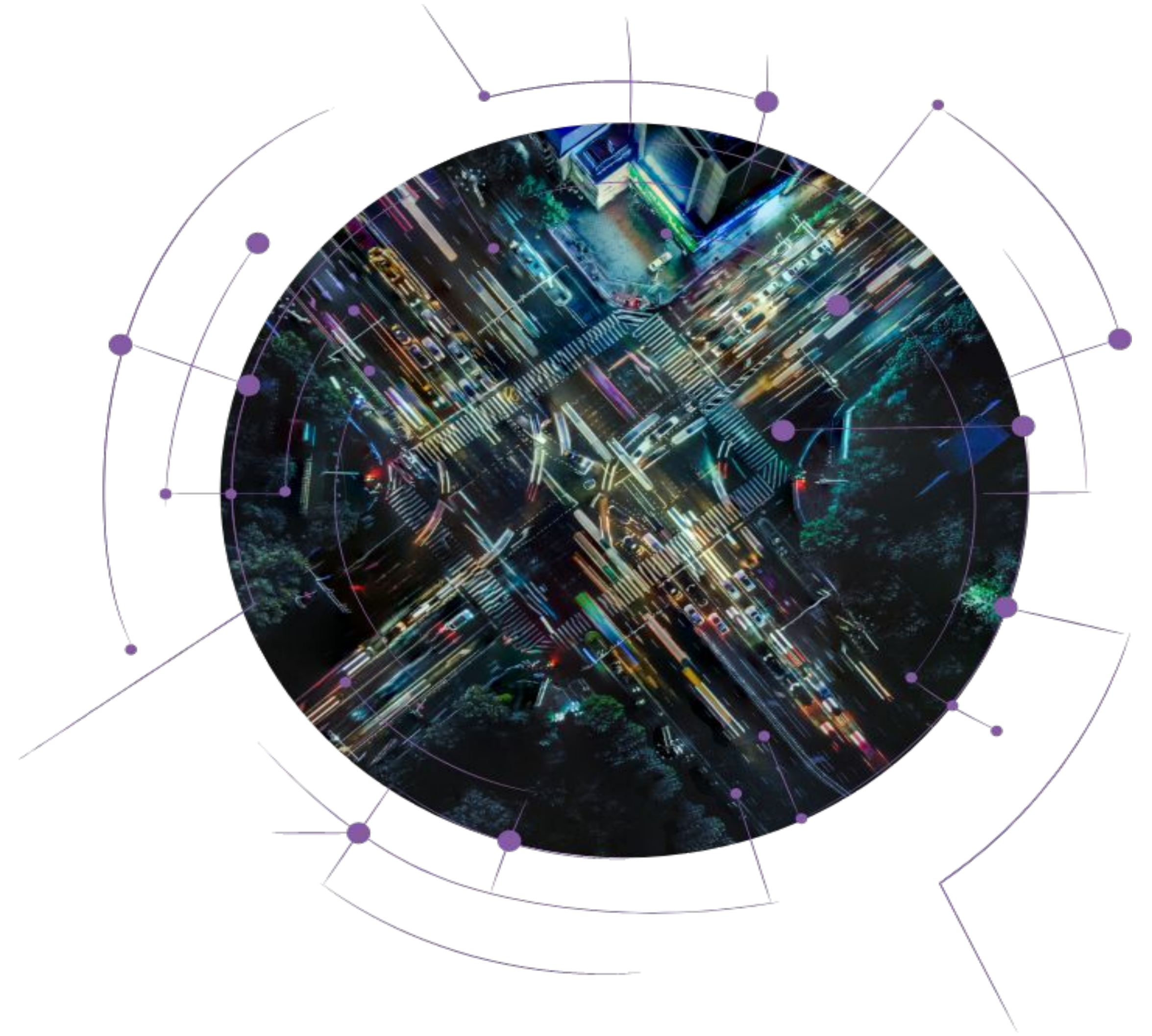


DATA SOCIETY:

Introduction to Tableau

Part 10



Module completion checklist

Objective	Complete
Implement string calculations on a given dataset	✓
Implement date calculations on a given dataset	
Implement type calculations on a given dataset	
Implement logic calculations on given dataset	

Date functions

- **Date functions** allow you to manipulate dates and extract certain features, like month, day, week, etc.

Date ▼

Enter search text

DATEADD

DATEDIFF

DATENAME

DATEPARSE

DATEPART

DATETRUNC

DAY

ISDATE

ISOQUARTER

ISOWEEK

ISOWEEKDAY

ISOYEAR

MAX

MIN

MONTH

NOW

QUARTER

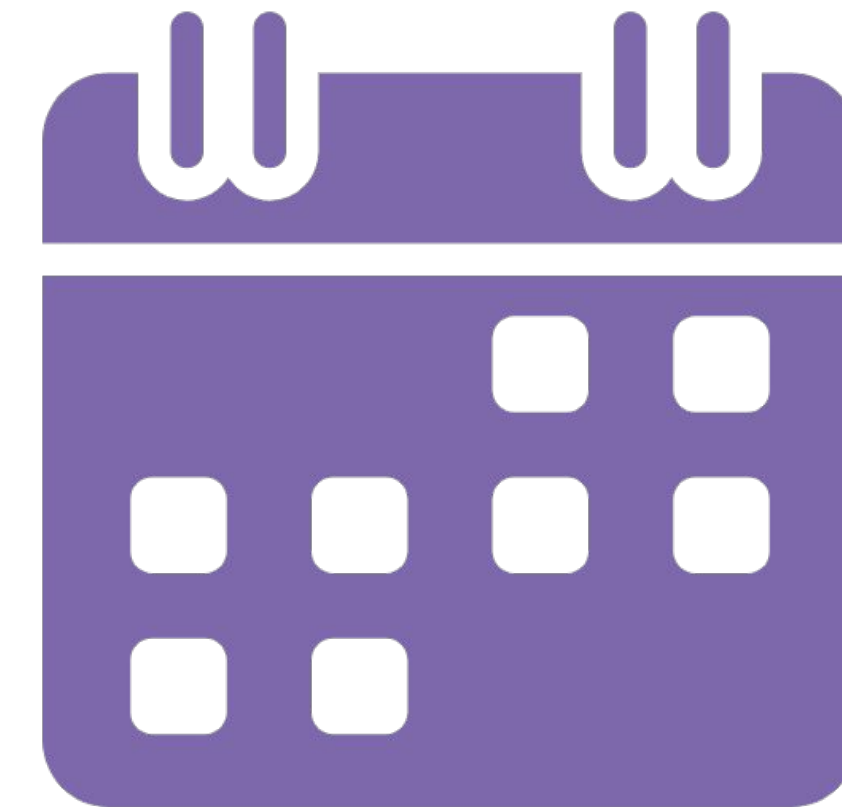
TODAY

WEEK

YEAR

Date function: DATENAME

- **DATENAME** takes two arguments, returning a requested **date_part** for a given **date**.
 - DATENAME('year', #2004-04-15#) = **"2004."**
 - DATENAME('month', #2004-04-15#) = **"April."**



DATEPARSE and DATEPART

- **DATEPARSE**

- Allows you to control date formatting.

```
Example: DATEPARSE  
("dd.MMMM.yyyy", "15.April.  
2004") = 2004-04-15 12:00:00  
AM
```

- **DATEPART**

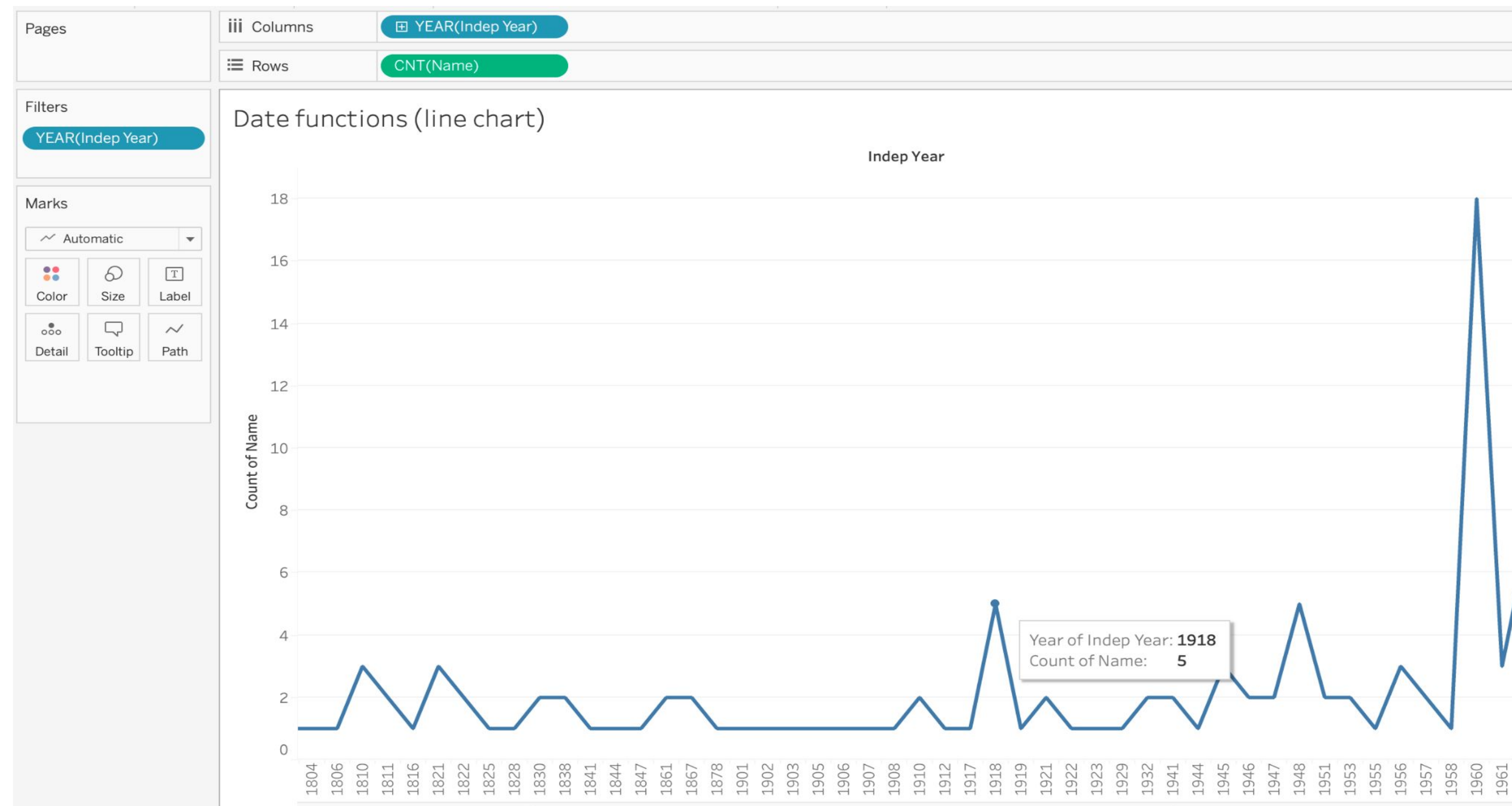
- Allows you to get a piece of a date.

```
Example: DATEPART('month',  
#2004-04-15#) = 4
```

- Both DATENAME and DATEPART can be used to return the month of a given date – so what's the difference?

Setting up a date-related chart

- To practice using these functions, let's use the world data to plot a **line chart** showing the number of countries that became independent in a given year.

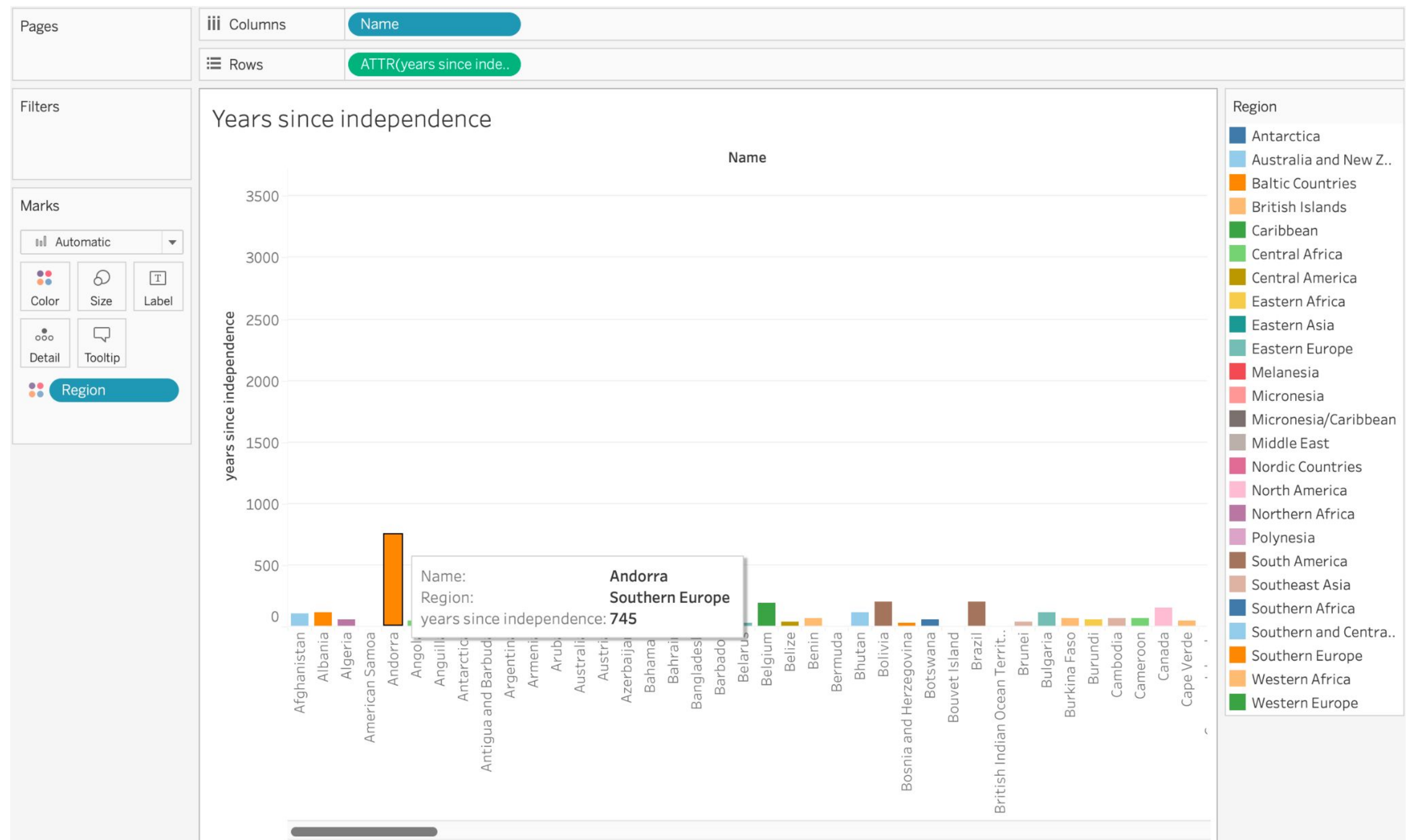


Date functions in the world dataset

- Get the years since independence for a country using these functions:
 - TODAY
 - YEAR
- Next, replot the independence.
 - Graph as a bar chart.
- What does this chart tell us?

years since independence

```
ZN(DATEDIFF("year", [Indep Year], TODAY(), "sunday"))
```

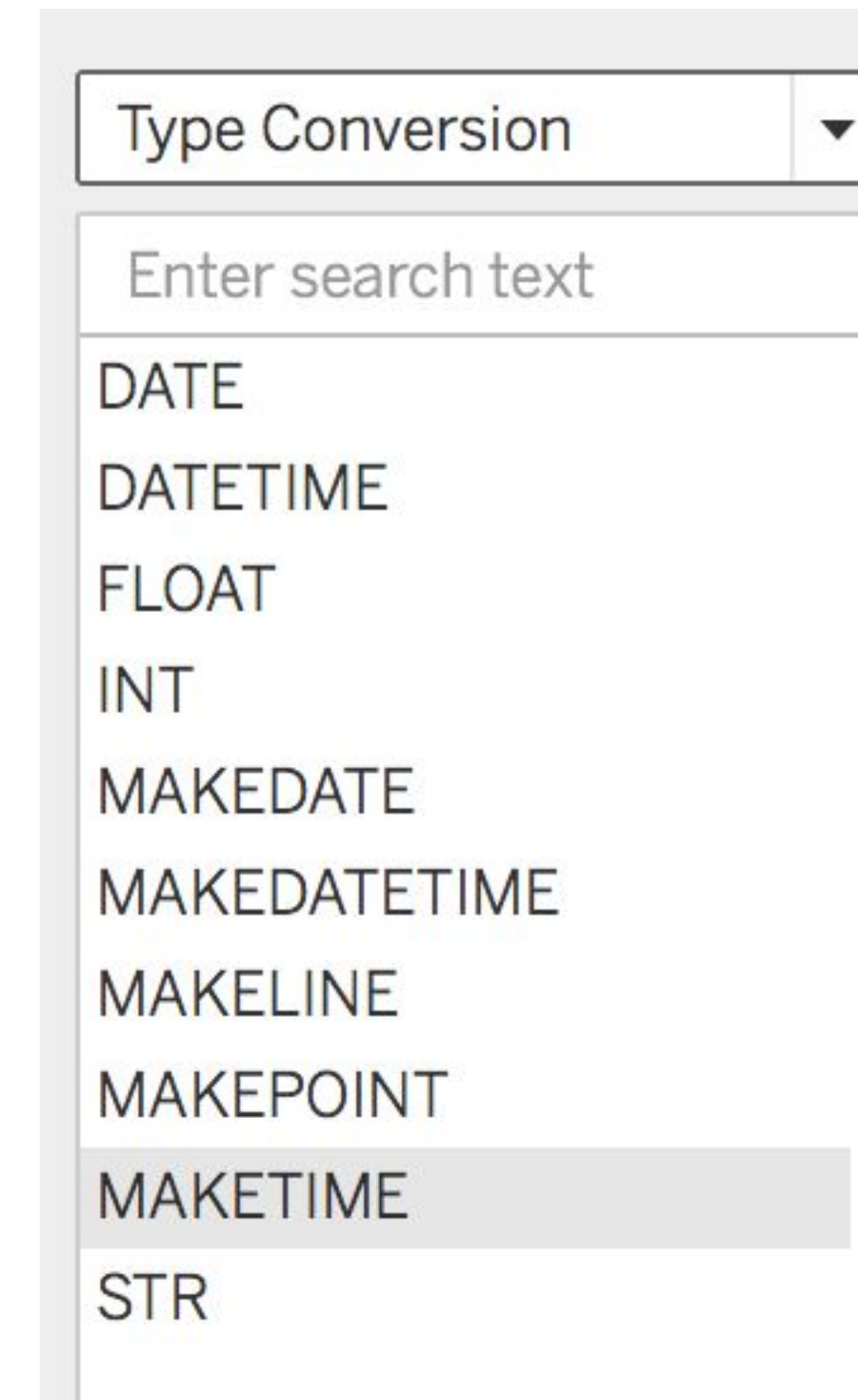


Module completion checklist

Objective	Complete
Implement string calculations on a given dataset	✓
Implement date calculations on a given dataset	✓
Implement type calculations on a given dataset	
Implement logic calculations on given dataset	

Type functions

- **Type functions** allow you to convert fields from one data type to another.
- For example, you can convert **numbers to strings** so that Tableau does not try to aggregate them.



Typecasting functions

- Typecasting functions are type functions that convert one data type to another.

- **STR**

```
STR([Age])
```

- **INT**

```
INT(8.0/3.0) = 2  
INT(4.0/1.5) = 2  
INT(0.50/1.0) = 0  
INT(-9.7) = -9
```

- **FLOAT**

```
FLOAT(3) = 3.000
```

MAKE functions

- **MAKE functions** return date and time values given certain arguments.
- **MAKEDATE** returns a time value constructed from the specified **year**, **month**, and **date**.
- **MAKEDATETIME** returns a datetime that combines a date and a time.
 - The **date** can be a date, datetime, or a string type.
 - The **time** must be a datetime.

```
MAKEDATE(2004, 4, 15) = #April  
15, 2004#
```

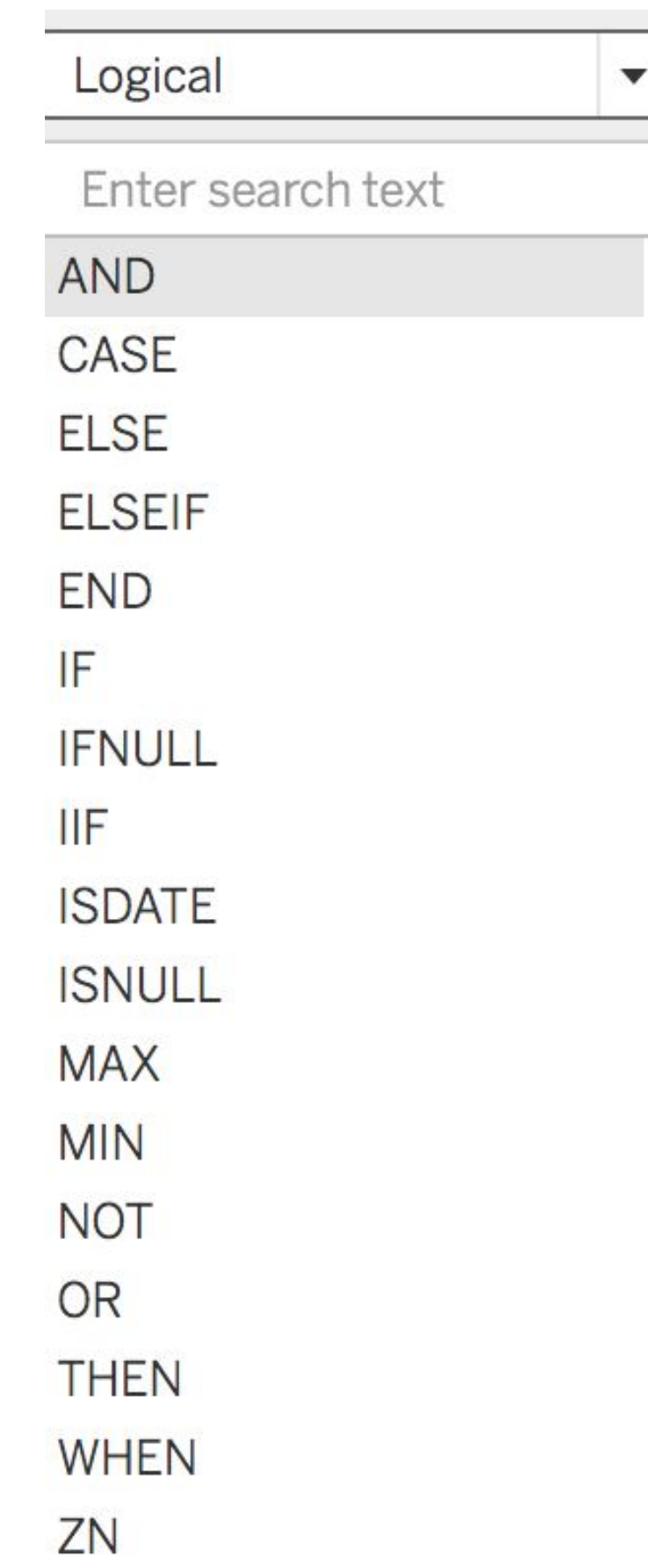
```
MAKEDATETIME("1899-12-30",  
#07:59:00#) = #12/30/1899 7:59:00  
AM#  
  
MAKEDATETIME([Date], [Time]) =  
#1/1/2001 6:00:00 AM#
```


Module completion checklist

Objective	Complete
Implement string calculations on a given dataset	✓
Implement date calculations on a given dataset	✓
Implement type calculations on a given dataset	✓
Implement logic calculations on given dataset	

Logical functions

- **Logical functions** allow you to determine if a certain condition is **true or false** (boolean logic).
- Tableau offers all the basic logical functions for managing **control flow** and performing **Boolean tests**.



Case switching and conditionals

- **Case switch functions :**
 - CASE , WHEN, THEN, ELSE , END

```
Example: CASE [RomanNumeral]  
WHEN 'I' THEN 1 WHEN 'II'  
THEN 2 ELSE 3 END
```

- **IF, ELSE conditionals:**
 - IF , THEN, ELSEIF, THEN, END

```
Example: IF [Profit] > 0  
THEN 'Profitable' ELSEIF  
[Profit] = 0 THEN  
'Breakeven' ELSE 'Loss' END
```


IIF test

- **IIF** checks whether a condition is met, and then returns:
 - One value if TRUE.
 - Another value if FALSE.
 - An optional third value or NULL if unknown.

```
IIF(test, then, else,  
[unknown])
```

Checks whether a condition is met, and returns one value if TRUE, another value if FALSE, and an optional third value or NULL if unknown.

```
Example: IIF([Profit] > 0,  
'Profit', 'Loss')
```

Logical functions in the world dataset

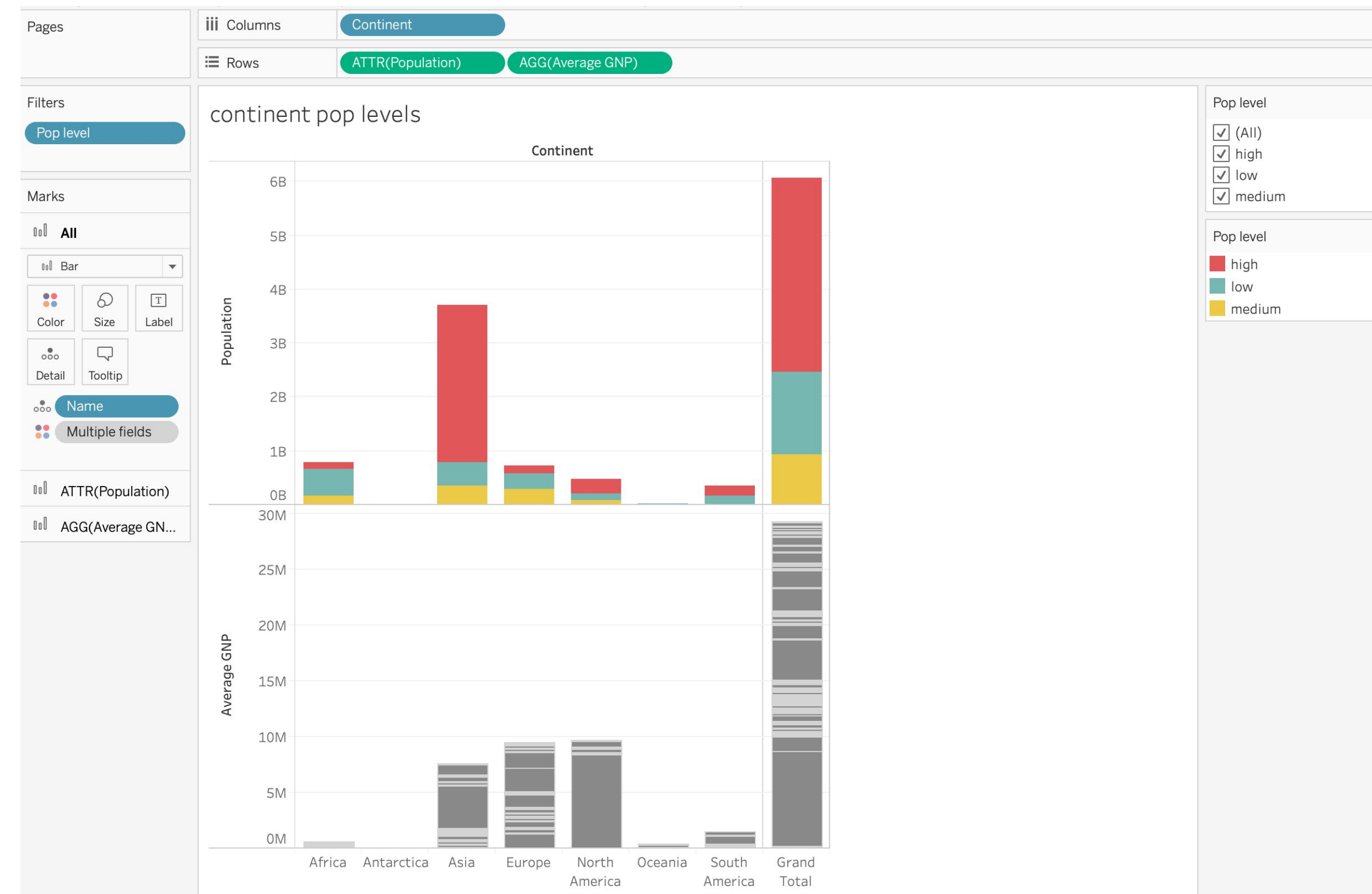
- Use an **IF ELSE** conditional to make a “**high**”, “**medium**” and “**low**” population column.
- Apply it to the population analysis.

Pop Level

```
IF [Population (country.csv)] > 100000000
THEN "high"
ELSEIF [Population (country.csv)] > 50000000
THEN "medium"
ELSE "low"
END
```

The calculation is valid. 1 Dependency

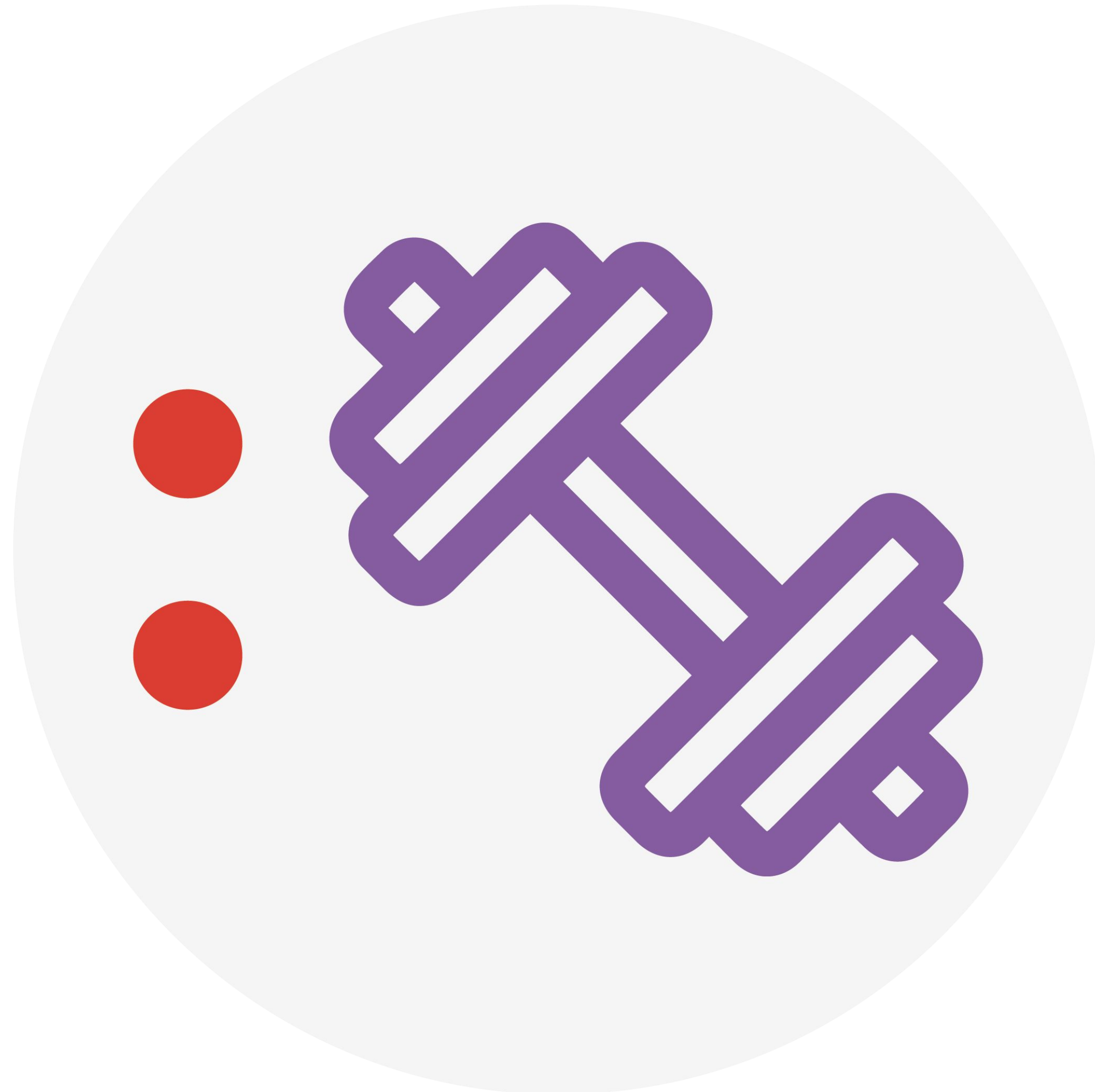
Apply OK



Knowledge check 10



Exercise 10



Module completion checklist

Objective	Complete
Implement string calculations on a given dataset	✓
Implement date calculations on a given dataset	✓
Implement type calculations on a given dataset	✓
Implement logic calculations on given dataset	✓

Congratulations!

In the past few modules, we covered:

- String Functions
 - MID
 - REPLACE
 - TRIM
- Date Functions
 - DATENAME
 - DATEPARSE
 - DATEPART
- Type Functions
 - STR
 - INT
 - FLOAT
 - MAKEDATE and MAKEDATETIME
- Logical Functions
 - CASE
 - IF, ELSE, and IIF conditionals

Next steps

Next few modules, we will begin mapping in Tableau!

- Defining geospatial visualization
- Generating coordinates
- Cleaning and fixing geospatial data
- Creating a final dashboard

Until then, stay excited!

● End of Part 10

