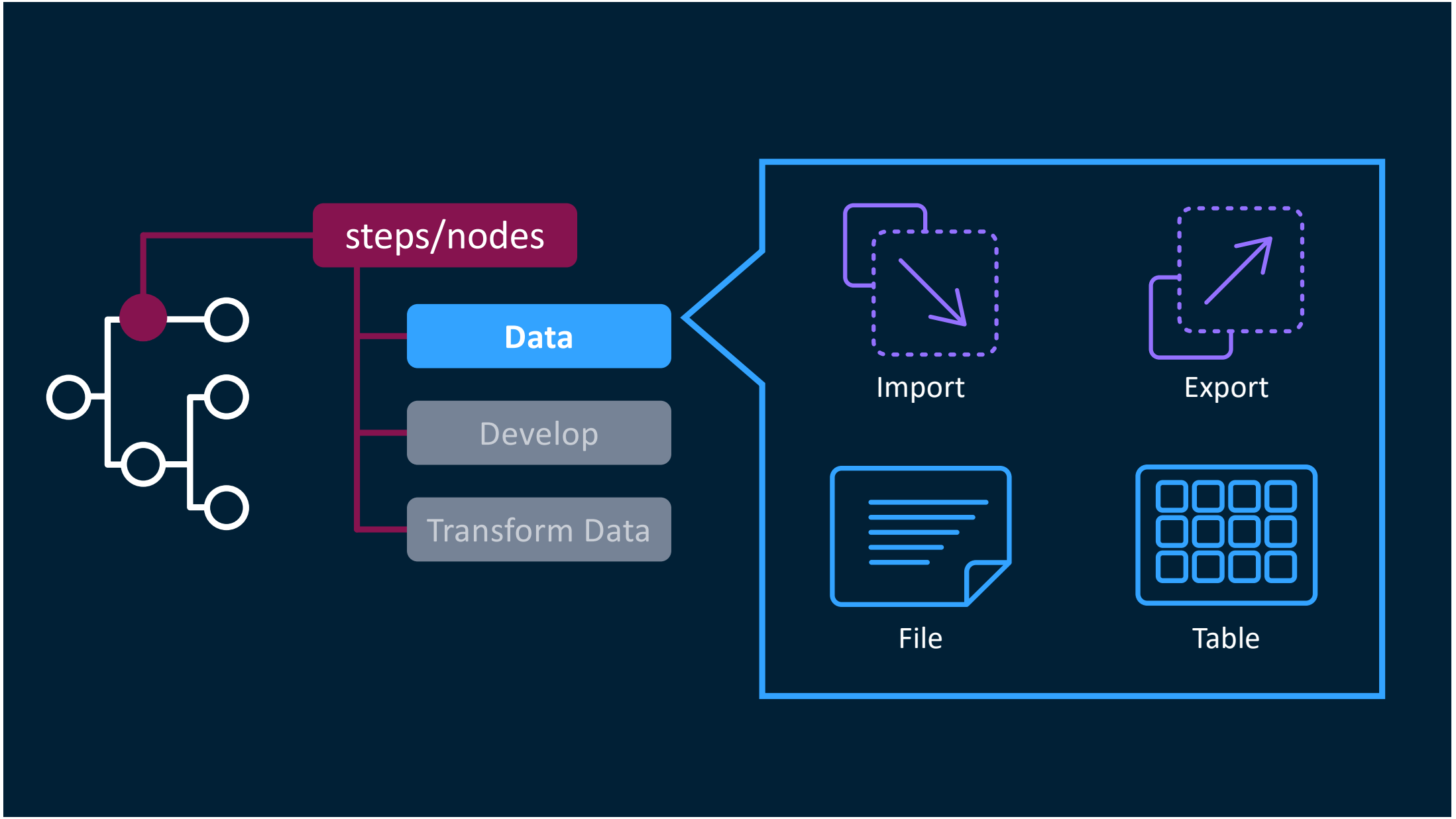
Lesson 1: Getting Started

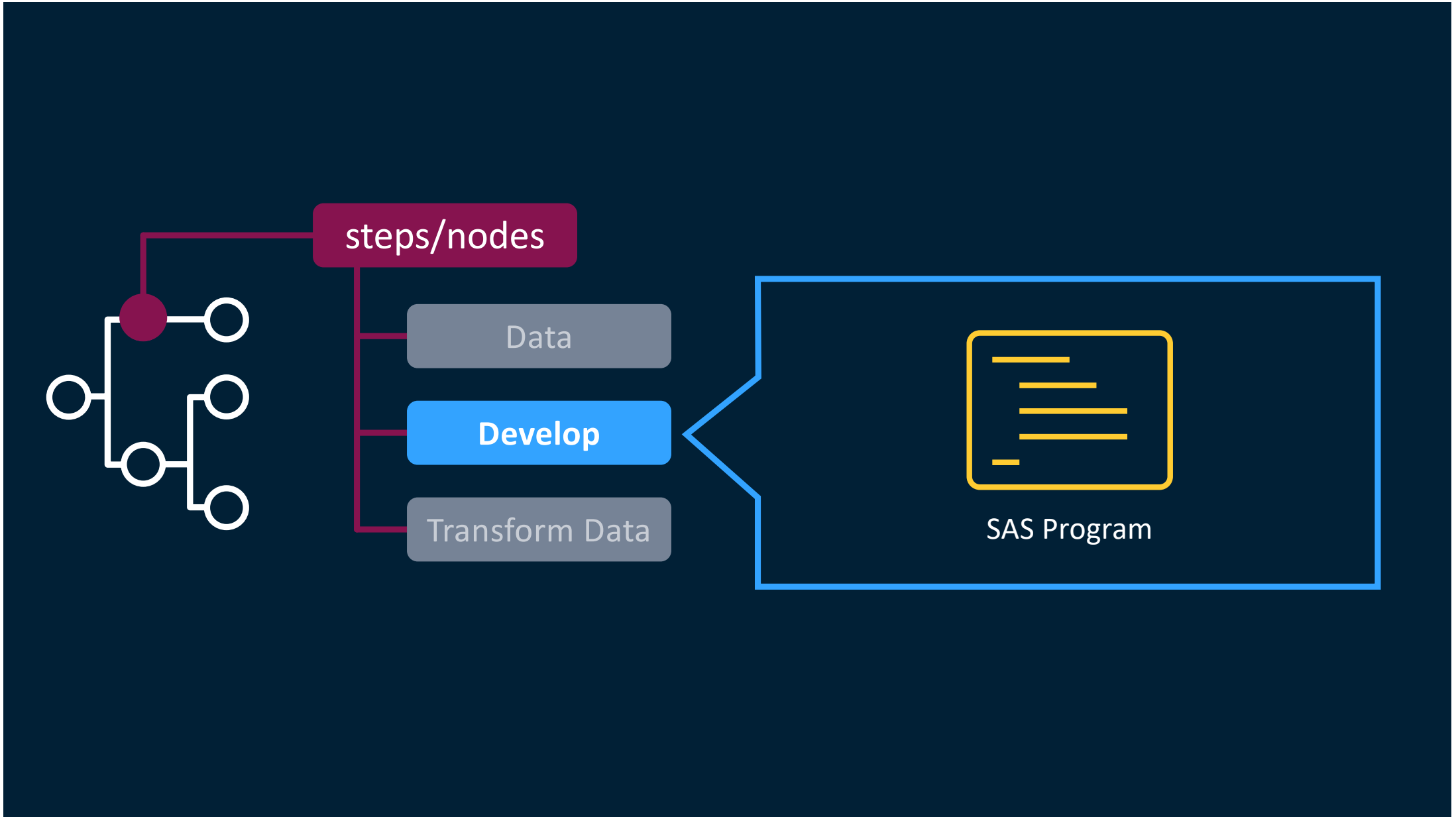
1.1 Introducing SAS Studio

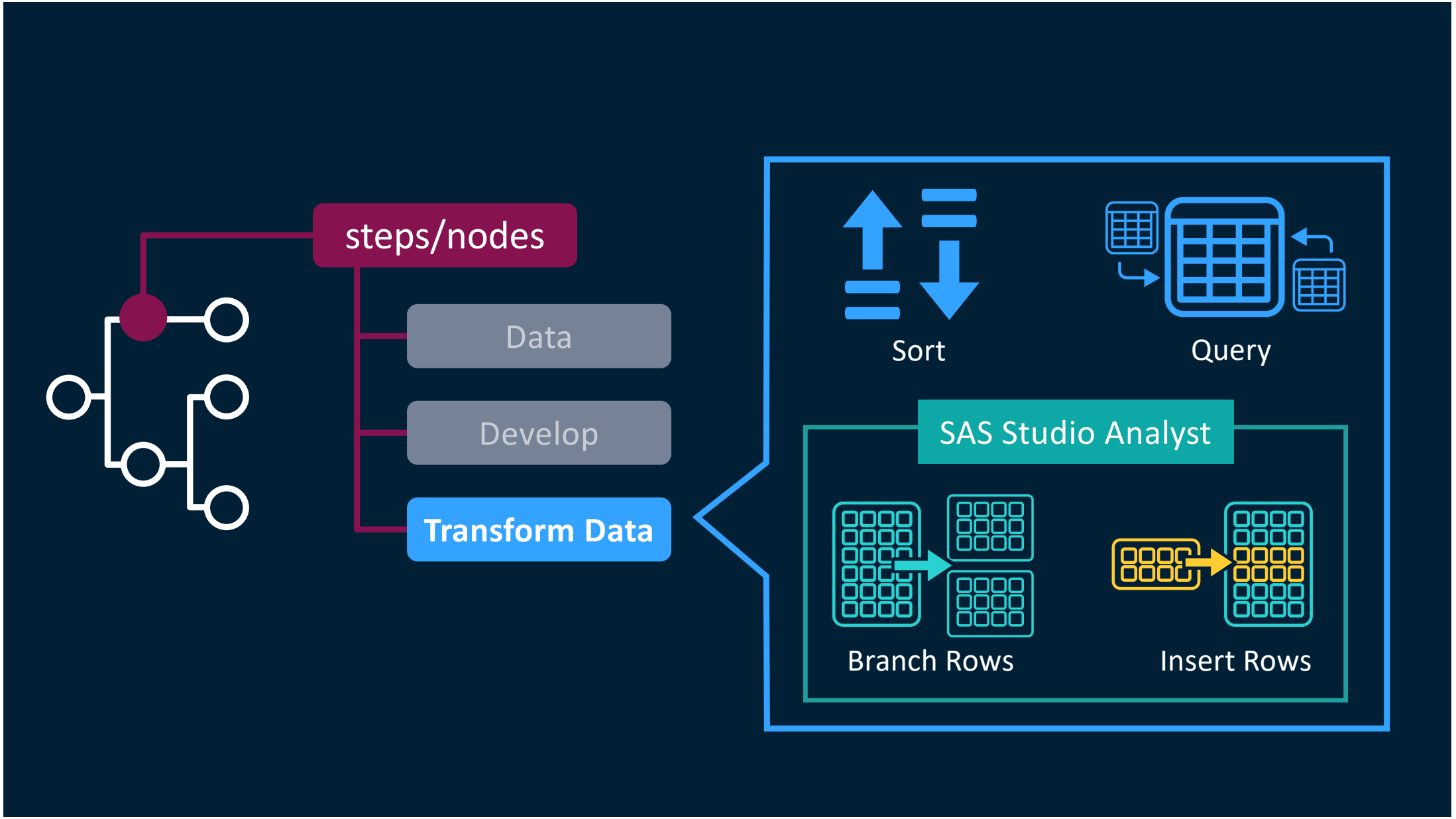
1.2 SAS Studio Flows

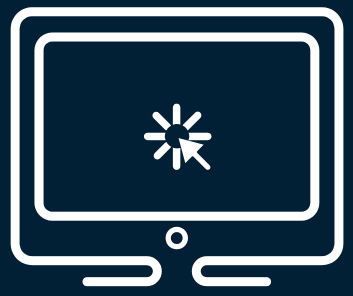
SAS Studio Flows




steps/nodes
Data
Develop
Transform Data
Import
Export
File
Table


steps/nodes
Data
Develop
Transform Data
SAS Program


steps/nodes
Data
Develop
Transform Data
Sort
Query
SAS Studio Analyst
Branch Rows
Insert Rows



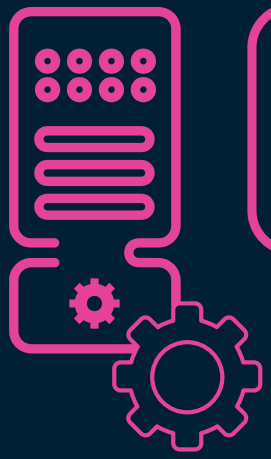
Navigating SAS Studio and Running a Flow

This demonstration illustrates how to navigate the SAS Studio interface, open and run a flow, and explore the results of a flow.

How Does SAS Studio Work?

Viya servers

SAS Compute Server



- default server
- executes traditional SAS®9 code

SAS Cloud Analytic Services (CAS)

- high-performance server
- performs parallel processing on in-memory data
- executes CAS-enabled code

Lesson 2: Accessing Data

2.1 Understanding SAS Data

2.2 Accessing Data through Libraries

2.3 Importing Data

Lesson 2: Accessing Data

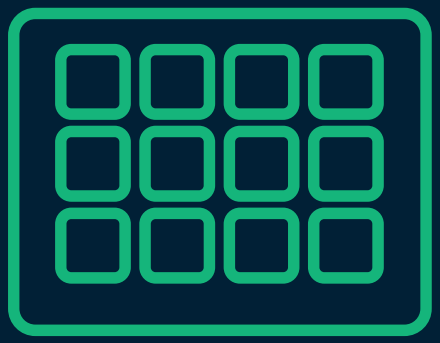
2.1 Understanding SAS Data

2.2 Accessing Data through Libraries

2.3 Importing Data

Types of Data

structured data

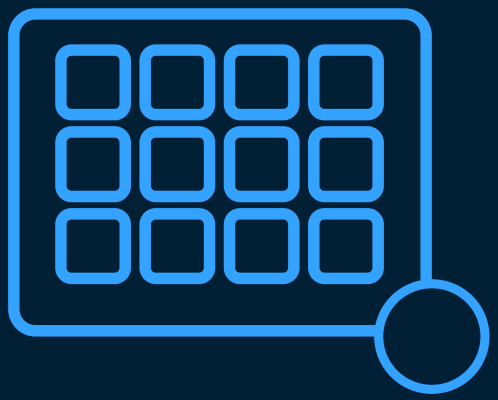


- includes defined rows and columns
- SAS, Microsoft Access and Excel, Oracle, Teradata, Hadoop, and others
- engines enable SAS to read the data

unstructured data

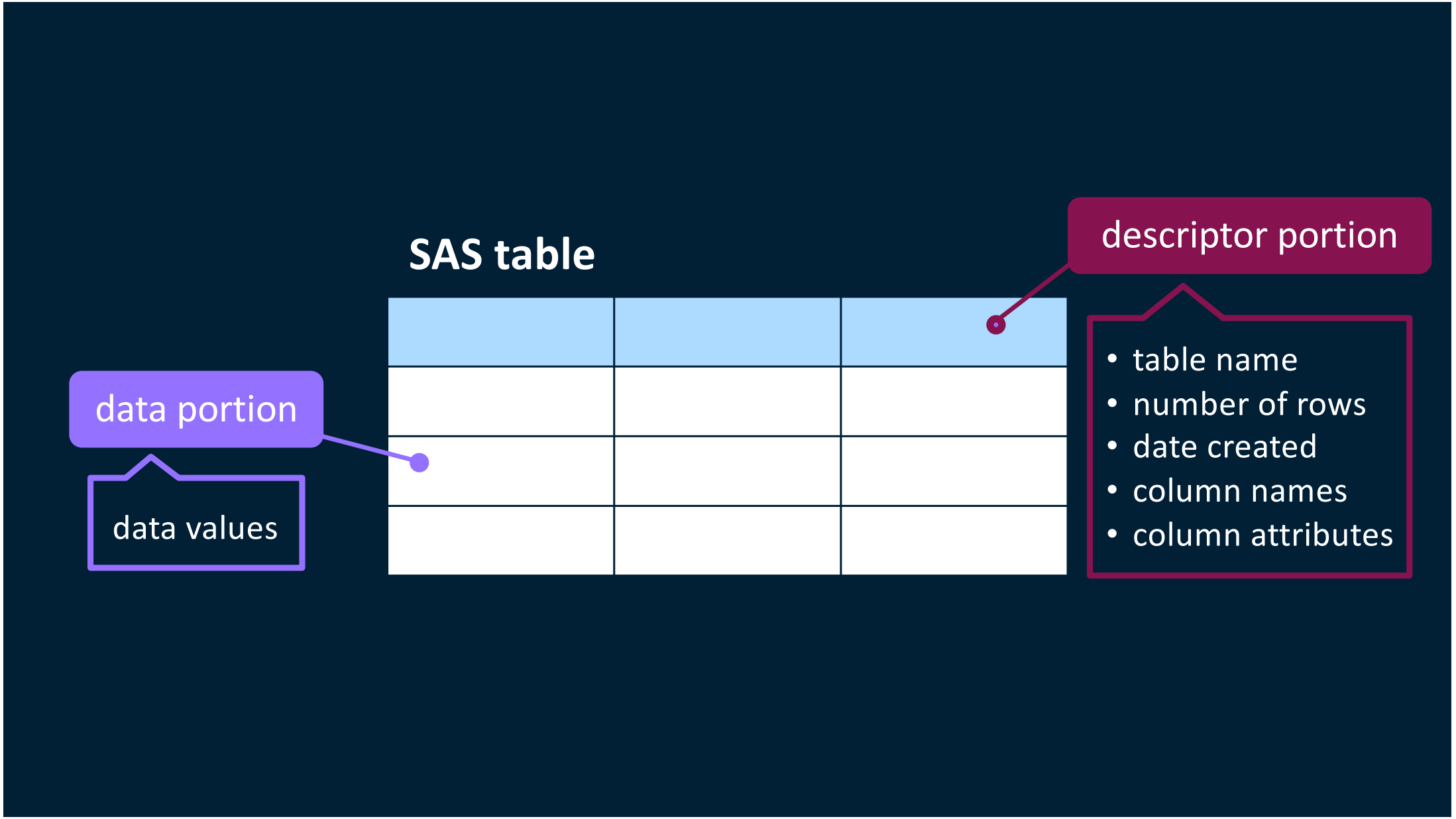
- no defined columns
- text, delimited, and others
- must be imported into SAS

SAS table



structured data

.sas7bdat

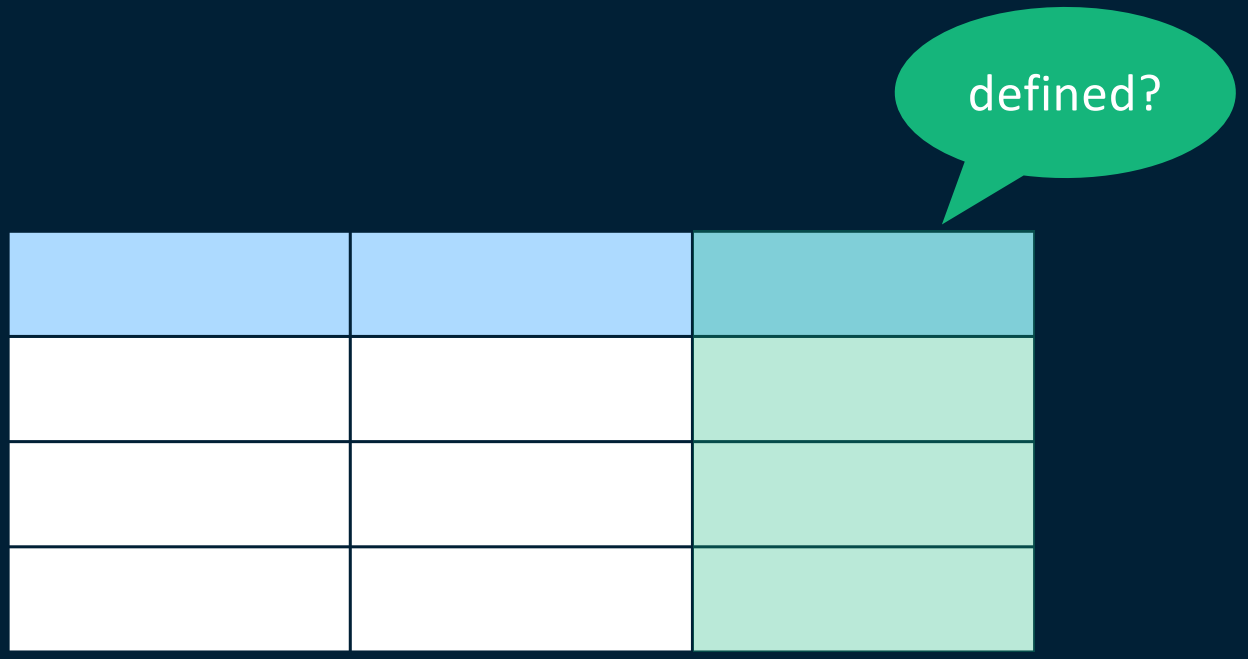

SAS table
descriptor portion
data portion
data values
• table name
• number of rows
• date created
• column names
• column attributes

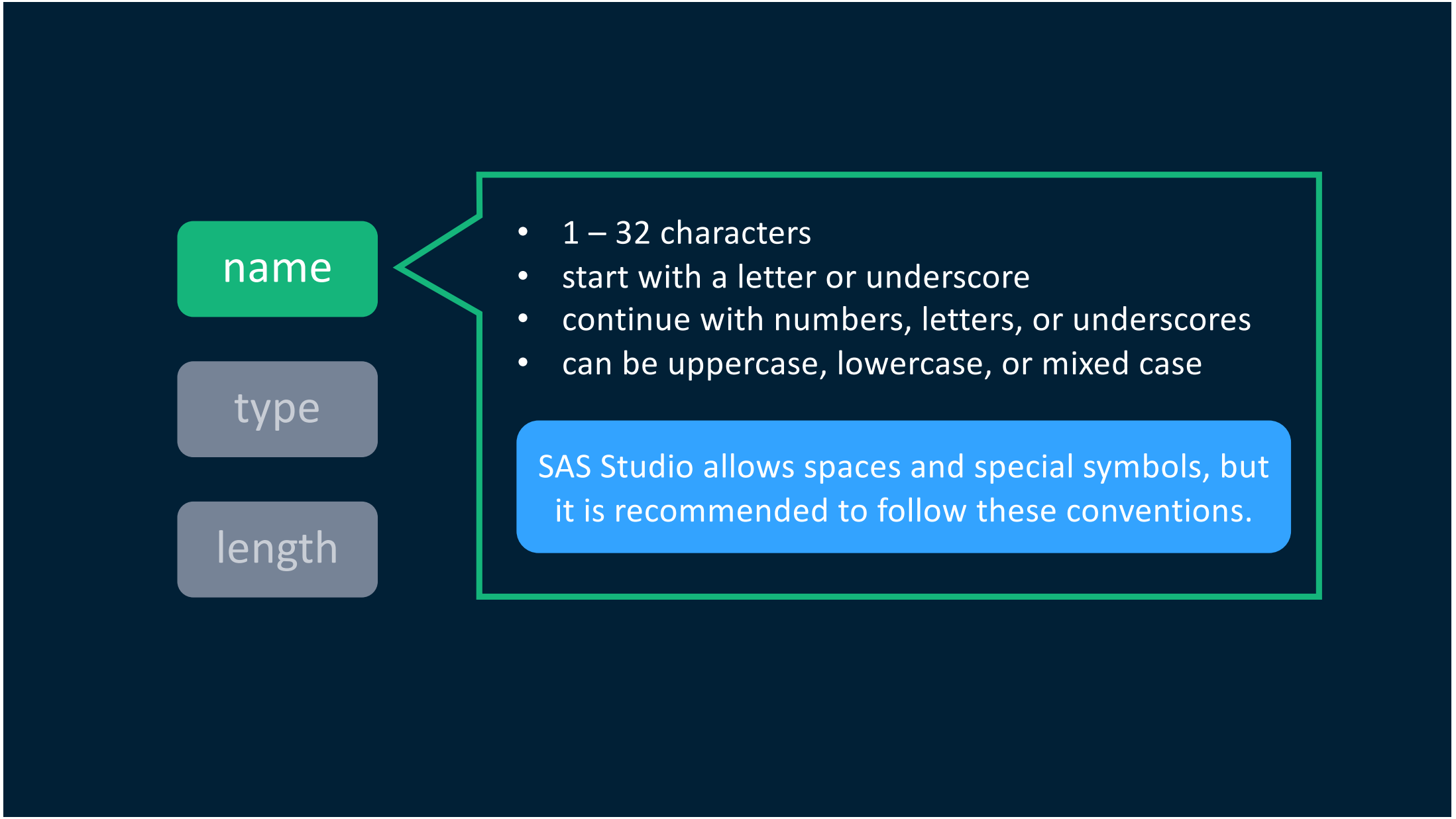
Required Column Attributes

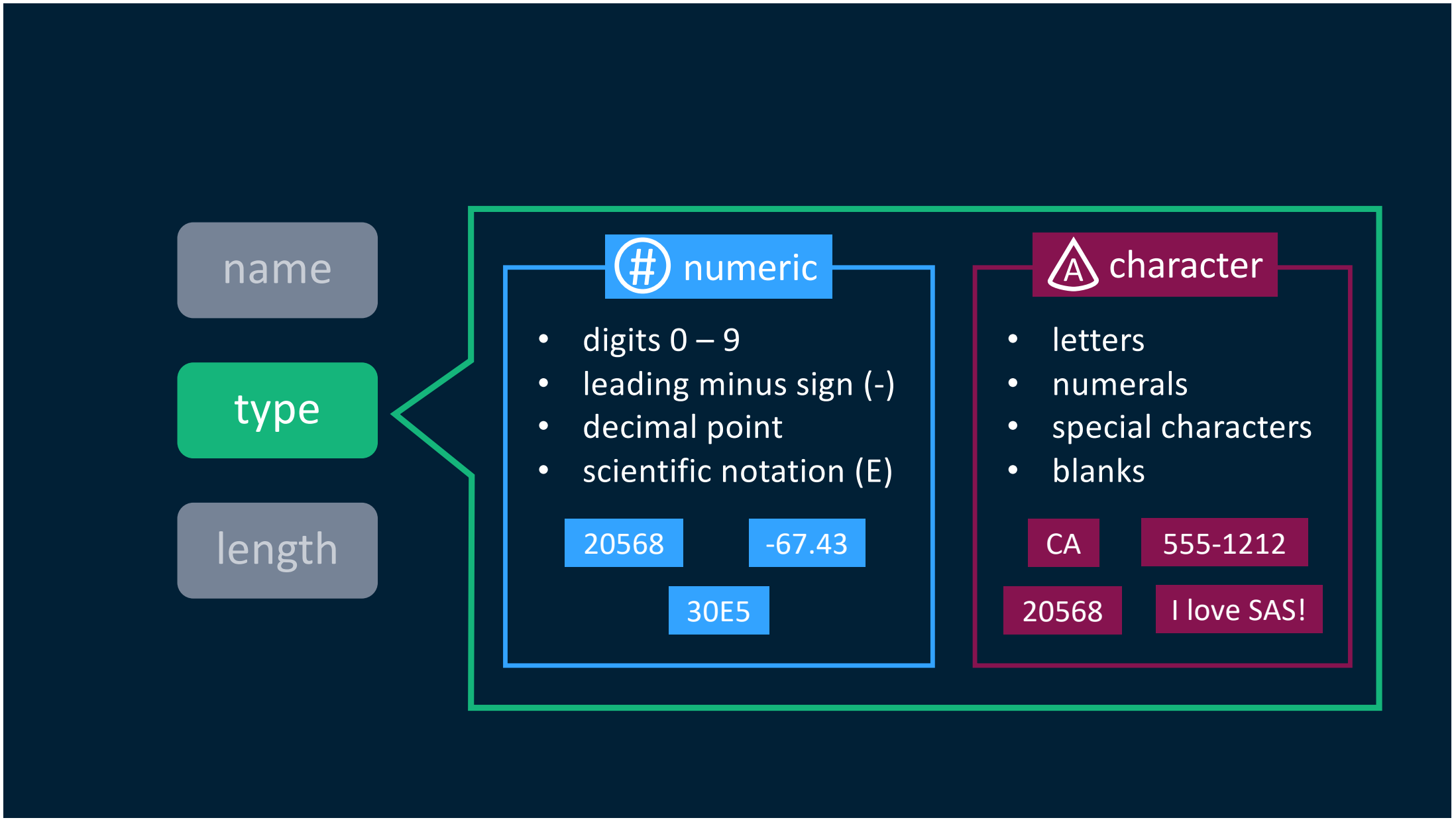
name

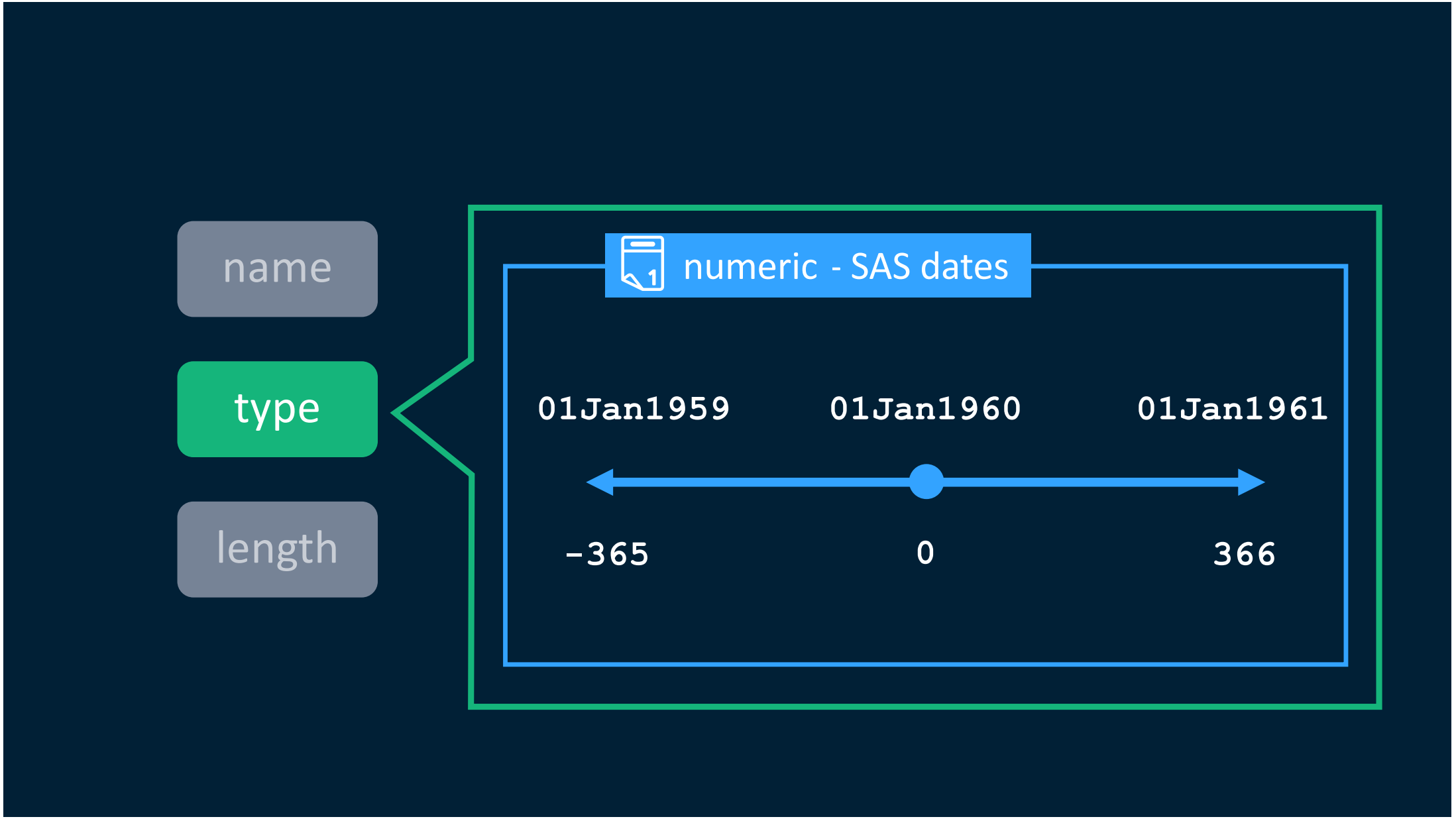
type

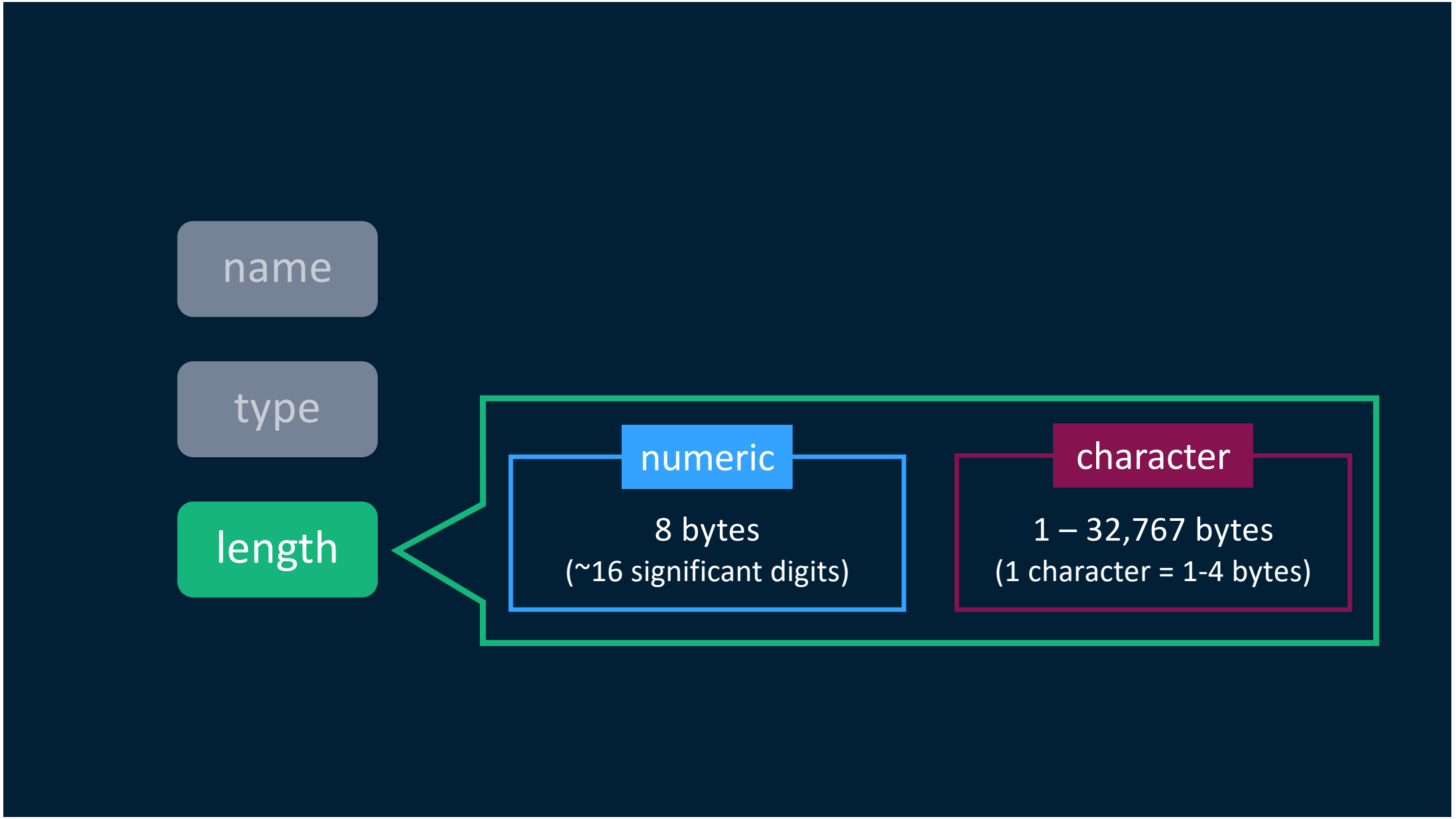
length

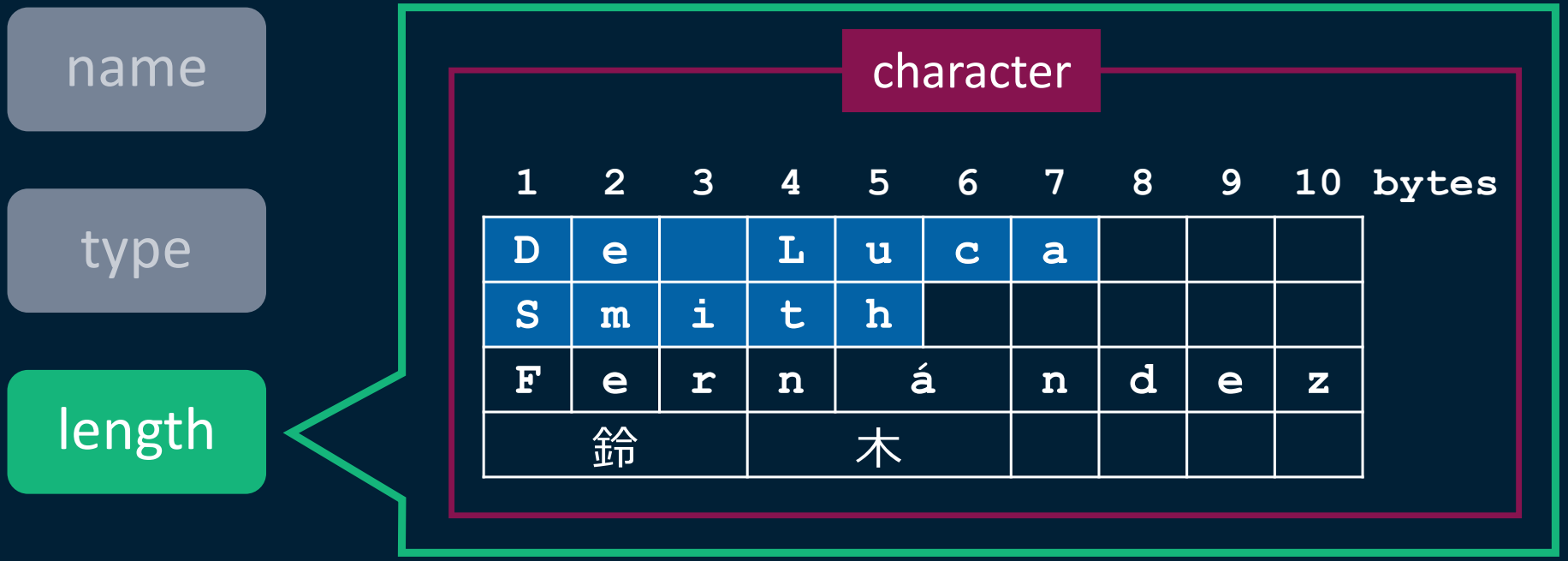


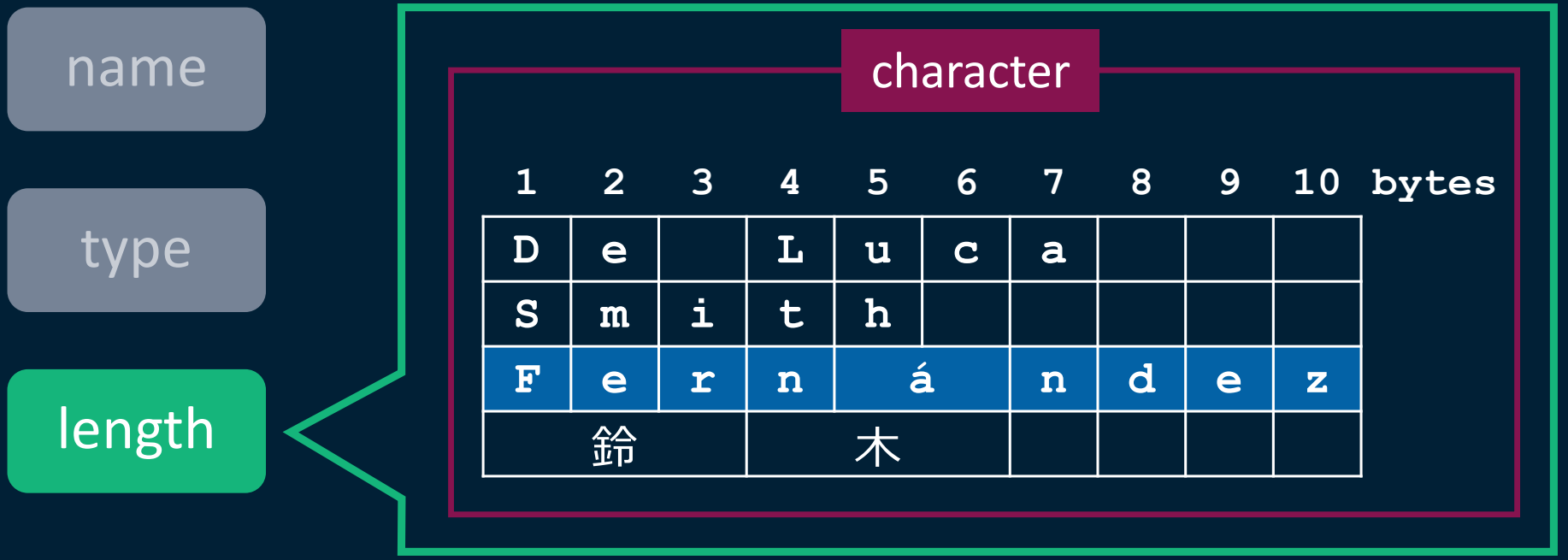

name
type
length
1 – 32 characters
start with a letter or underscore
continue with numbers, letters, or underscores
can be uppercase, lowercase, or mixed case
SAS Studio allows spaces and special symbols, but it is recommended to follow these conventions.

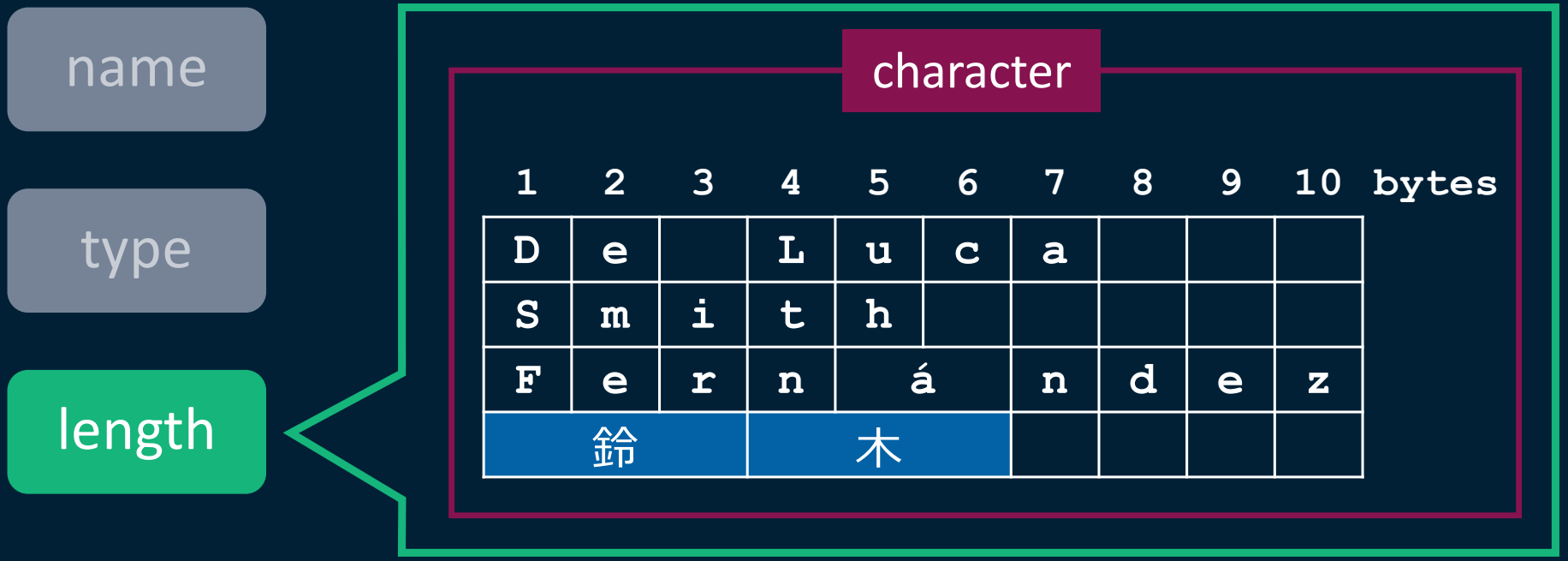

name
type
length
numeric
digits 0 – 9
leading minus sign (-)
decimal point
scientific notation (E)
20568
-67.43
30E5
character
letters
numerals
special characters
blanks
CA
555-1212
20568
I love SAS!

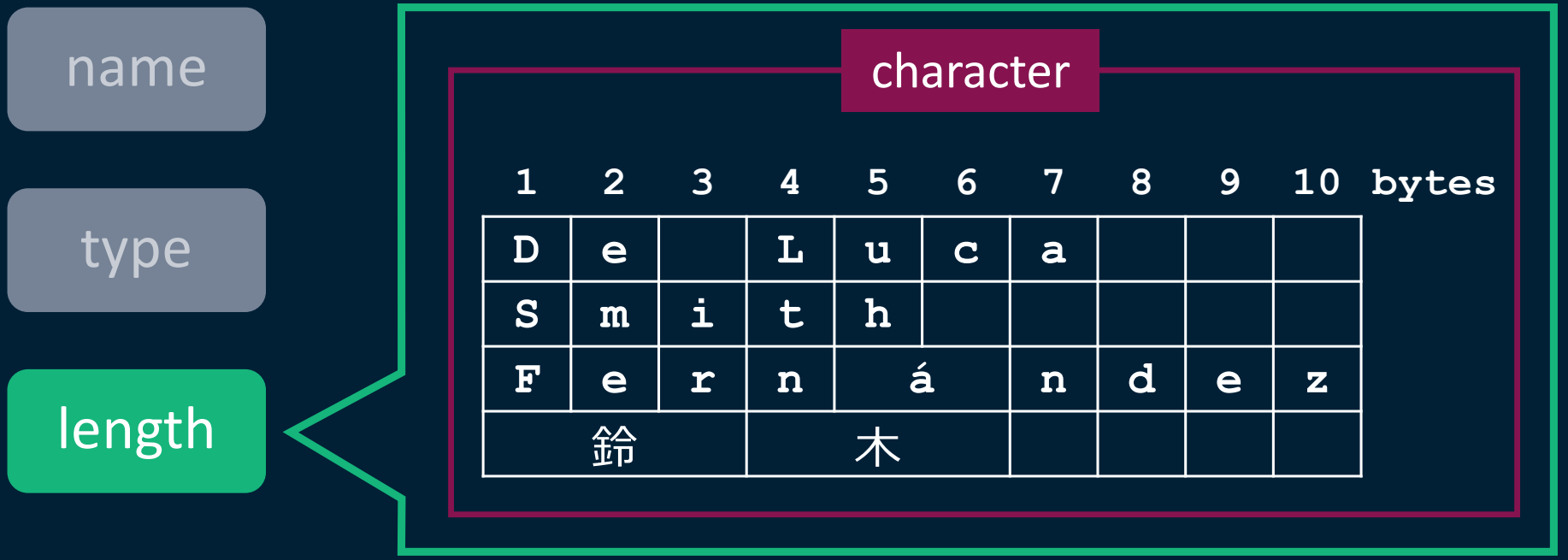

name
type
length
numeric - SAS dates
01Jan1959
01Jan1960
01Jan1961
-365
0
366


name
type
length
numeric
8 bytes
(~16 significant digits)
character
1 – 32,767 bytes
(1 character = 1-4 bytes)


name
type
length
character
1 2 3 4 5 6 7 8 9 10 bytes
D e L u c a
S m i t h
F e r n á n d e z
鈴 木


name
type
length
character
1 2 3 4 5 6 7 8 9 10 bytes
D e L u c a
S m i t h
F e r n á n d e z
鈴 木


name
type
length
character
1 2 3 4 5 6 7 8 9 10 bytes
D e L u c a
S m i t h
F e r n á n d e z
鈴 木


name
type
length
character
1 2 3 4 5 6 7 8 9 10 bytes
D e L u c a
S m i t h
F e r n á n d e z
鈴 木

Question – Character Value Storage

Name is defined as a character column with a length of 8. How is the value *François* stored in this column?

a. François
b. Françoi
c. Francois
d. a missing value

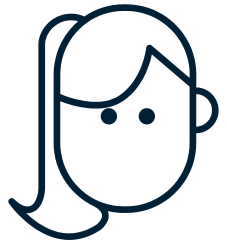
Answer – Character Value Storage

Name is defined as a character column with a length of 8. How is the value *François* stored in this column?

a. François
b. Françoi
c. Francois
d. a missing value

1	2	3	4	5	6	7	8
F	r	a	n	ç		o	i

Each value in **Name** is stored as 8 bytes. Because *ç* requires 2 bytes, the *s* is truncated.



Activity – Viewing Missing Values

1. In the Explorer section, expand **Folder Shortcuts** > **VSTU_data**.
2. Double-click the **employee_donations.sas7bdat** table to open it in the table viewer.

How are missing numeric and character values represented in the data?

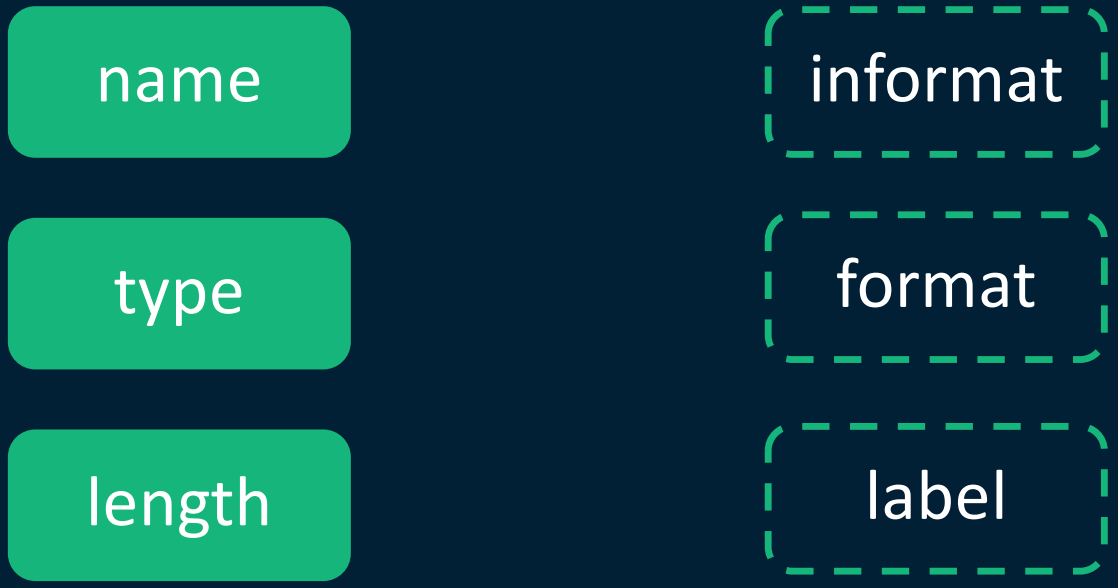
Answer – Viewing Missing Values

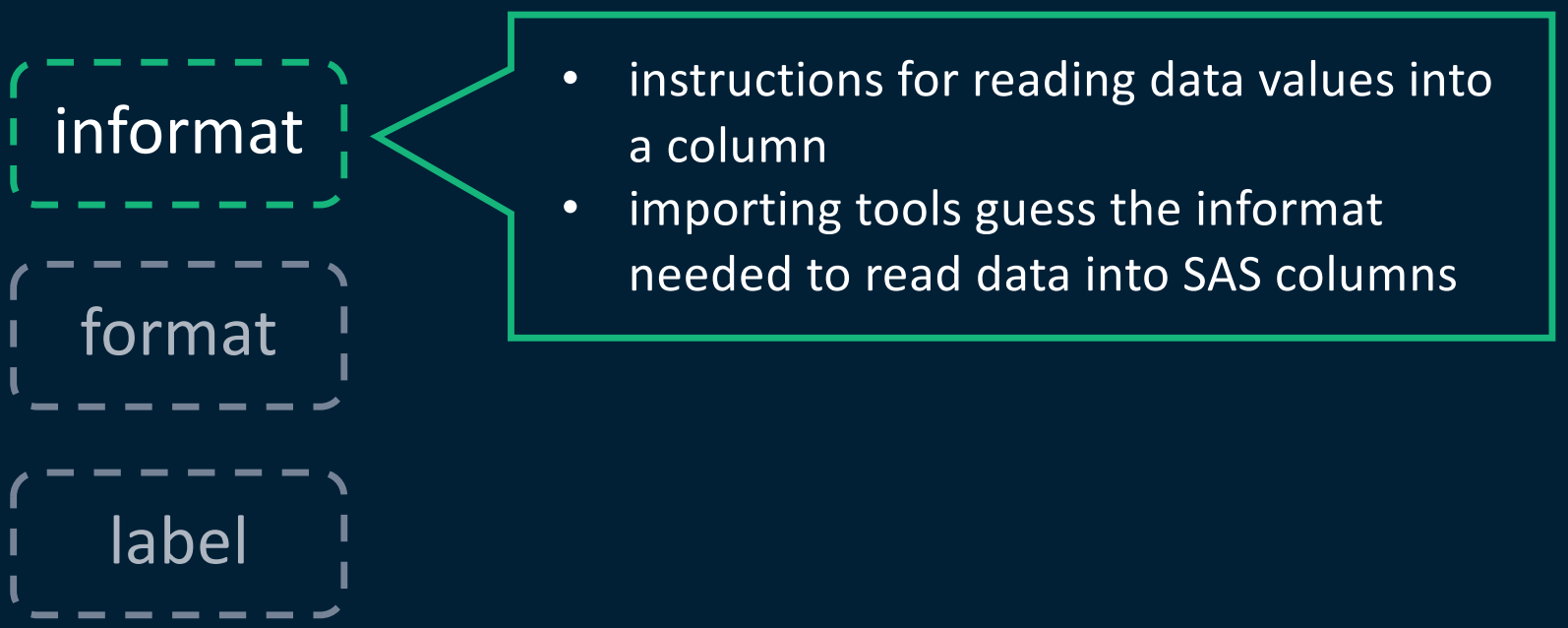
How are missing numeric and character values represented in the data?

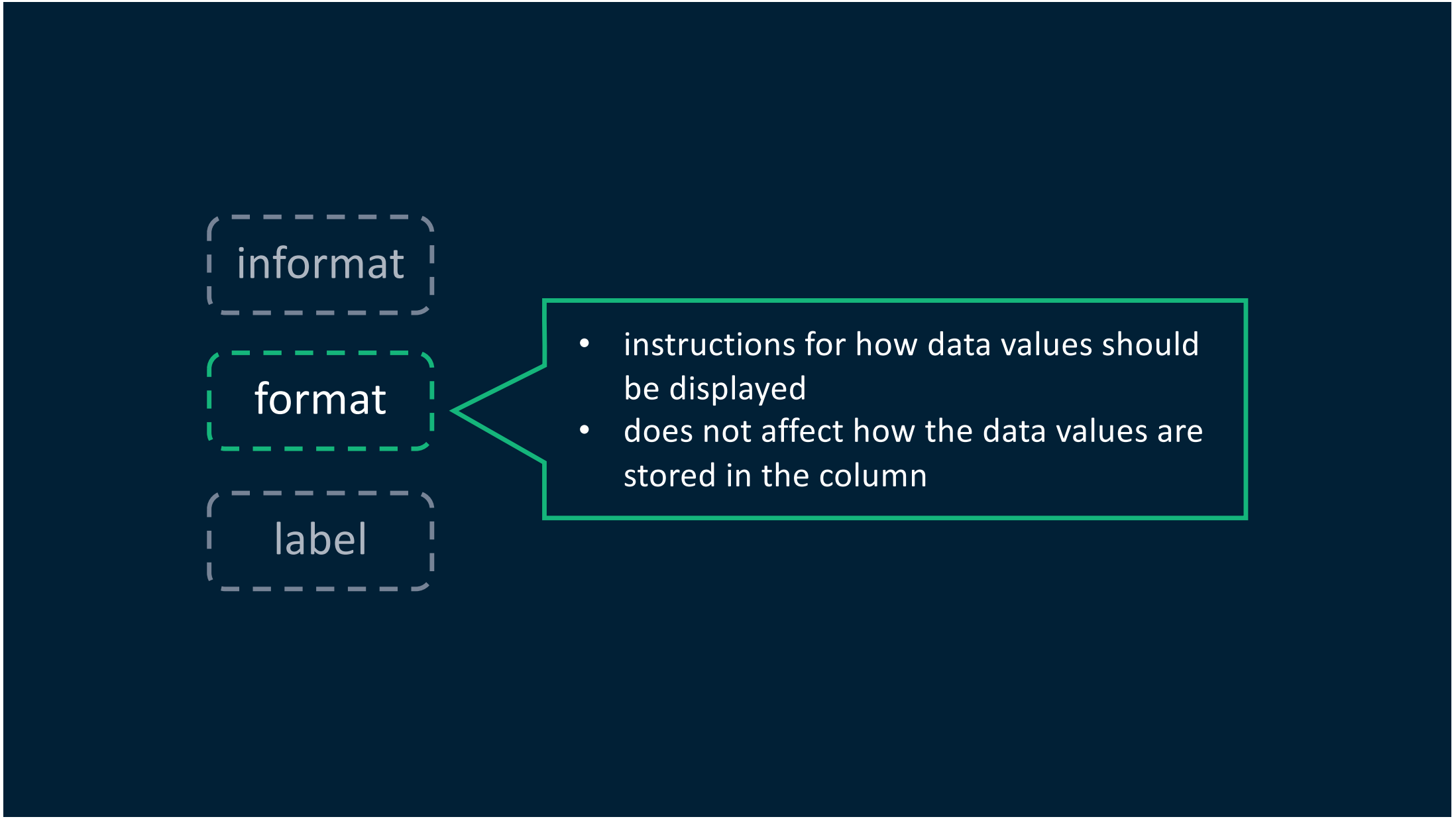
A period (.) for numeric values and a blank space for character values.

Qtr1	Qtr2	Qtr3	Qtr4	Recipients	Paid_By
.	.	.	25	Mitleid International 90%, Save the Baby Animals 10%	Cash or Check
15	15	15	15	Disaster Assist, Inc. 80%, Cancer Cures, Inc. 20%	Payroll Deduction
20	20	20	20	Cancer Cures, Inc. 10%, Cuidadores Ltd. 90%	Payroll Deduction
20	10	5	.	AquaMissions International 10%, Child Survivors 90%	
20	20	20	20	Cuidadores Ltd. 80%, Mitleid International 20%	Payroll Deduction
10	10	10	10	AquaMissions International 10%, Child Survivors 90%	Payroll Deduction

Optional Column Attributes




informat
format
label
• instructions for reading data values into a column
• importing tools guess the informat needed to read data into SAS columns


informat
format
label
• instructions for how data values should be displayed
• does not affect how the data values are stored in the column

format

orion_profit

Order_ID	Customer_Name	Order_Date	Quantity	Profit
1235183478	Lee Neppel	15MAR2017	1	$54.20
1230950247	Eulla Ivery	23JUN2015	2	$67.00
1241447239	Julia Yeo	15FEB2019	1	$4.75
1235156347	Reinhard Duisberg	13MAR2017	1	$56.80

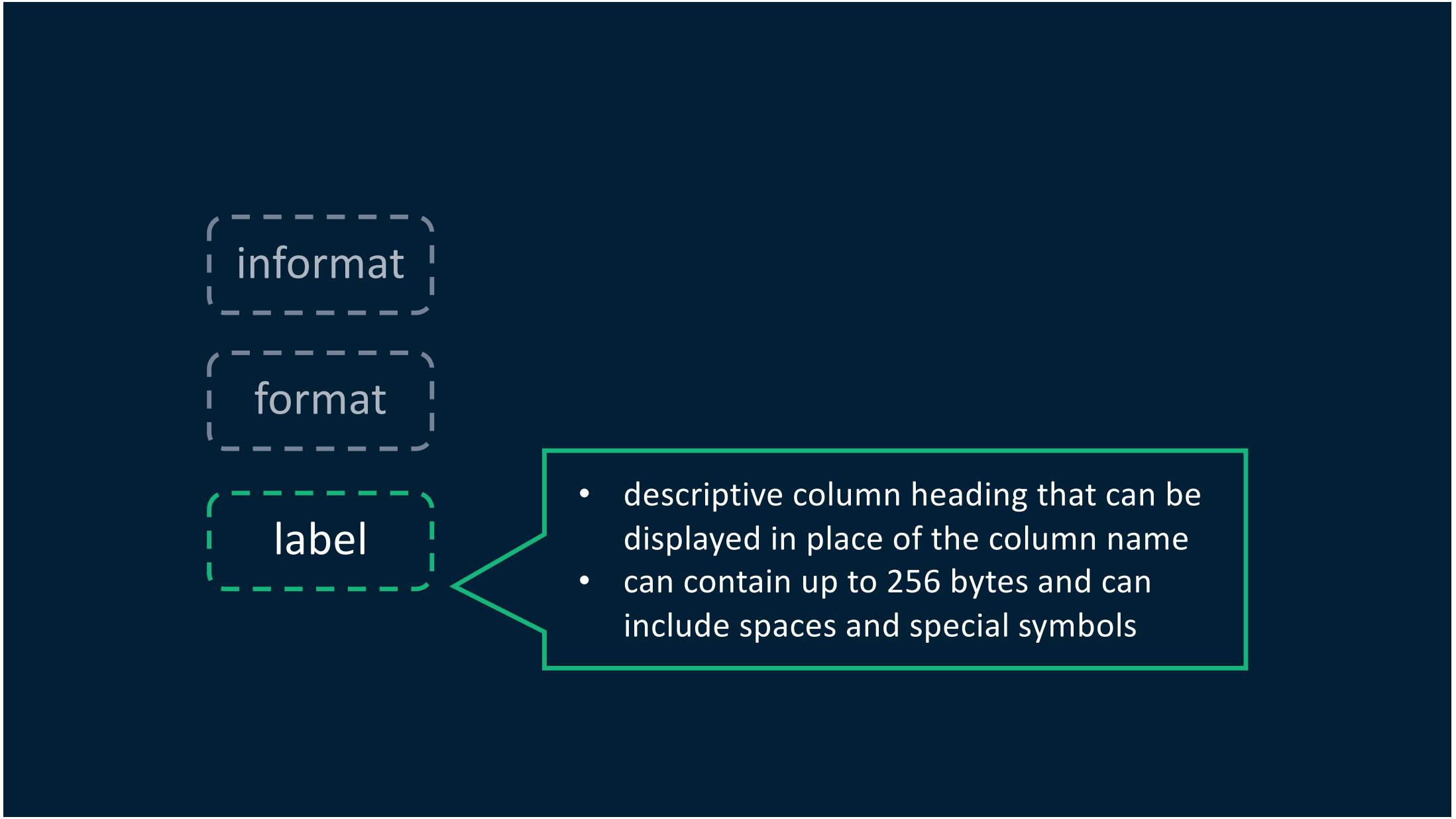
stored value:	20891
format:	DATE9.
format name:	DATE
width:	9

format

orion_profit

Order_ID	Customer_Name	Order_Date	Quantity	Profit
1235183478	Lee Neppel	15MAR2017	1	$54.20
1230950247	Eulla Ivery	23JUN2015	2	$67.00
1241447239	Julia Yeo	15FEB2019	1	$4.75
1235156347	Reinhard Duisberg	13MAR2017	1	$56.80

stored value: 56.8
format: DOLLAR10.2
format name: DOLLAR
width: 10
decimal places: 2


informat
format
label
• descriptive column heading that can be displayed in place of the column name
• can contain up to 256 bytes and can include spaces and special symbols

label

name: Order_Date
label: Order Placement Date

orion_profit

Order ID	Customer Name	Order Placement Date	Quantity Ordered	Profit
1235183478	Lee Neppel	15MAR2017	1	$54.20
1230950247	Eulla Ivery	23JUN2015	2	$67.00
1241447239	Julia Yeo	15FEB2019	1	$4.75
1235156347	Reinhard Duisberg	13MAR2017	1	$56.80

Activity – Exploring Column Attributes

1. In the Explorer section, expand **Folder Shortcuts** > **VSTU_data**.
2. Double-click the **employees.sas7bdat** table.
3. Examine the **Birth_Date**, **Hire_Date**, and **Termination** columns at the end of the table.
4. On the far right of the table viewer toolbar, click **More options** (the three vertical dots) and then select **Table properties**.
5. In the Table Properties window, click the **Column Properties** tab and examine the attributes of the **Birth_Date**, **Hire_Date**, and **Termination** columns.

Why is the **Termination** column displayed as a number?

Answer – Exploring Column Attributes

Why is the **Termination** column displayed as a number?

The Termination column does not have a format applied.

Column Name	Label	Type	Length	Format	Informat
Employee_ID	Employee ID	Numeric	8	12.	
Employee_Name	Employee Name	Character	60		
Country	Country	Character	3	$3.	
Department	Department	Character	60		
Job_Title	Job Title	Character	38		
Remote		Character	2		
Manager_ID	Manager ID	Numeric	8		
Salary	Annual Salary	Numeric	8	DOLLAR13.	
Birth_Date	Birth Date	Numeric	8	DATE9.	
Hire_Date	Employee Hire Date	Numeric	8	MMDDYY8.	DATE9.
Termination	Termination Date	Numeric	8		

Activity – Adding SAS Tables to a Flow

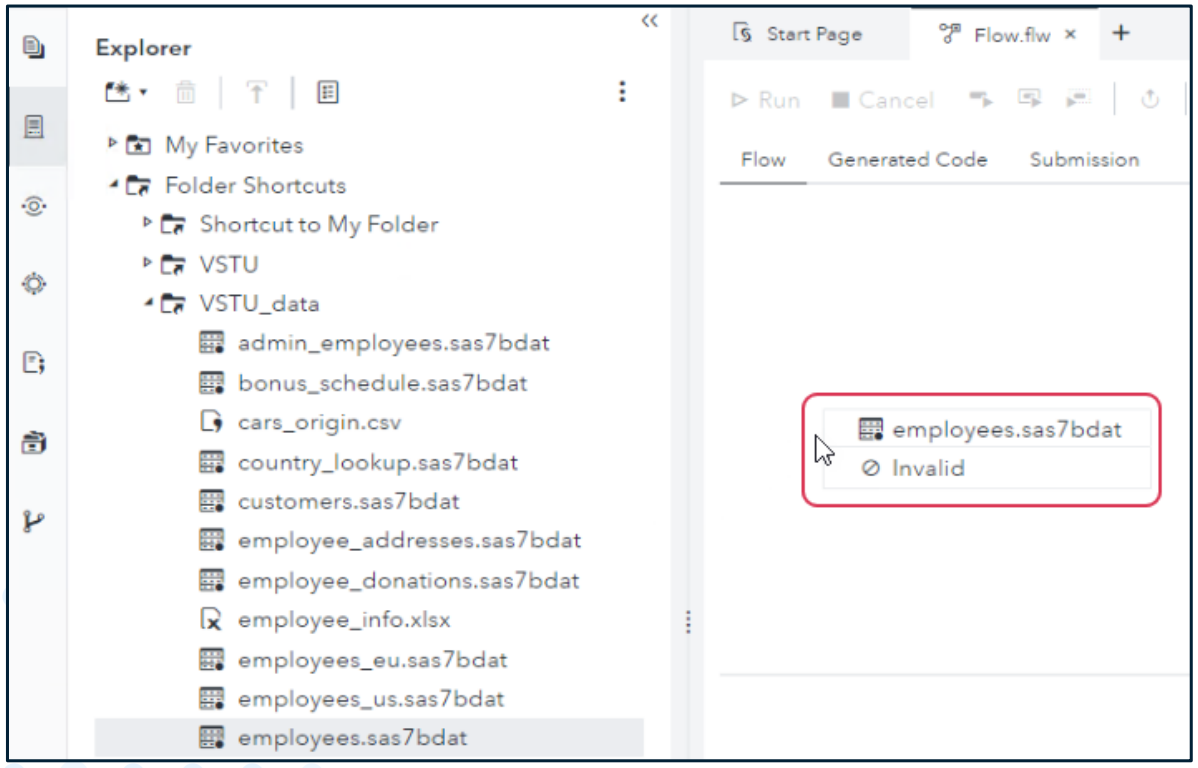
1. On the main SAS Studio toolbar, select **New** > **Flow**.
2. In the Explorer section, expand **Folder Shortcuts** > **VSTU_data**.
3. Drag the **employees.sas7bdat** table onto the flow canvas.

Does SAS Studio let you add the table to the flow?

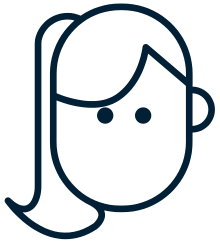
Answer – Adding SAS Tables to a Flow

Does SAS Studio let you add the table to the flow?

No, SAS Studio displays an Invalid message.



A SAS table must be accessed through a SAS library to be used in a flow.



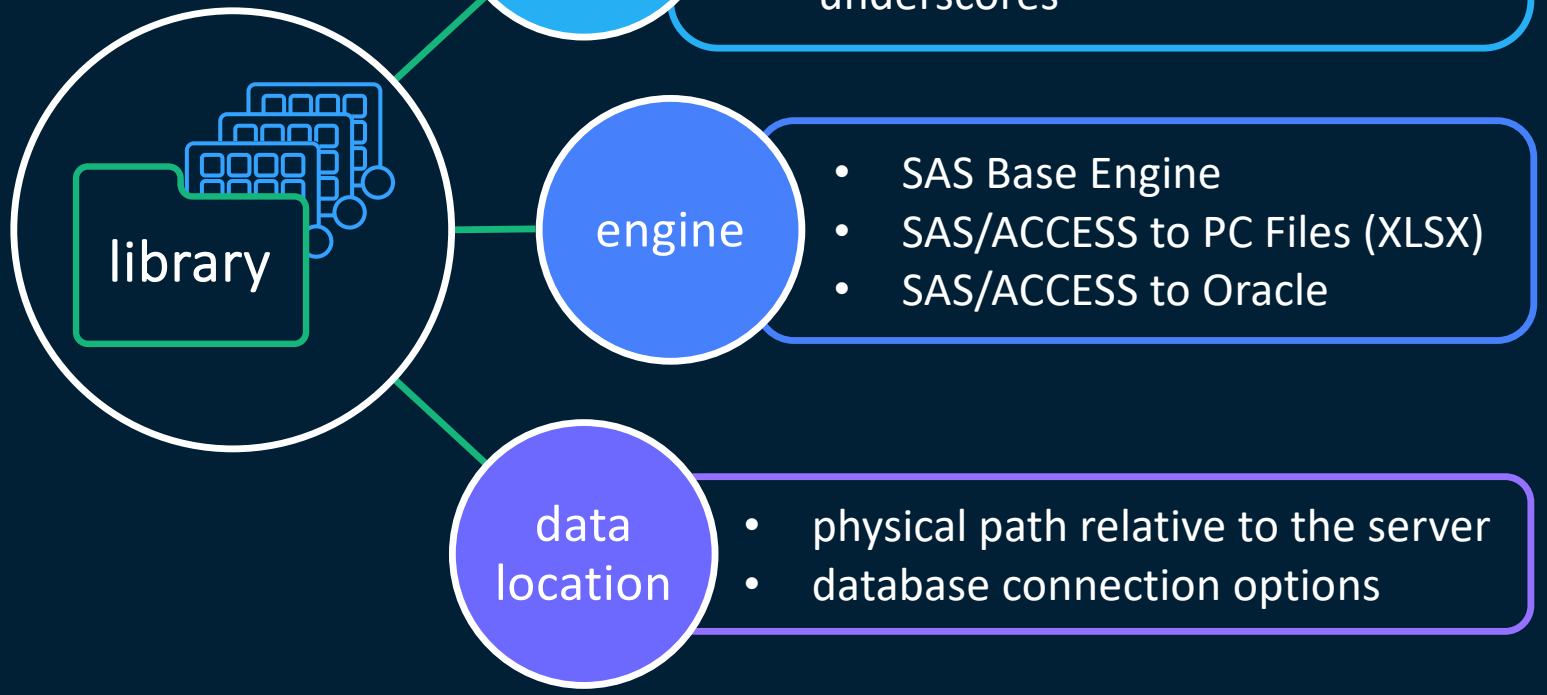
Lesson 2: Accessing Data

2.1 Understanding SAS Data

2.2 Accessing Data through Libraries

2.3 Importing Data

Accessing Data through Libraries



Assigning a SAS Library

This demonstration illustrates how to assign a library to access SAS tables.

Activity – Assigning the Orion Library (Required)

1. In the Explorer section, expand **Folder Shortcuts**.
2. Right-click the **VSTU_data** folder shortcut and select **Create library**.
3. In the **Name** field, enter **orion**, and select the **Assign and connect to data sources at startup** option. Click **OK**.
4. In the Libraries section, verify that the **ORION** library appears.
5. On the main SAS Studio toolbar, select **Options** > **Reset SAS session** > **Reset**.

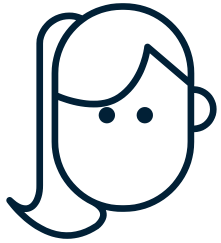
Does the **orion** library appear in the Libraries section?

Answer – Assigning the Orion Library (Required)

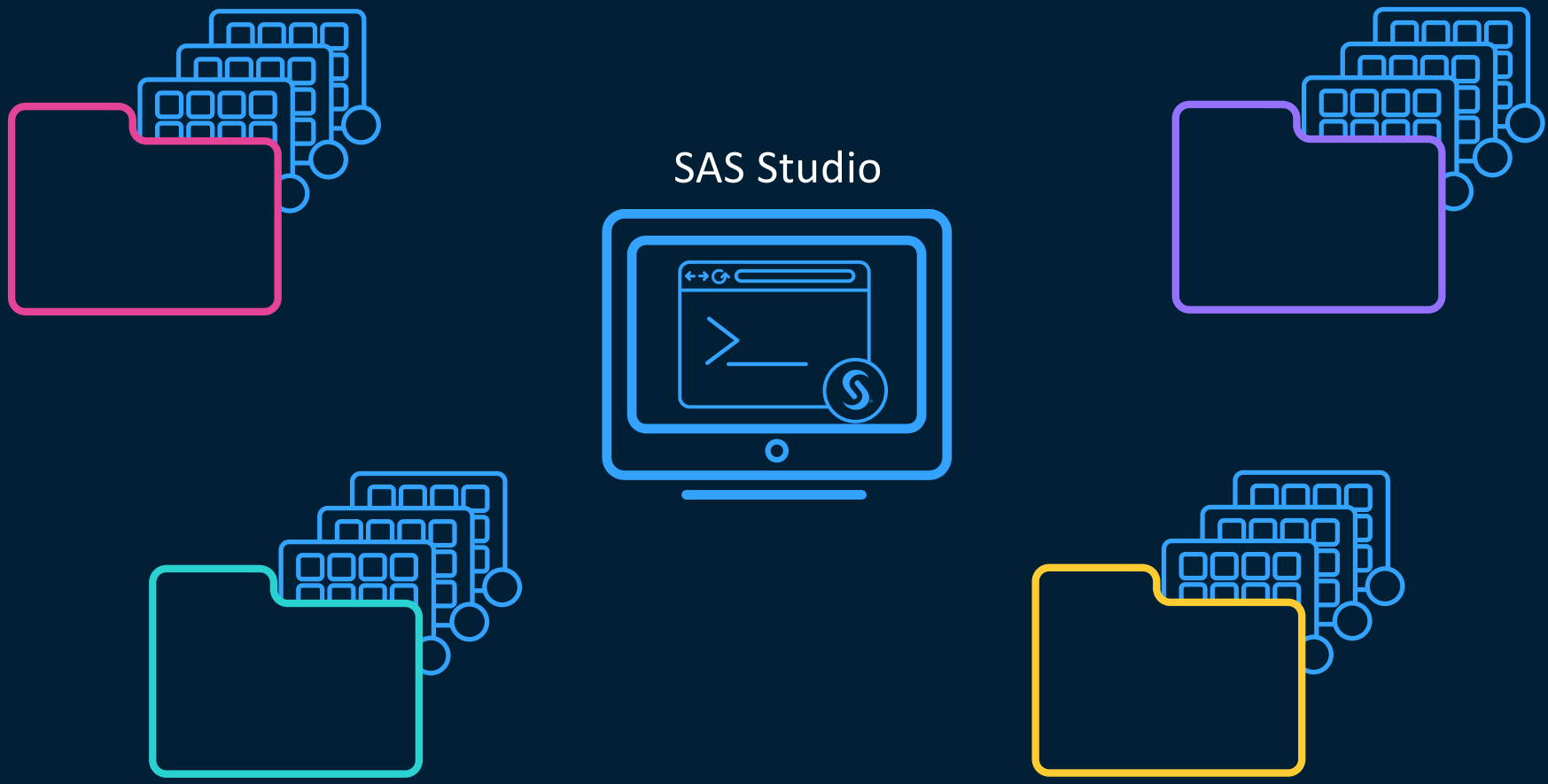
Does the **orion** library appear in the Libraries section?

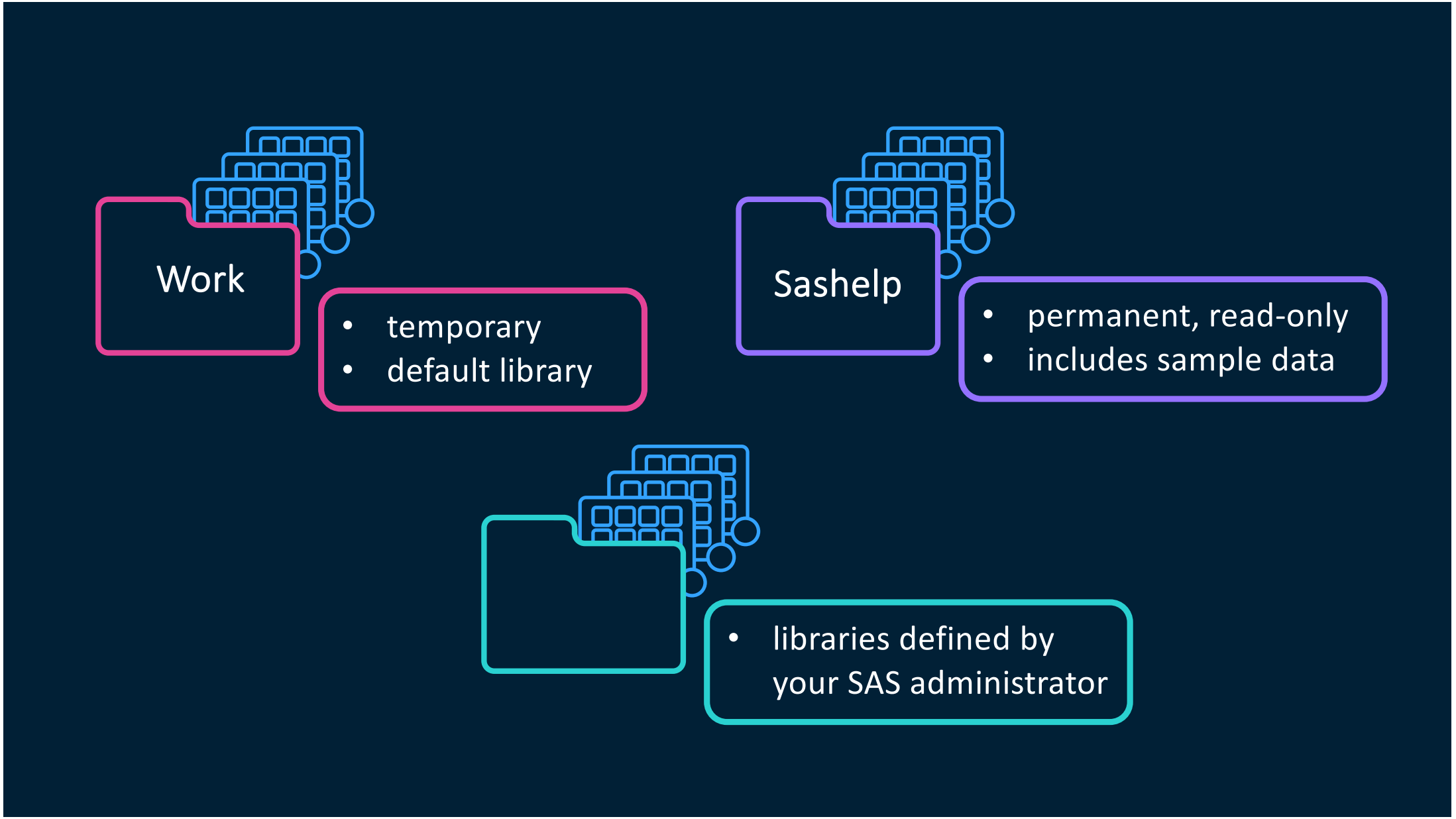
Yes.

The **Assign and connect to data sources at startup** option adds the generated code to an autoexec file. This file is executed each time a SAS session starts.



Automatic SAS Libraries




Work
• temporary
• default library
Sashelp
• permanent, read-only
• includes sample data
• libraries defined by your SAS administrator

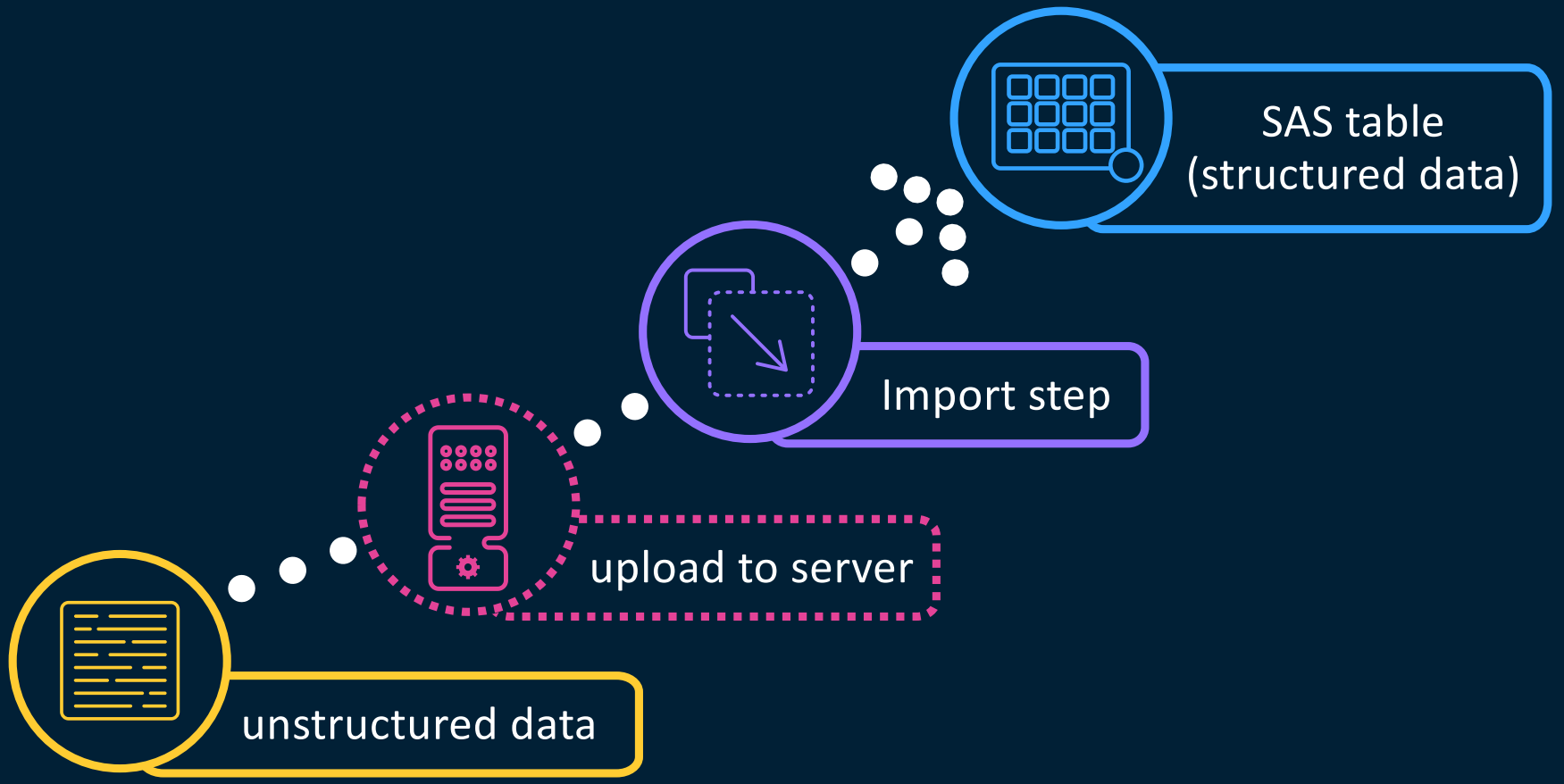
Lesson 2: Accessing Data

2.1 Understanding SAS Data

2.2 Accessing Data through Libraries

2.3 Importing Data

Importing Data in a Flow



Importing a CSV File

This demonstration illustrates how to use the Import step in a flow to import a CSV file.

Lesson 3: Transforming and Analyzing Data

3.1 Creating Simple Queries

3.2 Using SAS Studio Analyst Steps

3.3 Creating Results with Tasks

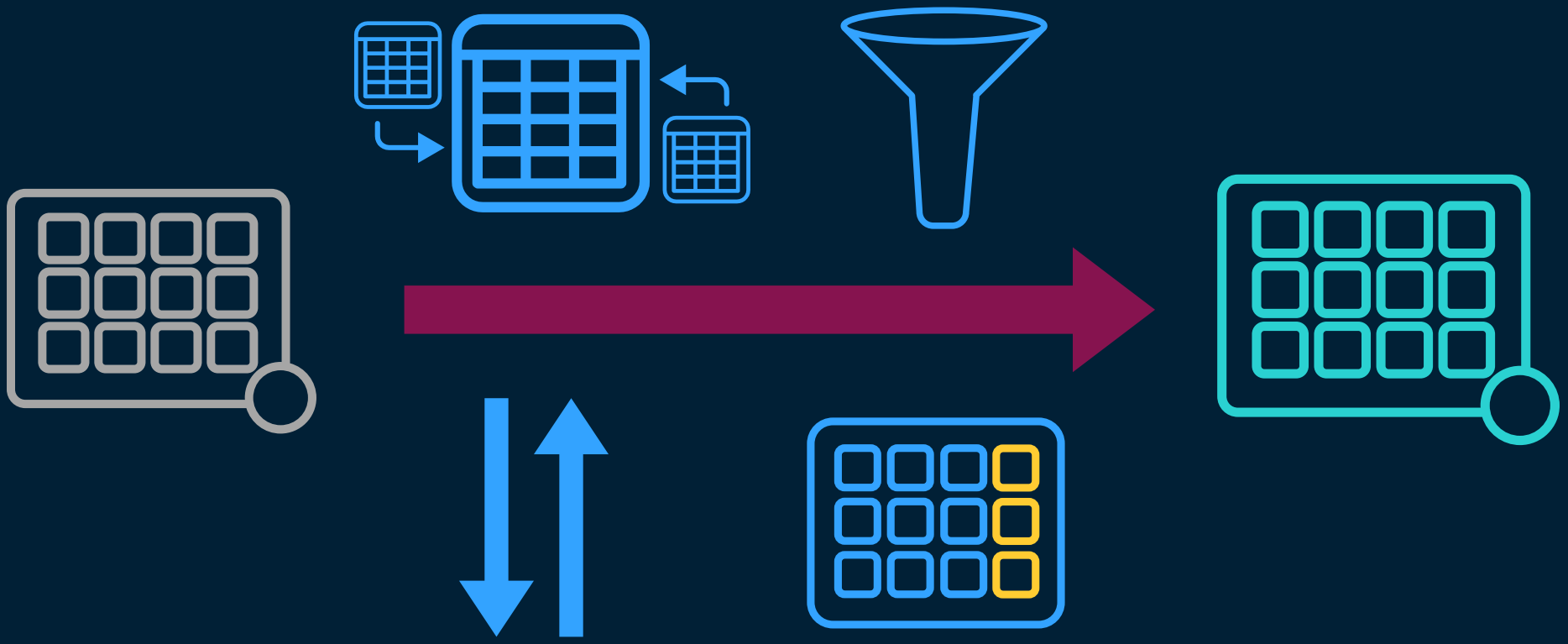
Lesson 3: Transforming and Analyzing Data

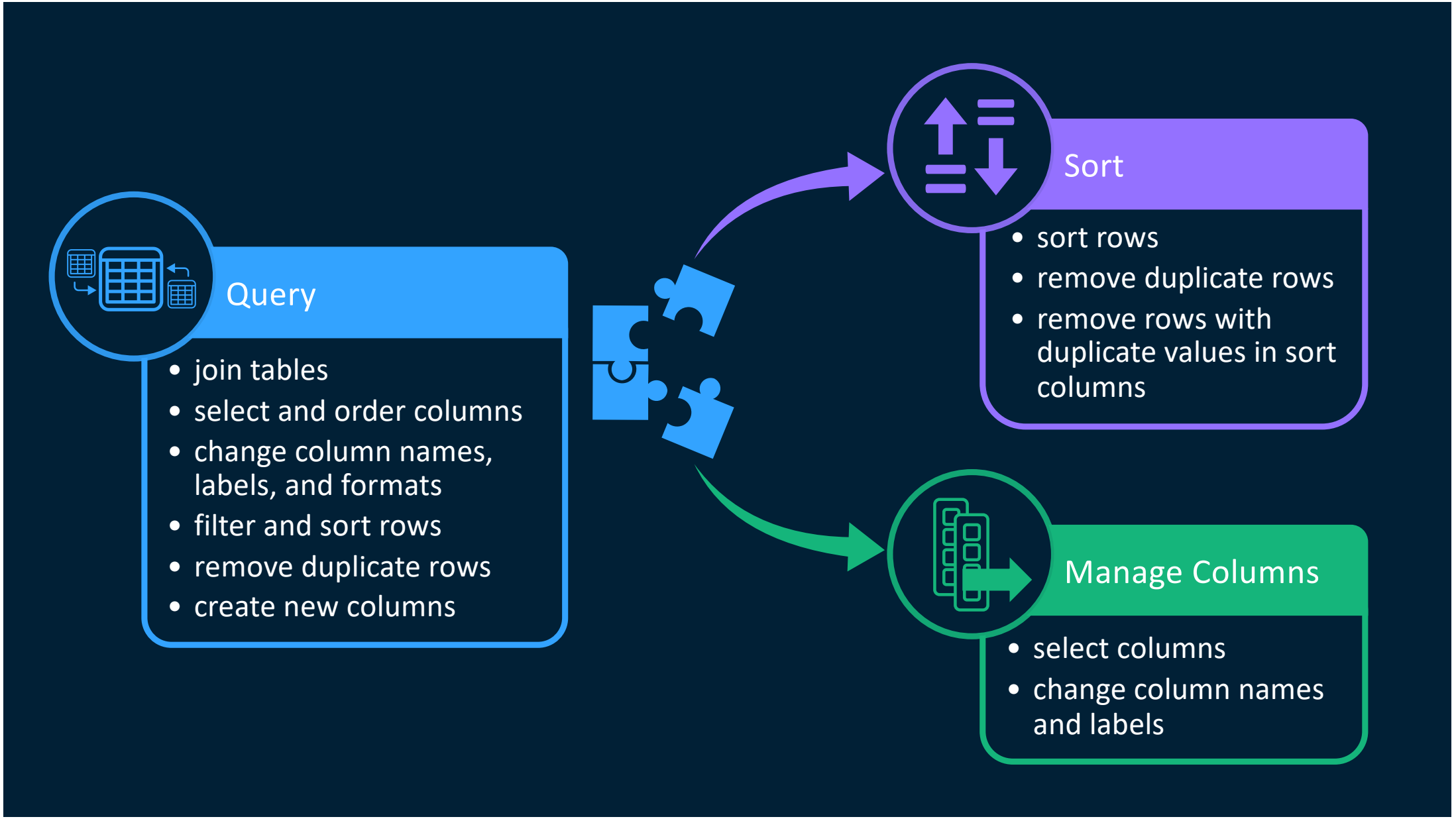
3.1 Creating Simple Queries

3.2 Using SAS Studio Analyst Steps

3.3 Creating Results with Tasks

Data Transformation Steps




Query
join tables
select and order columns
change column names, labels, and formats
filter and sort rows
remove duplicate rows
create new columns
Sort
sort rows
remove duplicate rows
remove rows with duplicate values in sort columns
Manage Columns
select columns
change column names and labels

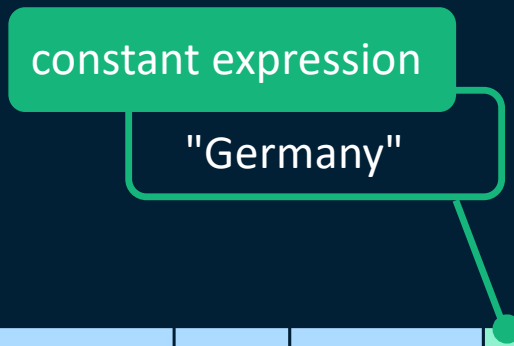
Creating New Columns

Customer	Order_Date	Sales	Shipping	Ctry
Martha Ganschow	05JUN2019	$234.40	$23.44	DE
Susanne Arens	05JUN2019	$35.98	$8.95	de
Katharina Pelda	06JUN2019	$78.20	$10.95	DE
Josef Blind	06JUN2019	$43.98	$8.95	DE

arithmetic expression

Sales + Shipping

Customer	Order_Date	Sales	Shipping	Ctry	Invoice	
Martha Ganschow	05JUN2019	$234.40	$23.44	DE	$257.84	
Susanne Arens	05JUN2019	$35.98	$8.95	de	$44.93	
Katharina Pelda	06JUN2019	$78.20	$10.95	DE	$89.15	
Josef Blind	06JUN2019	$43.98	$8.95	DE	$52.93	



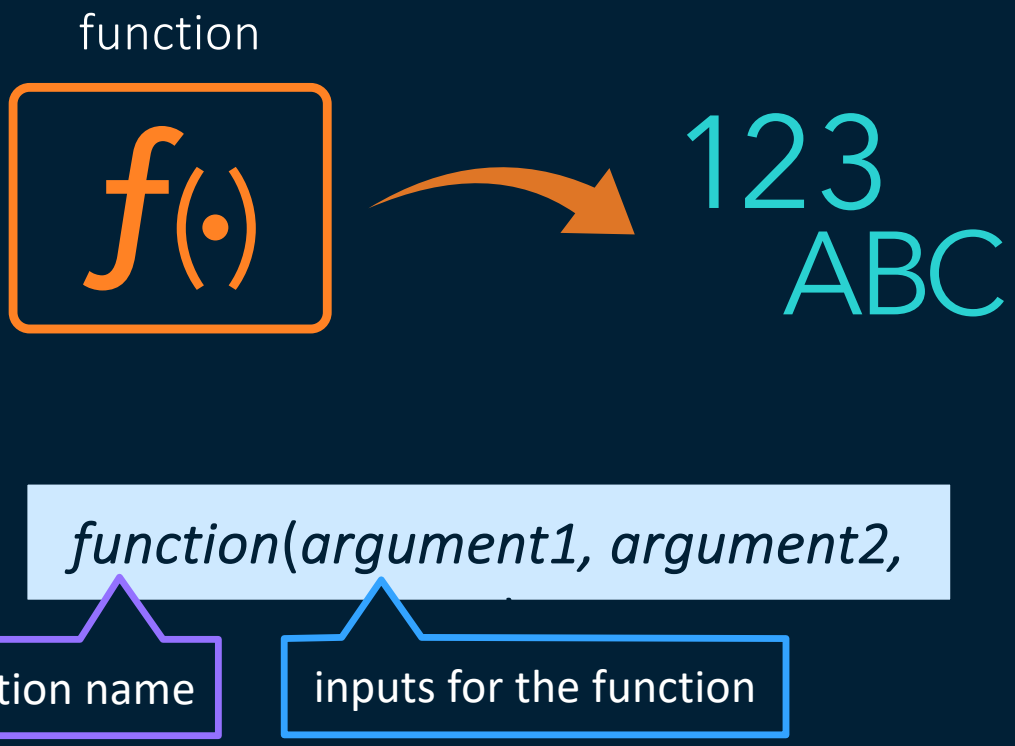
Customer	Order_Date	Sales	Shipping	Ctry	Invoice	Ctry_Full
Martha Ganschow	05JUN2019	$234.40	$23.44	DE	$257.84	Germany
Susanne Arens	05JUN2019	$35.98	$8.95	de	$44.93	Germany
Katharina Pelda	06JUN2019	$78.20	$10.95	DE	$89.15	Germany
Josef Blind	06JUN2019	$43.98	$8.95	DE	$52.93	Germany

function

extract month of order?

uppercase **Ctry** values?

Customer	Order_Date	Sales	Shipping	Ctry	Invoice	Ctry_Full
Martha Ganschow	05JUN2019	$234.40	$23.44	DE	$257.84	Germany
Susanne Arens	05JUN2019	$35.98	$8.95	de	$44.93	Germany
Katharina Pelda	06JUN2019	$78.20	$10.95	DE	$89.15	Germany
Josef Blind	06JUN2019	$43.98	$8.95	DE	$52.93	Germany


function
f(·)
123
ABC
function(argument1, argument2,
function name
inputs for the function

function

character function

upcase(Ctry)

Customer	Order_Date	Sales	Shipping	Ctry	Ctry_Up
Martha Ganschow	05JUN2019	$234.40	$23.44	DE	DE
Susanne Arens	05JUN2019	$35.98	$8.95	de	DE
Katharina Pelda	06JUN2019	$78.20	$10.95	DE	DE
Josef Blind	06JUN2019	$43.98	$8.95	DE	DE

function

date function

month(Order_Date)

Customer	Order_Date	Sales	Shipping	Ctry	Ctry_Up	Month
Martha Ganschow	05JUN2019	$234.40	$23.44	DE	DE	6
Susanne Arens	05JUN2019	$35.98	$8.95	de	DE	6
Katharina Pelda	06JUN2019	$78.20	$10.95	DE	DE	6
Josef Blind	06JUN2019	$43.98	$8.95	DE	DE	6

Activity – Viewing Documentation for SAS Functions

1. Go to the General Functions Listed Alphabetically page in the SAS Viya Programming documentation.
2. Select **T** at the top of the list and select any occurrence of the TODAY function.

How many arguments are required for the TODAY function?

Answer – Viewing Documentation for SAS Functions

How many arguments are required for the TODAY function?

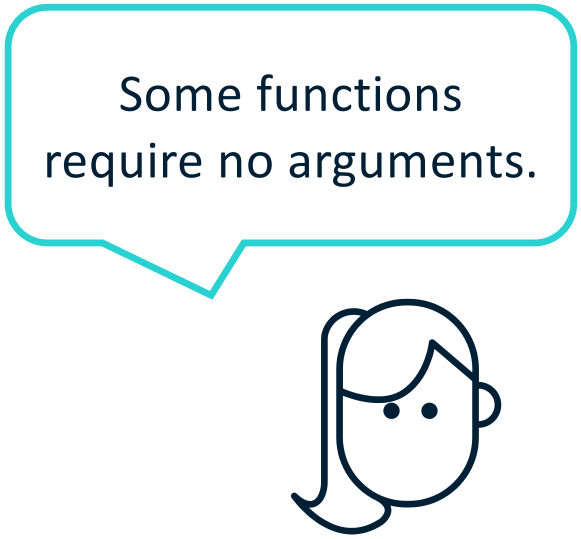
None.

Syntax

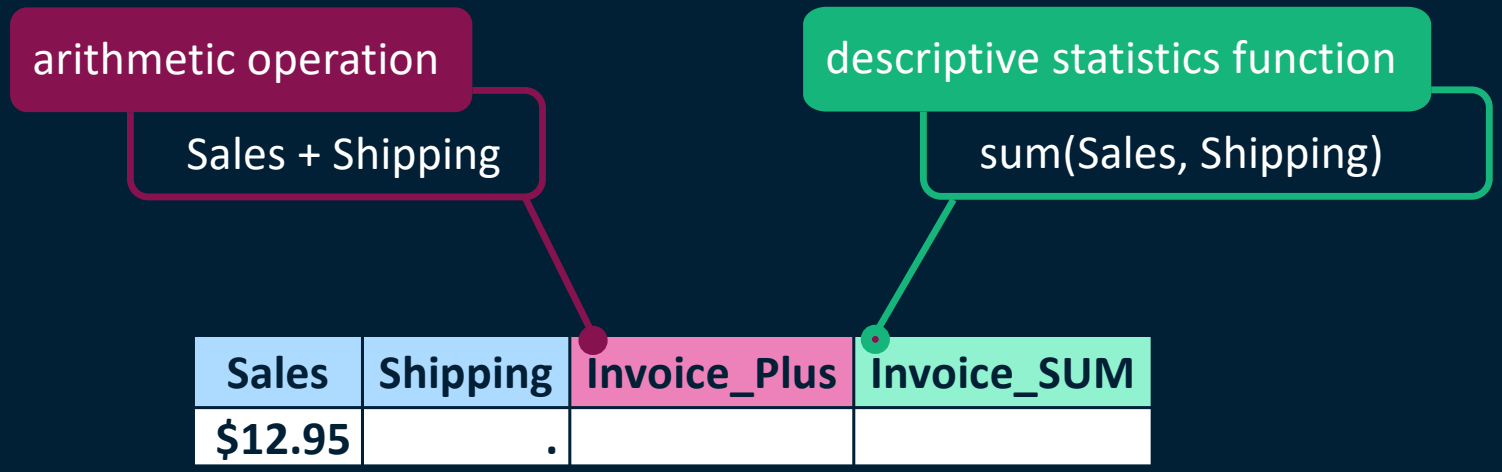
TODAY()

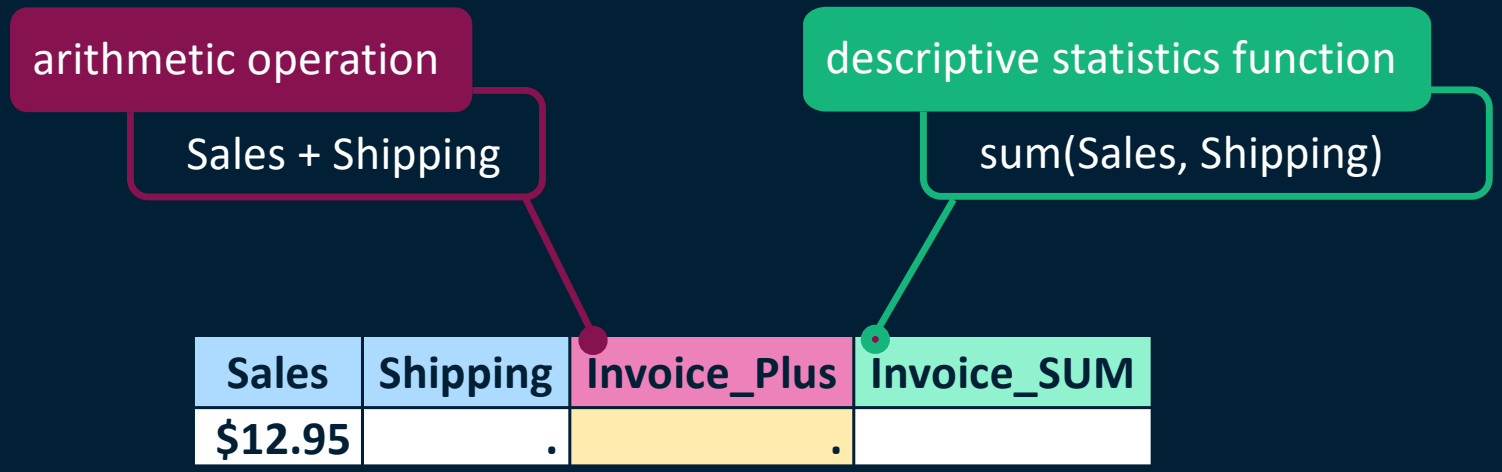
Details

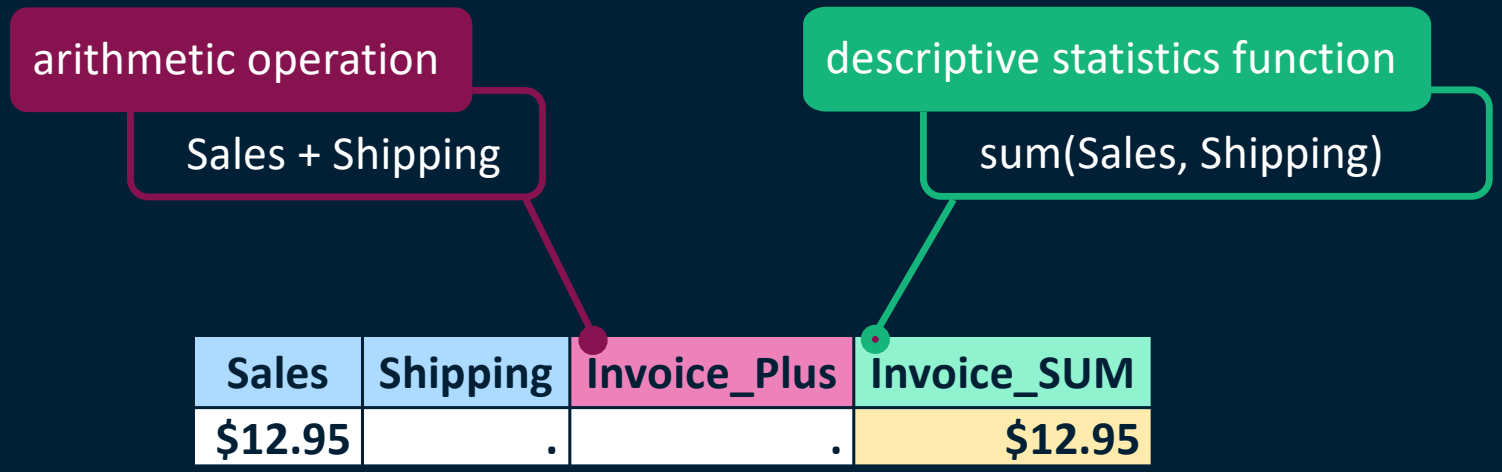
The TODAY function does not take any arguments.



Missing Values in Calculations




arithmetic operation
Sales + Shipping
descriptive statistics function
sum(Sales, Shipping)
Sales
Shipping
Invoice_Plus
Invoice_SUM
$12.95
.
.


arithmetic operation
Sales + Shipping
descriptive statistics function
sum(Sales, Shipping)
Sales
Shipping
Invoice_Plus
Invoice_SUM
$12.95
.
.
$12.95

Filtering Rows and Creating New Columns

This demonstration illustrates how to use the Query step in a flow to select columns, create new columns, and filter rows.

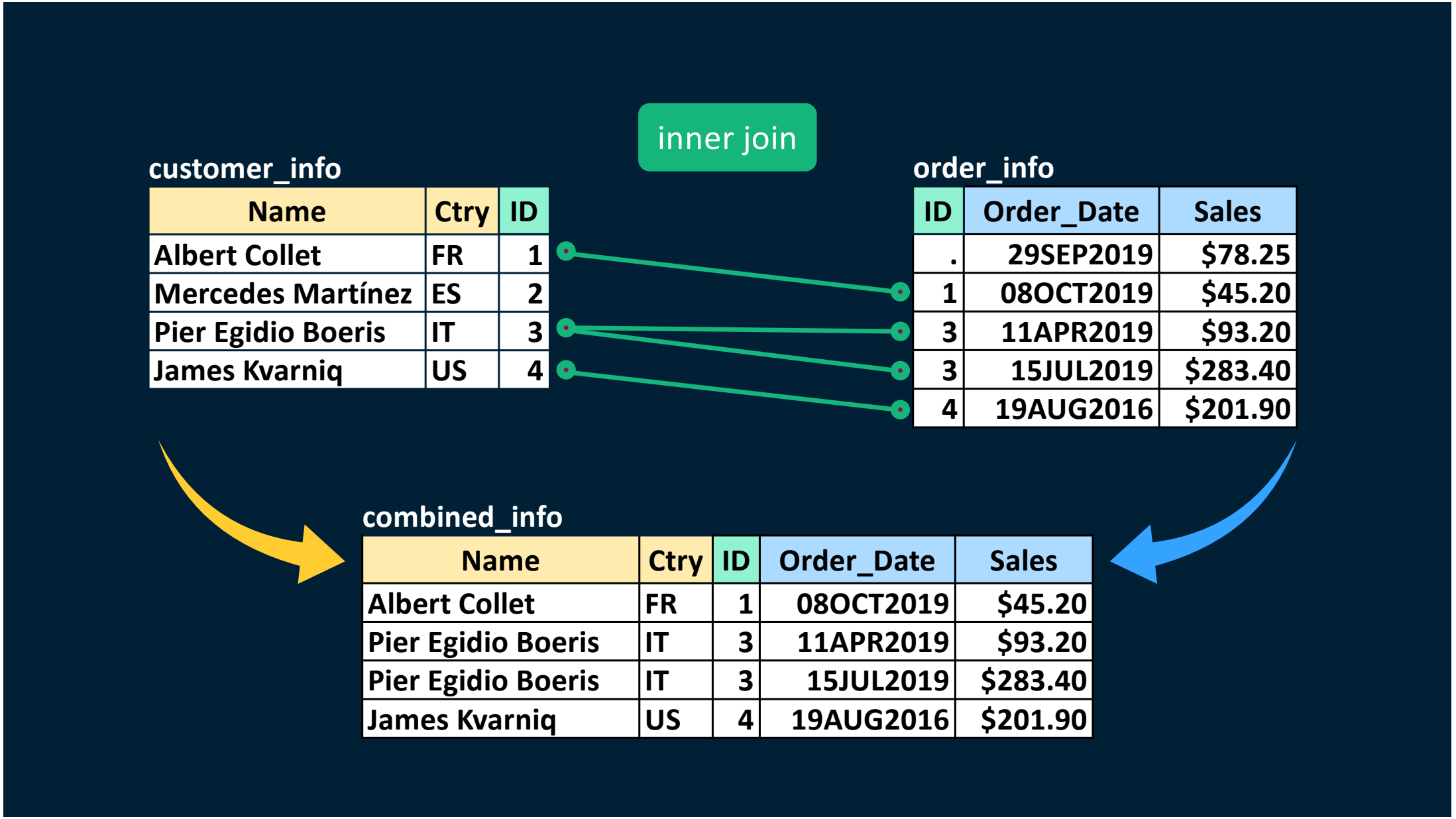
Joining Tables in a Query Step

customer_info

Name	Ctry	ID
Albert Collet	FR	1
Mercedes Martínez	ES	2
Pier Egidio Boeris	IT	3
James Kvarniq	US	4

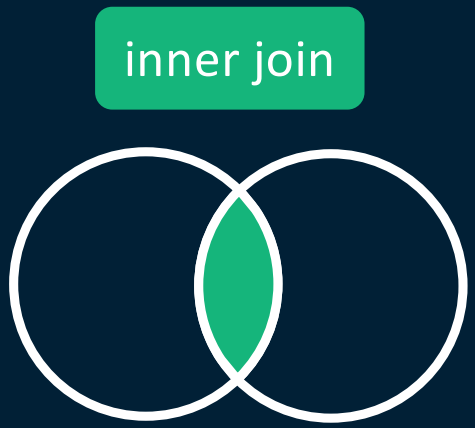
order_info

ID	Order_Date	Sales
.	29SEP2019	$78.25
1	08OCT2019	$45.20
3	11APR2019	$93.20
3	15JUL2019	$283.40
4	19AUG2016	$201.90


inner join
customer_info
Name
Ctry
ID
Albert Collet
FR
1
Mercedes Martínez
ES
2
Pier Egidio Boeris
IT
3
James Kvarniq
US
4
order_info
ID
Order_Date
Sales
.
29SEP2019
$78.25
1
08OCT2019
$45.20
3
11APR2019
$93.20
3
15JUL2019
$283.40
4
19AUG2016
$201.90
combined_info
Name
Ctry
ID
Order_Date
Sales
Albert Collet
FR
1
08OCT2019
$45.20
Pier Egidio Boeris
IT
3
11APR2019
$93.20
Pier Egidio Boeris
IT
3
15JUL2019
$283.40
James Kvarniq
US
4
19AUG2016
$201.90

customer_info

Name	Ctry	ID
Albert Collet	FR	1
Mercedes Martínez	ES	2
Pier Egidio Boeris	IT	3
James Kvarniq	US	4

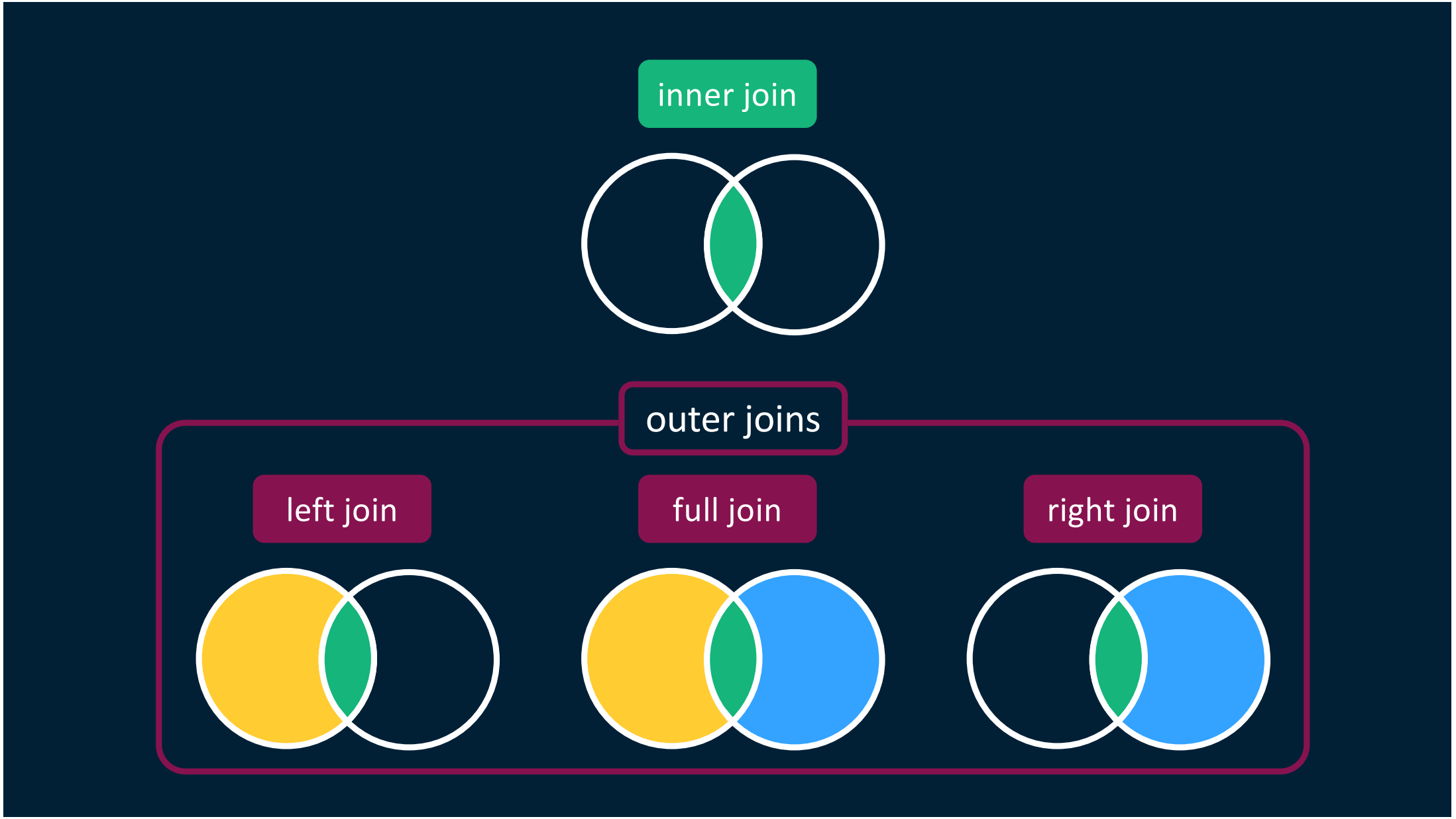


order_info

ID	Order_Date	Sales
.	29SEP2019	$78.25
1	08OCT2019	$45.20
3	11APR2019	$93.20
3	15JUL2019	$283.40
4	19AUG2016	$201.90

combined_info

Name	Ctry	ID	Order_Date	Sales
Albert Collet	FR	1	08OCT2019	$45.20
Pier Egidio Boeris	IT	3	11APR2019	$93.20
Pier Egidio Boeris	IT	3	15JUL2019	$283.40
James Kvarniq	US	4	19AUG2016	$201.90


inner join
outer joins
left join
full join
right join

Question – Right Join

Which **ID** values would be included in a ***right join***? (Select all that apply.)

a. .
b. 1
c. 2
d. 3
e. 4

customer_info

Name	Ctry	ID
Albert Collet	FR	1
Mercedes Martínez	ES	2
Pier Egidio Boeris	IT	3
James Kvarniq	US	4

order_info

ID	Order_Date	Sales
.	29SEP2019	$78.25
1	08OCT2019	$45.20
3	11APR2019	$93.20
3	15JUL2019	$283.40
4	19AUG2016	$201.90

Answer – Right Join

Which **ID** values would be included in a ***right join***? (Select all that apply.)

a. .
b. 1
c. 2
d. 3
e. 4

customer_info

Name	Ctry	ID
Albert Collet	FR	1
Mercedes Martínez	ES	2
Pier Egidio Boeris	IT	3
James Kvarniq	US	4

order_info

ID	Order_Date	Sales
.	29SEP2019	$78.25
1	08OCT2019	$45.20
3	11APR2019	$93.20
3	15JUL2019	$283.40
4	19AUG2016	$201.90

combined_info

Name	Ctry	ID	Order_Date	Sales
		.	29SEP2019	$78.25
Albert Collet	FR	1	08OCT2019	$45.20
Pier Egidio Boeris	IT	3	11APR2019	$93.20
Pier Egidio Boeris	IT	3	15JUL2019	$283.40
James Kvarniq	US	4	19AUG2016	$201.90

Joining Tables

This demonstration illustrates using the Query step in a flow to join two tables together.

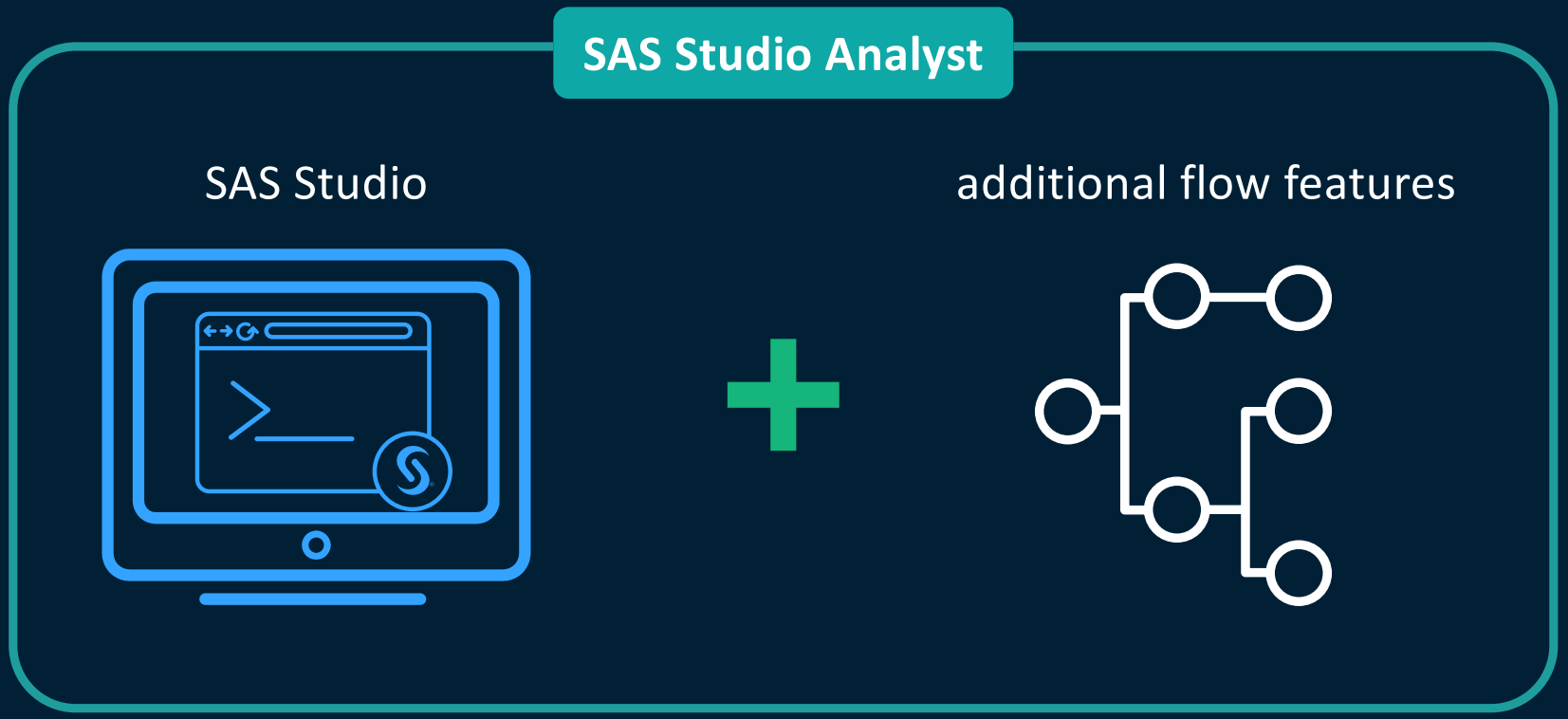
Lesson 3: Transforming and Analyzing Data

3.1 Creating Simple Queries

3.2 Using SAS Studio Analyst Steps

3.3 Creating Results with Tasks

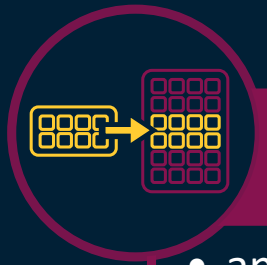
What Is SAS Studio Analyst?



additional flow features

Filter Rows step

select a subset of rows to write to output based on conditions

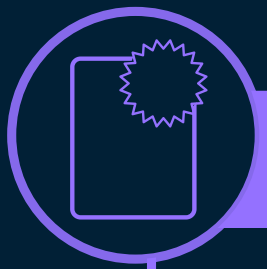


Insert Rows step

- append rows to a new or existing output table
- replace existing rows in the output table with rows from the input table

Branch Rows step

split a table into multiple output tables based on conditions



custom steps

use the Designer interface with SAS code to create shareable custom steps

Branching Rows to Multiple Tables

This demonstration illustrates how to use the Branch Rows step to split a table into multiple output tables.

Activity – Streaming Nonmatching Rows

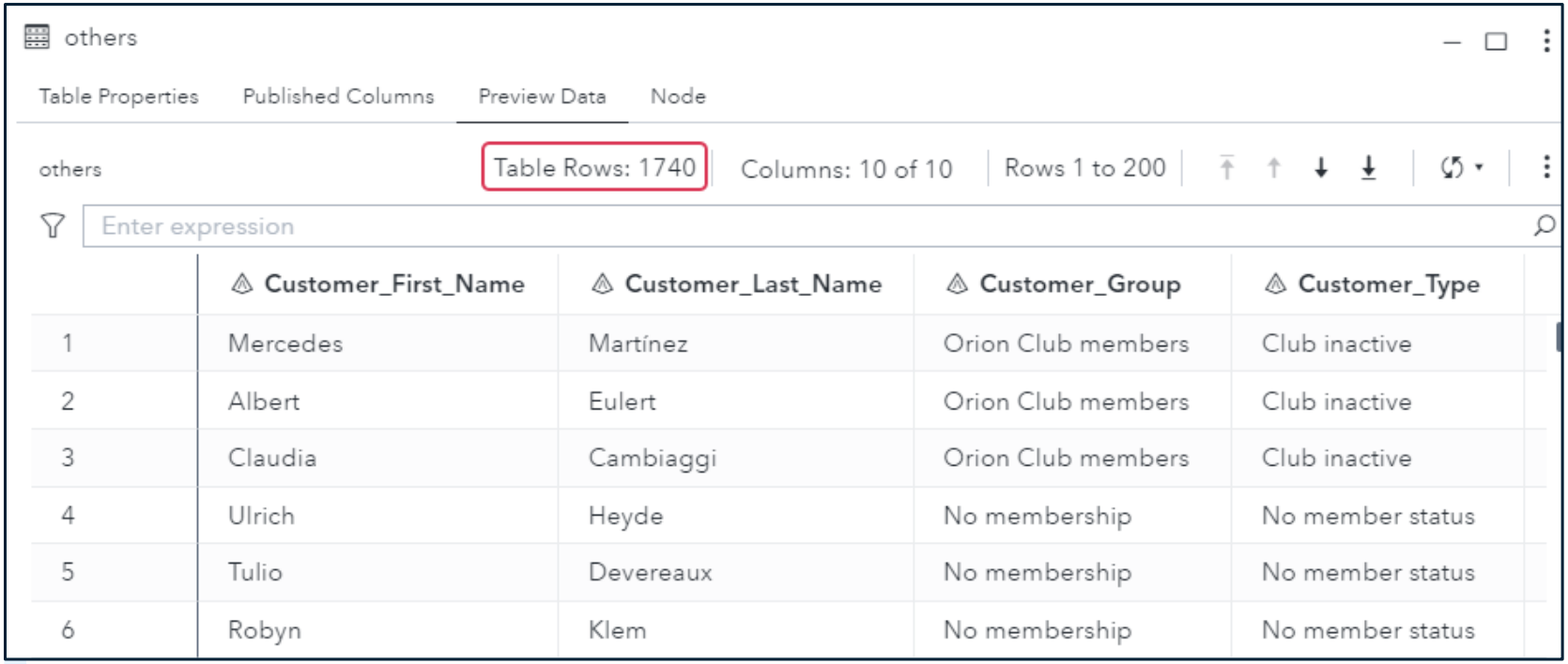
1. In the Explorer section, expand **Folder Shortcuts** > **VSTU**, and open the **vs03a03.flw** flow file.
2. Select the **Split by Activity Level** Branch Rows node. In the node details, select **Condition** > **Add nonmatching condition** to stream all nonmatching rows to a fourth output table.
3. Right-click the output port for the fourth output table and select **Add a table**. In the **Library** box, enter **work**, and in the **Table name** box, enter **others**.
4. Right-click the **Split by Activity Level** node and select **Run node**.

How many customers are listed in the **others** table?

Answer – Streaming Nonmatching Rows

How many customers are listed in the **others** table?

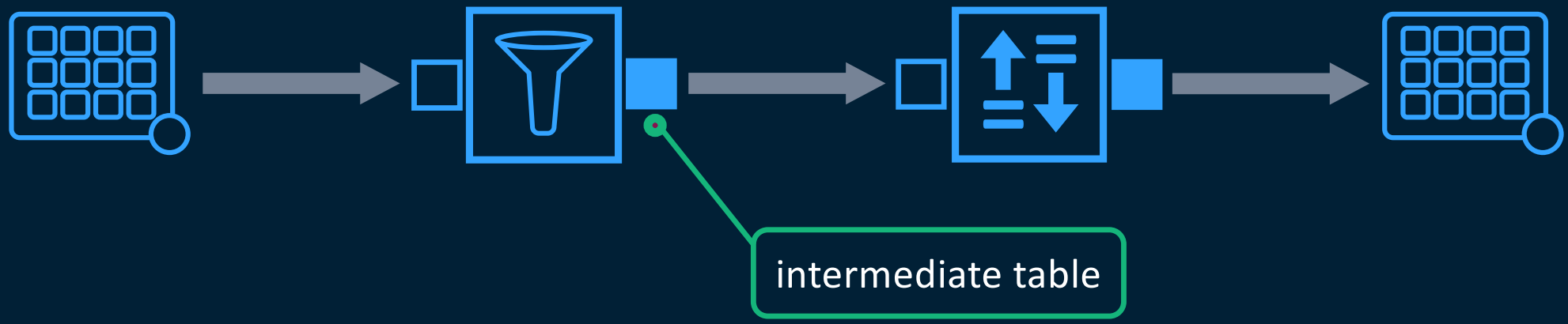
1740



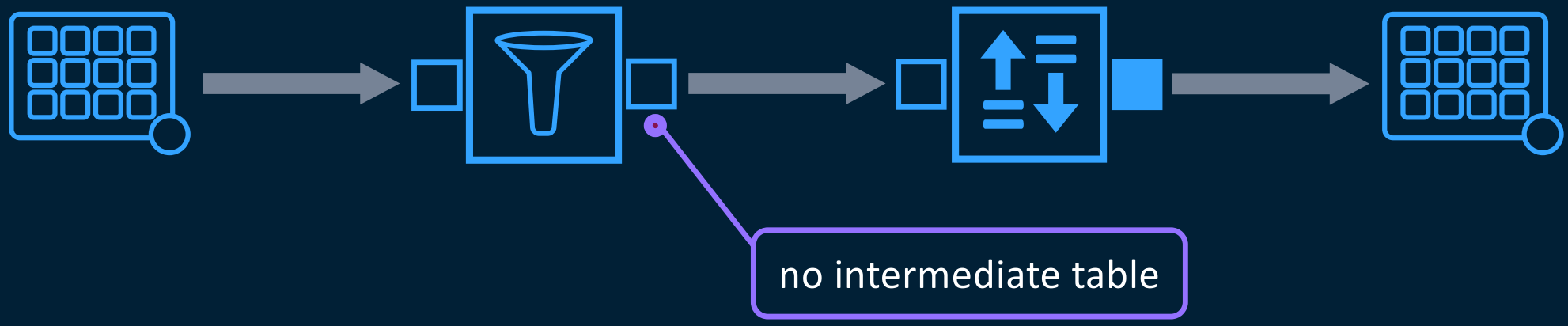
	Customer_First_Name	Customer_Last_Name	Customer_Group	Customer_Type
1	Mercedes	Martínez	Orion Club members	Club inactive
2	Albert	Eulert	Orion Club members	Club inactive
3	Claudia	Cambiaggi	Orion Club members	Club inactive
4	Ulrich	Heyde	No membership	No member status
5	Tulio	Devereaux	No membership	No member status
6	Robyn	Klem	No membership	No member status

Optimizing Steps in a Flow

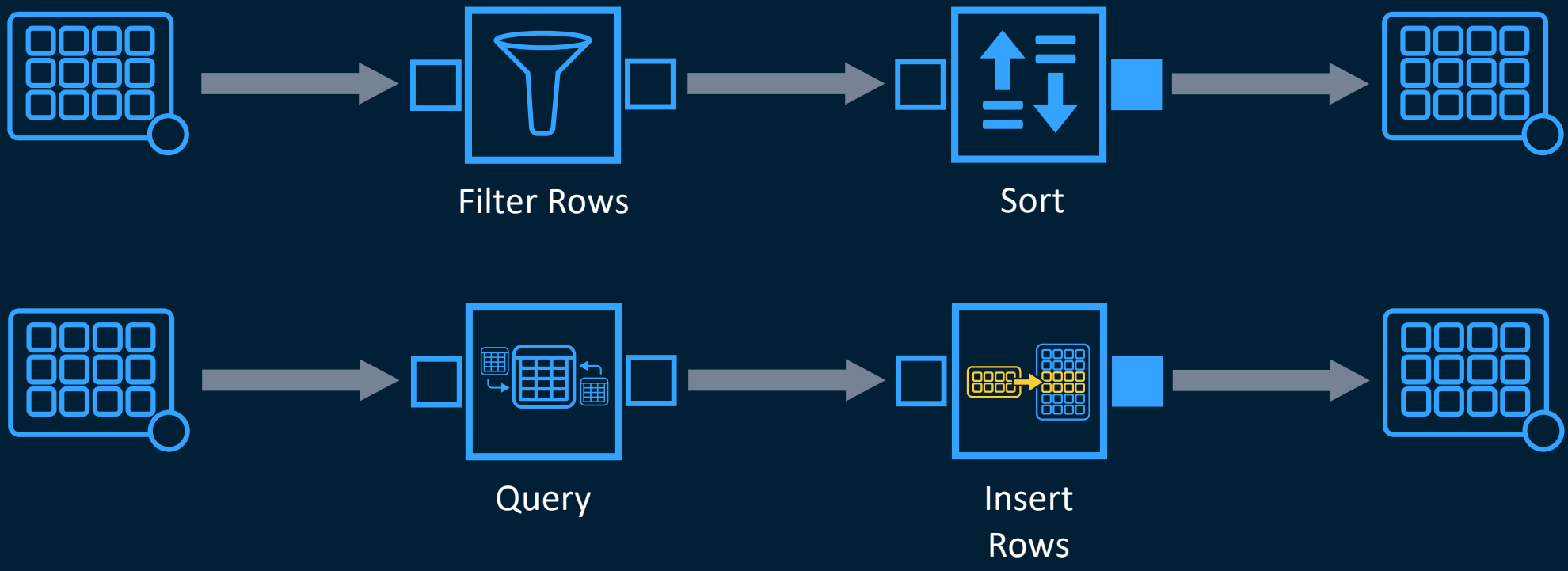
without optimization



with optimization



node combinations



Inserting Rows with Optimization

This demonstration illustrates how to use the Insert Rows step and optimize the steps in a flow.

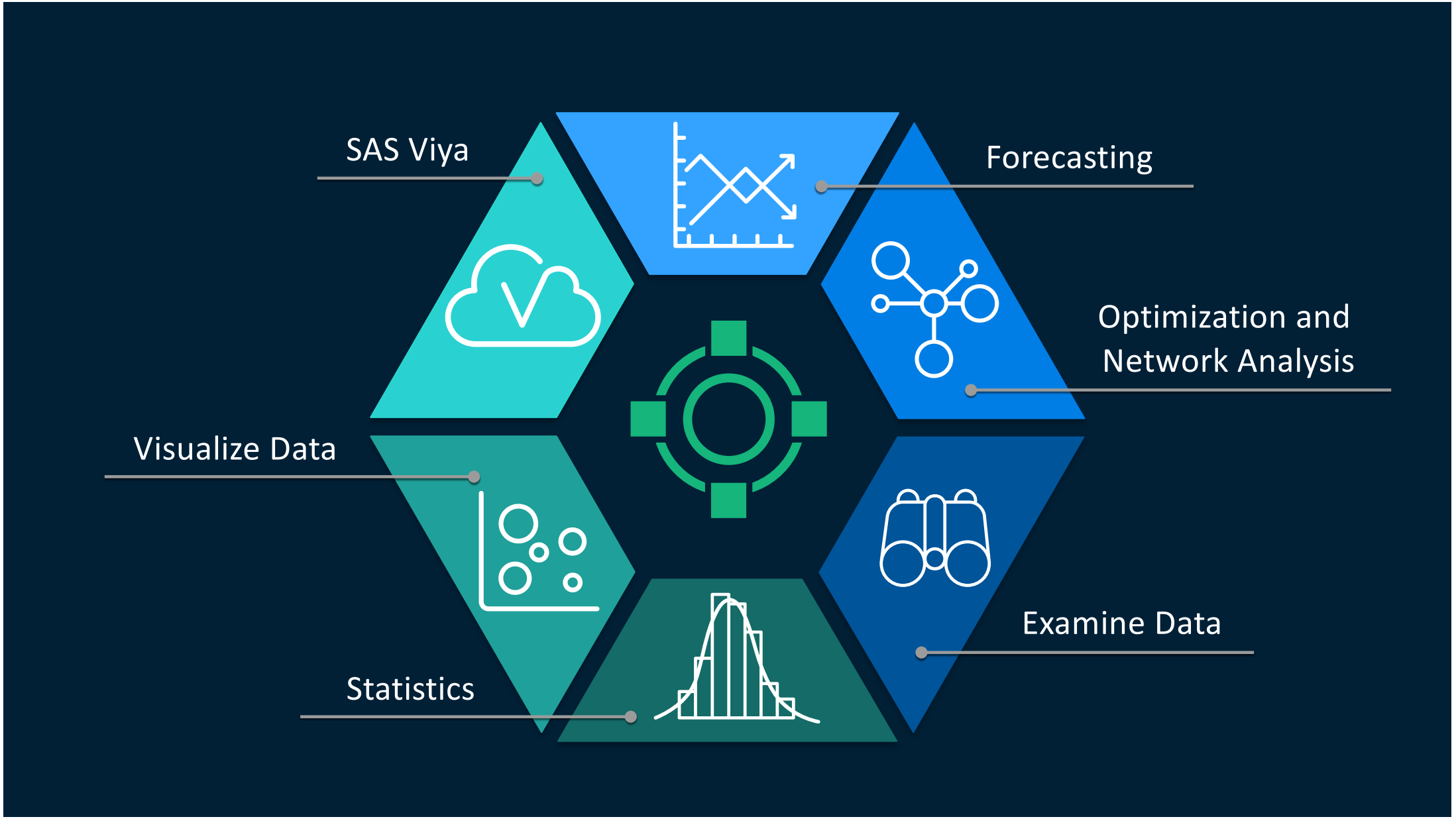
Lesson 3: Transforming and Analyzing Data

3.1 Creating Simple Queries

3.2 Using SAS Studio Analyst Steps

3.3 Creating Results with Tasks

What Is a Task?


SAS Viya
Forecasting
Optimization and
Network Analysis
Visualize Data
Examine Data
Statistics

Generating a One-Way Frequency Report

This demonstration illustrates using the One-Way Frequencies task to generate a report and graph.

Activity – Using the Bar Chart Task

1. In the Tasks section, expand **Visualize Data** > **Graph** and double-click **Bar Chart**.
2. Specify the **orders** table from the **orion** library as the input table.
3. Assign **Product_Category** to the **Category** role.
4. On the **Appearance** tab, under the **Bars** heading, select the **Show labels** check box.
5. Run the task.

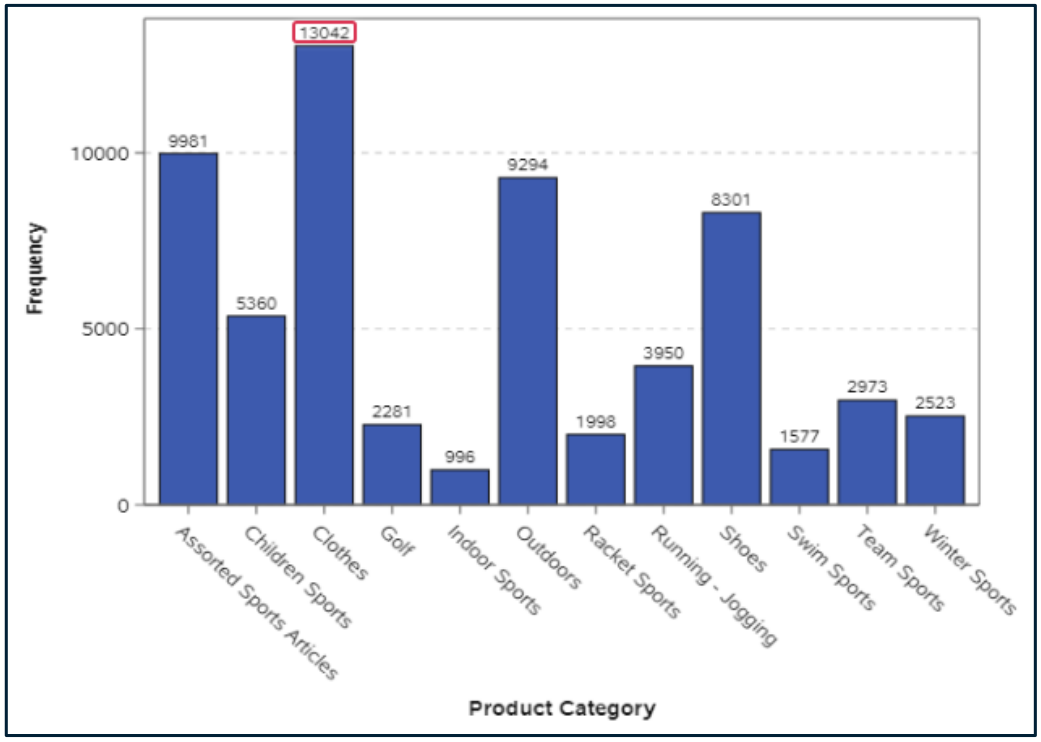
Which product category has the highest number of orders?

How many orders were placed for that product category?

Answer – Using the Bar Chart Task

Which product category has the highest number of orders? **Clothes**

How many orders were placed for that product category? **13,042**



Copying Code to a Flow


```
proc means data=sashelp.class;
    var Age Height Weight;
    output out=work.class_stats;
run;
```


```
proc means data=sashelp.class;
    var Age Height Weight;
    output out=work.class_stats;
run;
```

not automatically updated

macro variable

&

```
proc means data=sashelp.class;
    var Age Height Weight;
    output out=work.class_stats;
run;
```

_input1

&

```
proc means data=&_input1;
    var Age Height Weight;
    output out=&_output1;
run;
```

_output1

&


```
proc means data=&_input1;
    var Age Height Weight;
    output out=&_output1;
run;
```


```
proc means data=&_input1;
    var Age Height Weight;
    output out=&_output1;
run;
```

Copying Code to a Flow

This demonstration illustrates how to copy task-generated code to a flow and use macro variables to reference the input table.

Question – Input Port Macro Variable

Suppose the following code is in a SAS program node and the **orion.orders** table is connected at the input port. To use the **_input1** macro variable to reference the table connected at the input port, you replace **orion.orders** with which of the following?

a. **_input1**
b. **"_input1"**
c. **&_input1**
d. **%_input1**

```
proc sql;
select *
    from orion.orders
    where Profit > 250;
quit;
```

Answer – Input Port Macro Variable

Suppose the following code is in a SAS program node and the **orion.orders** table is connected at the input port. To use the **_input1** macro variable to reference the table connected at the input port, you replace **orion.orders** with which of the following?

a. **_input1**
b. **"_input1"**
c. **&_input1**
d. **%_input1**

```
proc sql;
select *
    from &_input1
    where Profit > 250;
quit;
```