Tableau Fundamentals to Advanced Dashboards

A Complete Guide for Weeks 5 & 6

Table of Contents

Week 5: Fundamentals, Data Import, and Basic Visualisations

Learning Objectives

By the end of Week 5, you will:

- Understand Tableau Desktop interface and core components
- Import and connect CSV data files with proper data types
- Clean and prepare data using Tableau's data preparation features
- Create fundamental visualisation types and charts
- Build your first basic dashboard

Getting Started with Tableau

What is Tableau?

Tableau is a leading data visualisation platform that enables users to connect, visualise, and share data insights. The Tableau ecosystem consists of:

- **Tableau Desktop**: Primary authoring tool for creating visualisations and dashboards
- **Tableau Server**: On-premises collaboration and sharing platform
- **Tableau Cloud**: Cloud-based sharing and collaboration service
- **Tableau Public**: Free platform for sharing public visualisations
- **Tableau Mobile**: Mobile applications for iOS and Android

Tableau Desktop Interface Overview

Main Components:

1. **Data Pane**: Contains data sources, dimensions, and measures
2. **Analytics Pane**: Provides access to statistical models and reference lines
3. **Pages Shelf**: Creates animated visualisations over time or categories
4. **Filters Shelf**: Applies filters to limit data shown in views
5. **Marks Card**: Controls visual properties like colour, size, and shape
6. **Columns and Rows Shelves**: Define the structure of your visualisation
7. **Canvas**: Central workspace where visualisations are built

Key Concepts:

- **Dimensions**: Categorical fields that define data structure (Blue pills)
- **Measures**: Numerical fields that can be aggregated (Green pills)
- **Discrete vs Continuous**: Discrete creates headers, continuous creates axes
- **Live vs Extract**: Live connects directly to data, Extract imports data locally

---

Data Import and Connection

Preparing Northwind CSV Files

Before connecting to Tableau, we need to export the Northwind database tables as CSV files.

Required CSV Files:

```
- categories.csv
- customers.csv
- employees.csv
- orders.csv
- order_details.csv
- products.csv
- shippers.csv
- suppliers.csv
```

Exporting from Database to CSV:

```sql
-- Export Categories table
SELECT * FROM Categories
INTO OUTFILE 'categories.csv'
FIELDS TERMINATED BY ','
ENCLOSED BY '"'
LINES TERMINATED BY '\n';

-- Export Customers table
SELECT * FROM Customers
```

```
INTO OUTFILE 'customers.csv'
FIELDS TERMINATED BY ','
ENCLOSED BY '"'
LINES TERMINATED BY '\n';

-- Export Employees table
SELECT * FROM Employees
INTO OUTFILE 'employees.csv'
FIELDS TERMINATED BY ','
ENCLOSED BY '"'
LINES TERMINATED BY '\n';

-- Export Orders table
SELECT * FROM Orders
INTO OUTFILE 'orders.csv'
FIELDS TERMINATED BY ','
ENCLOSED BY '"'
LINES TERMINATED BY '\n';

-- Export OrderDetails table
SELECT * FROM OrderDetails
INTO OUTFILE 'order_details.csv'
FIELDS TERMINATED BY ','
ENCLOSED BY '"'
LINES TERMINATED BY '\n';

-- Export Products table
SELECT * FROM Products
INTO OUTFILE 'products.csv'
FIELDS TERMINATED BY ','
ENCLOSED BY '"'
LINES TERMINATED BY '\n';

-- Export Shippers table
SELECT * FROM Shippers
INTO OUTFILE 'shippers.csv'
FIELDS TERMINATED BY ','
ENCLOSED BY '"'
LINES TERMINATED BY '\n';

-- Export Suppliers table
SELECT * FROM Suppliers
INTO OUTFILE 'suppliers.csv'
FIELDS TERMINATED BY ','
ENCLOSED BY '"'
LINES TERMINATED BY '\n';
```

Connecting to CSV Files in Tableau

Step 1: Start New Workbook

```
1. Open Tableau Desktop
2. Click "Text file" under Connect
3. Navigate to your Northwind CSV files folder
4. Select the first CSV file (e.g., orders.csv)
```

Step 2: Data Source Page Setup

```
1. Review data preview in bottom pane
2. Check column headers are detected correctly
3. Verify data types for each field
4. Rename data source to "Northwind Orders"
```

Step 3: Join Additional Tables

```
1. Drag additional CSV files to join area
2. Configure joins between tables:
   - Orders ↔ Customers: CustomerID
   - Orders ↔ Employees: EmployeeID
   - Orders ↔ Shippers: ShipperID
   - Orders ↔ OrderDetails: OrderID
   - OrderDetails ↔ Products: ProductID
   - Products ↔ Categories: CategoryID
   - Products ↔ Suppliers: SupplierID
```

Step 4: Data Type Verification

```
Common data type adjustments:
- OrderDate: Date
- CustomerID: String
- ProductID: Number (whole)
- UnitPrice: Number (decimal)
- Quantity: Number (whole)
- Discount: Number (decimal)
```

Alternative: Union Multiple CSV Files

For Similar Structure Files:

```
1. Drag first CSV to canvas
2. Drag second CSV below first to create union
3. Configure union options:
   - Wildcard union for pattern matching
   - Specific union for selected files
4. Add "Table Name" field to distinguish sources
```

Connection Best Practices

Data Source Management:

- **Extract vs Live**: Use extracts for better performance with CSV files
- **Refresh Schedule**: Plan for regular data updates
- **File Organisation**: Maintain consistent folder structure
- **Naming Conventions**: Use descriptive, consistent file names

Performance Optimisation:

```
Extract Configuration:
1. Data → Extract Data
2. Choose "All rows" or apply filters
3. Configure aggregation if needed
4. Set refresh schedule
5. Save extract locally
```

Data Preparation and Cleaning

Data Source Filters

Applying Source Filters:

```
1. Navigate to Data Source tab
2. Click "Add Filter" in top-right
3. Select field to filter
4. Configure filter conditions:
   - Date ranges for OrderDate
   - Exclude test customers
   - Remove cancelled orders
```

Filter Types:

- **Wildcard**: Text pattern matching
- **Condition**: Logical conditions (>, <, =)
- **Top/Bottom**: Limit to top N values
- **Custom**: Advanced filter expressions

Data Cleaning Operations

Handling Missing Values:

```
Options for NULL values:
1. Filter out NULL values
2. Replace with default values
3. Create calculated field to handle NULLs
4. Use ISNULL() function in calculations
```

String Cleaning:

```
Common string operations:
- TRIM(): Remove leading/trailing spaces
- UPPER()/LOWER(): Change case
- REPLACE(): Replace specific text
- SPLIT(): Divide text into parts
- LEFT()/RIGHT(): Extract portions
```

Date Formatting:

```
Date field operations:
- DATEPARSE(): Convert text to date
- YEAR(), MONTH(), DAY(): Extract date parts
- DATEADD(): Add time periods
- DATEDIFF(): Calculate differences
- TODAY(), NOW(): Current date/time
```

Calculated Fields

Creating Calculated Fields:

```
1. Right-click in Data pane
2. Select "Create Calculated Field"
3. Enter field name and formula
4. Validate syntax
5. Click OK to create
```

Common Business Calculations:

```
// Revenue Calculation
Revenue = [Quantity] * [Unit Price] * (1 - [Discount])

// Order Age in Days
Order Age = DATEDIFF('day', [Order Date], TODAY())

// Full Customer Name
Customer Full Name = [Contact Name] + " (" + [Customer Name] + ")"

// Category Display
Category Short = LEFT([Category Name], 3)

// Price Tier
Price Tier =
IF [Unit Price] > 50 THEN "Premium"
ELSEIF [Unit Price] > 20 THEN "Standard"
ELSE "Budget"
END
```

Data Validation

Quality Checks:

```
1. Check for duplicates
2. Validate date ranges
3. Verify numerical ranges
4. Confirm join relationships
5. Test calculated fields
```

Data Profiling:

```
Techniques:
- View data in tabular format
- Check distinct value counts
- Identify outliers
- Validate aggregations
- Test sample records
```

Creating Basic Visualisations

Fundamental Chart Types

Bar Charts

```
Purpose: Compare categorical data
Building Steps:
1. Drag dimension to Rows shelf
2. Drag measure to Columns shelf
3. Sort by measure value
4. Add colour encoding if needed
5. Format labels and titles
```

Line Charts

```
Purpose: Show trends over time
Building Steps:
1. Drag date field to Columns shelf
2. Drag measure to Rows shelf
3. Ensure date is continuous (green pill)
4. Add trend lines if appropriate
5. Configure date aggregation level
```

Scatter Plots

```
Purpose: Show relationships between two measures
Building Steps:
1. Drag first measure to Columns shelf
2. Drag second measure to Rows shelf
3. Drag dimension to Detail on Marks card
4. Add colour or size encoding
5. Include trend line for correlation
```

Pie Charts

```
Purpose: Show part-to-whole relationships
Building Steps:
1. Use "Pie" mark type
2. Drag dimension to Colour
3. Drag measure to Angle
4. Limit to 5-7 categories
5. Add percentage labels
```

Tree Maps

```
Purpose: Hierarchical data with size encoding
Building Steps:
1. Use "Square" mark type
2. Drag dimension to Colour
3. Drag measure to Size
4. Add labels for clarity
5. Configure colour palette
```

Marks Card Configuration

Visual Properties:

- **Colour**: Categorical or continuous encoding
- **Size**: Numerical representation
- **Label**: Text annotations
- **Detail**: Additional data context
- **Tooltip**: Hover information
- **Shape**: Different symbols
- **Path**: Connect points in sequence

Formatting Options:

```
Colour Formatting:
1. Click Colour on Marks card
2. Choose palette type
3. Configure stepped/continuous colours
4. Set transparency levels
5. Add borders if needed

Size Formatting:
1. Click Size on Marks card
2. Adjust size range
3. Configure proportional/absolute sizing
4. Test readability at different sizes
```

Show Me Panel

Quick Chart Creation:

```
Using Show Me:
1. Select relevant fields in Data pane
2. Open Show Me panel (top-right)
3. Choose appropriate chart type
4. Tableau automatically configures shelves
5. Customise as needed
```

Chart Recommendations:

- Tableau suggests charts based on selected fields
- Considers data types and cardinality
- Follows visualisation best practices
- Provides starting point for customisation

Hands-on Lab Session 1

Lab Exercise: Northwind Traders Sales Analysis

Objective: Create a sales dashboard using Northwind CSV data

Dataset: Northwind CSV files (exported from database)

Part 1: Data Connection and Preparation

Task 1.1: Initial Data Connection

```
1. Open Tableau Desktop
2. Connect to Text file → select orders.csv
3. Review data structure and field types
4. Rename data source to "Northwind Sales"
```

Task 1.2: Join Additional Tables

```
Join Configuration:
1. Drag customers.csv to join area
   - Join type: Inner Join
   - Join clause: Orders.CustomerID = Customers.CustomerID

2. Drag employees.csv to join area
   - Join type: Left Join
   - Join clause: Orders.EmployeeID = Employees.EmployeeID

3. Drag order_details.csv to join area
   - Join type: Inner Join
   - Join clause: Orders.OrderID = OrderDetails.OrderID

4. Drag products.csv to join area
   - Join type: Inner Join
   - Join clause: OrderDetails.ProductID = Products.ProductID

5. Drag categories.csv to join area
   - Join type: Inner Join
   - Join clause: Products.CategoryID = Categories.CategoryID
```

```
6. Drag suppliers.csv to join area
   - Join type: Left Join
   - Join clause: Products.SupplierID = Suppliers.SupplierID
```

Task 1.3: Data Type Verification

```
Field corrections:
- Order Date: Date
- Customer ID: String (if numeric)
- Product ID: Number (whole)
- Unit Price: Number (decimal)
- Quantity: Number (whole)
- Discount: Number (decimal, between 0-1)
- Employee ID: Number (whole)
- Category ID: Number (whole)
```

Task 1.4: Create Calculated Fields

```
// Revenue calculation
Revenue = [Quantity] * [Unit Price] * (1 - [Discount])

// Full Employee Name
Employee Name = [First Name] + " " + [Last Name]

// Order Year
Order Year = YEAR([Order Date])

// Discount Category
Discount Level =
IF [Discount] = 0 THEN "No Discount"
ELSEIF [Discount] <= 0.1 THEN "Low Discount"
ELSEIF [Discount] <= 0.2 THEN "Medium Discount"
ELSE "High Discount"
END

// Customer Country Group
Region =
IF [Country] IN ("USA", "Canada") THEN "North America"
ELSEIF [Country] IN ("UK", "Ireland", "Germany", "France", "Italy",
"Spain", "Portugal", "Belgium", "Netherlands", "Switzerland", "Austria",
"Denmark", "Sweden", "Finland", "Norway", "Poland") THEN "Europe"
ELSEIF [Country] IN ("Brazil", "Argentina", "Venezuela", "Mexico") THEN
"Latin America"
ELSE "Other"
END
```

Part 2: Basic Visualisations

Task 2.1: Sales Overview Dashboard

Create a new worksheet for each visualisation:

Visual 1: Total Sales KPI

```
1. Create new worksheet: "Sales KPI"
2. Drag Revenue to Text on Marks card
3. Change mark type to Text
4. Format as currency with no decimal places
5. Increase font size to 48pt
6. Add title: "Total Sales"
7. Change aggregation to SUM if needed
```

Visual 2: Sales by Category (Horizontal Bar Chart)

```
1. Create new worksheet: "Sales by Category"
2. Drag Category Name to Rows shelf
3. Drag Revenue to Columns shelf
4. Sort descending by Revenue
5. Add colour: Drag Category Name to Colour
6. Format currency on axis
7. Add data labels showing revenue values
```

Visual 3: Monthly Sales Trend (Line Chart)

```
1. Create new worksheet: "Sales Trend"
2. Drag Order Date to Columns shelf
3. Change Order Date to continuous (green pill)
4. Right-click Order Date → select Month
5. Drag Revenue to Rows shelf
6. Add trend line: Analytics → Trend Line → Linear
7. Format axes and add reference line for average
```

Visual 4: Sales by Country (Map)

```
1. Create new worksheet: "Sales by Country"
2. Drag Country to Detail on Marks card
3. Tableau automatically creates map
4. Drag Revenue to Colour on Marks card
5. Drag Revenue to Size on Marks card
6. Configure colour to use sequential palette
7. Add country labels for top 5 countries
```

Visual 5: Top 10 Products (Bar Chart)

```
1. Create new worksheet: "Top Products"
2. Drag Product Name to Rows shelf
3. Drag Revenue to Columns shelf
4. Apply filter: Product Name → Top 10 by Revenue
5. Sort descending by Revenue
6. Format for readability
7. Add tooltip with additional product information
```

Visual 6: Employee Performance (Tree Map)

```
1. Create new worksheet: "Employee Performance"
2. Change mark type to Square
3. Drag Employee Name to Colour
4. Drag Revenue to Size
5. Drag Employee Name to Label
6. Configure colour palette for distinction
7. Add tooltip with order count and average order value
```

Part 3: Dashboard Creation

Task 3.1: Create Sales Overview Dashboard

```
1. Create new dashboard
2. Set size to "Automatic" or "Desktop (1000x800)"
3. Drag sheets onto dashboard in logical layout:
   - Sales KPI (top-left, small)
   - Sales Trend (top-right, wide)
   - Sales by Category (bottom-left, medium)
   - Sales by Country (bottom-right, medium)
```

Task 3.2: Add Interactivity

```
Dashboard Actions:
1. Dashboard → Actions → Add Action → Filter
2. Source: Sales by Category
3. Target: All other sheets
4. Run on: Select
5. Test interaction by clicking category bars

2. Add Highlight Action:
   - Source: Sales by Country
```

```
   - Target: Sales Trend
   - Run on: Hover
```

Task 3.3: Add Dashboard Filters

```
1. Drag Order Date from any sheet to dashboard
2. Configure as date range filter
3. Set default to "Last 12 months"
4. Apply to all worksheets

2. Add Category filter:
   - Drag Category Name to dashboard
   - Configure as multi-select dropdown
   - Apply to relevant worksheets
```

Part 4: Formatting and Design

Task 4.1: Consistent Formatting

```
Colour Scheme:
1. Create custom colour palette
2. Apply consistently across all charts
3. Use semantic colours (blue for sales, green for positive trends)

Typography:
1. Set consistent font family (Arial or Tableau default)
2. Use hierarchy: 16pt titles, 12pt labels, 10pt details
3. Apply consistent spacing

Layout:
1. Align all elements to grid
2. Maintain consistent margins
3. Group related visualisations
4. Add white space for clarity
```

Task 4.2: Dashboard Polish

```
1. Add dashboard title: "Northwind Traders - Sales Overview"
2. Include last updated date
3. Add company logo placeholder
4. Configure tooltip formatting
5. Test responsiveness at different sizes
6. Add explanatory text where helpful
```

Part 5: Additional Analysis Worksheets

Task 5.1: Customer Analysis

```
Customer Segmentation:
1. Create calculated field for Customer Tier:
   Customer Tier =
   IF [Revenue] > 5000 THEN "VIP"
   ELSEIF [Revenue] > 2000 THEN "Premium"
   ELSEIF [Revenue] > 500 THEN "Standard"
   ELSE "Basic"
   END

2. Create scatter plot:
   - X-axis: Number of Orders
   - Y-axis: Average Order Value
   - Colour: Customer Tier
   - Size: Total Revenue
```

Task 5.2: Product Analysis

```
ABC Analysis:
1. Create calculated field for product ranking
2. Classify products into A (top 20%), B (next 30%), C (remaining 50%)
3. Create tree map showing product performance by ABC category
4. Add filter for Category to drill down
```

Task 5.3: Time Analysis

```
Seasonal Patterns:
1. Create heat map: Month vs Weekday
2. Show Revenue as colour intensity
3. Identify seasonal trends and patterns
4. Add reference lines for benchmarks
```

Part 6: Quality Assurance

Task 6.1: Data Validation

```
Verification Checks:
1. Cross-check totals with source data
2. Verify filter interactions work correctly
3. Test drill-down functionality
4. Confirm calculations are accurate
5. Check for missing data or outliers
```

Task 6.2: User Testing

```
Usability Testing:
1. Test dashboard on different screen sizes
2. Verify mobile responsiveness
3. Check loading performance
4. Ensure intuitive navigation
5. Validate colour accessibility
```

Week 6: Advanced Dashboards, Reports, and Publishing

Learning Objectives

By the end of Week 6, you will:

- Create sophisticated visualisations using advanced chart types and analytics
- Design professional, interactive dashboards following UX best practices
- Implement advanced filtering, parameters, and dashboard actions
- Publish and share workbooks using Tableau Server/Cloud
- Configure security and governance settings

Advanced Visualisation Techniques

Advanced Chart Types

Dual-Axis Charts

```
Purpose: Compare two measures with different scales
Implementation:
1. Create first chart (e.g., bar chart for Revenue)
2. Drag second measure to right side of view
3. Right-click second measure → Dual Axis
4. Synchronise axes if measures are similar
5. Format each axis independently
6. Configure mark types for each measure
```

Bullet Charts

```
Purpose: Show performance against targets
Building Steps:
1. Create bar chart with actual values
2. Add reference line for target
3. Add reference distribution for performance ranges
4. Configure colour coding for performance zones
5. Minimal axis and clean formatting
```

Box Plots

```
Purpose: Show distribution and outliers
Implementation:
1. Select dimension and measure
2. Show Me → Box Plot
3. Configure whisker settings
4. Add individual point overlay if needed
5. Format for statistical clarity
```

Waterfall Charts

```
Purpose: Show cumulative impact of changes
Building Steps:
1. Create calculated field for cumulative values
2. Use Gantt chart mark type
3. Configure bar sizing
4. Add colour coding for positive/negative
5. Format for clear flow visualization
```

Statistical Analytics

Trend Lines and Forecasting

```
Trend Analysis:
1. Analytics pane → Trend Line
2. Configure trend model (Linear, Logarithmic, Exponential, Polynomial)
3. Show confidence bands
4. Display R-squared and P-values
5. Describe trend significance

Forecasting:
1. Analytics pane → Forecast
2. Set forecast period
3. Configure prediction intervals
4. Ignore last N periods for validation
5. Show forecast quality indicators
```

Reference Lines and Bands

```
Reference Lines:
1. Analytics pane → Reference Line
2. Choose scope (Table, Pane, Cell)
3. Configure value (Constant, Average, Median, etc.)
```

```
4. Add label and formatting
5. Set line style and colour

Reference Bands:
1. Analytics pane → Reference Band
2. Define band range
3. Configure fill colour and transparency
4. Add descriptive labels
```

Clustering Analysis

```
Statistical Clustering:
1. Use two or more measures
2. Analytics pane → Cluster
3. Tableau automatically identifies clusters
4. Configure number of clusters
5. Colour-code cluster assignments
6. Interpret cluster characteristics
```

Advanced Calculated Fields

Table Calculations

```
Running Total:
RUNNING_SUM([Revenue])

Percent of Total:
[Revenue] / TOTAL([Revenue])

Rank:
RANK([Revenue], 'desc')

Moving Average (3 periods):
WINDOW_AVG([Revenue], -2, 0)

Year-over-Year Growth:
(ZN([Revenue]) - LOOKUP(ZN([Revenue]), -12)) / ABS(LOOKUP(ZN([Revenue]),
-12))

Percent Difference from Average:
([Revenue] - WINDOW_AVG([Revenue])) / WINDOW_AVG([Revenue])
```

Level of Detail (LOD) Expressions

```
Fixed LOD - Customer Total Sales:
{FIXED [Customer Name] : SUM([Revenue])}
```

```
Include LOD - Category Average within Region:
{INCLUDE [Category Name] : AVG([Revenue])}

Exclude LOD - Overall Average Excluding Current Category:
{EXCLUDE [Category Name] : AVG([Revenue])}

Cohort Analysis:
{FIXED [Customer Name] : MIN([Order Date])}

Customer Lifetime Value:
{FIXED [Customer Name] : SUM([Revenue])}
```

Advanced String and Date Functions

```
Text Processing:
// Extract domain from email
RIGHT([Email], LEN([Email]) - FIND("@", [Email]))

// Clean phone numbers
REGEX_REPLACE([Phone], "[\D]", "")

Date Intelligence:
// Fiscal year (April start)
IF MONTH([Order Date]) >= 4
THEN YEAR([Order Date]) + 1
ELSE YEAR([Order Date])
END

// Business days between dates
NETWORKDAYS([Start Date], [End Date])

// Same period last year
DATEADD('year', -1, [Order Date])
```

Dashboard Design Principles

User Experience (UX) Design

Visual Hierarchy

```
Information Architecture:
1. Primary KPIs: Top-left, largest size
2. Secondary metrics: Top row, medium size
3. Supporting charts: Middle section
4. Detailed analysis: Bottom section
5. Filters and controls: Left sidebar or top

Layout Principles:
```

```
- F-pattern for Western audiences
- Z-pattern for focused dashboards
- Golden ratio for proportions
- Rule of thirds for positioning
```

Cognitive Load Management

```
Reduce Complexity:
1. Limit to 5±2 visualizations per dashboard
2. Use consistent colour coding
3. Minimize chart types used
4. Group related information
5. Progressive disclosure for details

Information Density:
- High-level overview first
- Drill-down for details
- Context-sensitive filtering
- Clear navigation paths
```

Colour Theory and Accessibility

Colour Strategy

```
Colour Purposes:
1. Brand colours for primary elements
2. Semantic colours for status (Red/Yellow/Green)
3. Sequential colours for continuous data
4. Diverging colours for deviation from norm
5. Categorical colours for distinct groups

Accessibility Guidelines:
- WCAG 2.1 AA compliance (4.5:1 contrast ratio)
- Colour-blind friendly palettes
- Alternative encodings (shape, pattern, size)
- High contrast mode support
```

Typography and Readability

```
Font Hierarchy:
- Dashboard title: 18-24pt, bold
- Section headers: 14-16pt, medium
- Chart titles: 12-14pt, medium
- Axis labels: 10-12pt, regular
- Tooltips: 9-11pt, regular

Readability Guidelines:
```

```
- Sans-serif fonts for digital display
- Adequate line spacing (1.2-1.5x font size)
- Sufficient colour contrast
- Consistent alignment
```

Responsive Design

Device Considerations

```
Desktop (1920x1080):
- Full-featured dashboard
- Multiple visualizations
- Detailed tooltips
- Advanced interactions

Tablet (1024x768):
- Simplified layout
- Larger touch targets
- Essential visualizations only
- Swipe navigation

Mobile (375x667):
- Single-column layout
- Key metrics focus
- Minimal interactions
- Scroll-based navigation
```

Layout Strategies

```
Flexible Design:
1. Use percentage-based sizing
2. Configure device-specific layouts
3. Test on multiple screen sizes
4. Prioritize content for small screens
5. Consider orientation changes
```

Interactive Features and Filters

Parameters and Dynamic Controls

Creating Parameters

```
Parameter Setup:
1. Right-click in Data pane → Create Parameter
2. Configure data type and allowable values
3. Set default value
4. Add to dashboard as control
5. Reference in calculated fields

Use Cases:
- Date range selection
- Top N filtering
- What-if analysis
- Metric switching
- Threshold setting
```

Dynamic Measures

```
Metric Selector Parameter:
1. Create parameter with metric options
2. Create calculated field referencing parameter:

Selected Metric =
CASE [Metric Parameter]
WHEN "Revenue" THEN [Revenue]
WHEN "Quantity" THEN [Quantity]
WHEN "Profit" THEN [Profit]
WHEN "Discount" THEN [Discount]
END

3. Use in visualizations for dynamic switching
```

What-If Analysis

```
Scenario Planning:
// Price increase parameter
New Revenue = [Quantity] * [Unit Price] * (1 + [Price Increase %]) * (1 -
[Discount])

// Target achievement
Target Achievement = [Actual Sales] / [Target Sales Parameter]
```

```
// Break-even analysis
Break Even Quantity = [Fixed Costs] / ([Unit Price Parameter] - [Variable
Cost])
```

Advanced Filtering Techniques

Context Filters

```
Purpose: Filter data before other calculations
Implementation:
1. Add filter to view
2. Right-click filter → Add to Context
3. Filter icon changes to grey
4. Affects all subsequent calculations
5. Improves performance for large datasets
```

Conditional Filtering

```
Dynamic Date Filtering:
// Relative date filter
Order Date >= DATEADD('month', -[Months Parameter], TODAY())

// Fiscal year filter
Fiscal Year =
IF MONTH([Order Date]) >= 4
THEN YEAR([Order Date]) + 1
ELSE YEAR([Order Date])
END
```

Set-Based Filtering

```
Dynamic Sets:
1. Create set based on condition
2. Use "Top N by measure" for dynamic ranking
3. Reference set in other calculations
4. Combine sets with Boolean logic

Example - Top Customers Set:
Top Customers = [Customer Revenue] >= [Revenue Threshold Parameter]
```

Dashboard Actions

Filter Actions

```
Cross-Dashboard Filtering:
1. Dashboard → Actions → Add Filter Action
2. Source: Select source sheet
3. Target: Select target sheets
4. Run action on: Select, Hover, or Menu
5. Configure clearing behavior
6. Test interaction behavior
```

Highlight Actions

```
Visual Highlighting:
1. Dashboard → Actions → Add Highlight Action
2. Configure source and target
3. Select fields to highlight on
4. Choose highlight colour
5. Test highlighting behavior
```

URL Actions

```
External Navigation:
1. Dashboard → Actions → Add URL Action
2. Configure source worksheet
3. Build dynamic URL with field values:
   https://example.com/search?query=<Customer Name>
4. Set target (New tab, Same tab, etc.)
5. Test URL generation
```

Parameter Actions

```
Dynamic Parameter Updates:
1. Dashboard → Actions → Add Parameter Action
2. Select source worksheet
3. Choose target parameter
4. Map source field to parameter
5. Enable dynamic parameter updates via selection
```

Publishing and Sharing

Tableau Server and Tableau Cloud

Platform Comparison

```
Tableau Server (On-premises):
- Full control over infrastructure
- Enhanced security options
- Custom integrations possible
- Requires IT maintenance
- Custom authentication systems

Tableau Cloud (SaaS):
- Managed by Tableau
- Automatic updates and maintenance
- Quick setup and deployment
- Standard security features
- Integration with cloud services
```

Publishing Workflow

Preparing for Publication

```
Pre-Publication Checklist:
1. Optimize workbook performance
2. Remove unused fields and data sources
3. Test all interactive features
4. Validate data accuracy
5. Configure extract refresh schedules
6. Document data sources and calculations
7. Test on target platform
```

Publishing Process

```
From Tableau Desktop:
1. Server → Publish Workbook
2. Select target server/site
3. Choose project location
4. Configure permissions
5. Set data source publishing options
6. Add description and tags
7. Choose sheets to publish
8. Configure extract refresh schedule
```

Data Source Management

```
Publishing Data Sources:
1. Data → Publish Data Source
2. Configure authentication
3. Set refresh schedule
4. Manage permissions
5. Add certification if applicable

Authentication Options:
- Embedded credentials
- Viewer credentials
- Server/database authentication
- OAuth for cloud services
```

Security and Permissions

User Roles and Capabilities

```
Server Admin:
- Full system administration
- User and site management
- Content and permission management

Site Admin:
- Site-level administration
- User management within site
- Project and content management

Creator:
- Full content creation capabilities
- Connect to data sources
- Publish workbooks and data sources

Explorer:
- View and interact with published content
- Create views and custom analysis
- Limited publishing capabilities

Viewer:
- View published content only
- No editing or creation capabilities
- Basic interaction with dashboards
```

Row-Level Security (RLS)

```
User Filter Implementation:
1. Create calculated field for user filtering:
```

```
   User Filter = USERNAME() = [Assigned User]

2. Add calculated field to Filters shelf
3. Set to True
4. Apply to all relevant worksheets
5. Test with different user accounts

Dynamic Security:
// Role-based filtering
Role Filter =
CASE
WHEN USERNAME() = "admin@company.com" THEN TRUE
WHEN USERNAME() = "manager@company.com" AND [Region] = "North" THEN TRUE
WHEN USERNAME() = "sales@company.com" AND [Sales Rep] = USERNAME() THEN
TRUE
ELSE FALSE
END
```

Project-Level Security

```
Content Organization:
1. Create projects for different departments
2. Set project permissions
3. Configure default permissions for new content
4. Implement nested project structure
5. Use project descriptions and images

Permission Inheritance:
- Projects inherit site permissions
- Content inherits project permissions
- Explicit permissions override inherited
- Test permission cascading
```

Performance Optimization

Extract Optimization

```
Extract Configuration:
1. Filter unnecessary data at source
2. Remove unused fields
3. Aggregate data when possible
4. Use incremental refresh for large datasets
5. Configure optimal refresh schedule

Extract Refresh Strategies:
- Full refresh: Complete data replacement
- Incremental refresh: Add new/changed data only
- Live to extract: Convert for better performance
```

Dashboard Performance

```
Performance Best Practices:
1. Limit number of marks displayed
2. Use data source filters
3. Avoid complex calculated fields in views
4. Minimize cross-data source joins
5. Use context filters appropriately
6. Optimize LOD expressions

Performance Recording:
1. Help → Settings and Performance → Start Performance Recording
2. Interact with dashboard
3. Help → Settings and Performance → Stop Performance Recording
4. Analyze query execution times
5. Identify bottlenecks and optimize
```

Hands-on Lab Session 2

Lab Exercise: Advanced Northwind Executive Dashboard

Objective: Create an executive dashboard with advanced analytics and interactivity

Part 1: Advanced Data Preparation

Task 1.1: Create Advanced Calculated Fields

```
// Customer Segmentation (RFM Analysis)
Days Since Last Order = DATEDIFF('day',
    {FIXED [Customer Name] : MAX([Order Date])},
    TODAY()
)

Order Frequency = {FIXED [Customer Name] : COUNTD([Order Id])}

Customer Lifetime Value = {FIXED [Customer Name] : SUM([Revenue])}

// RFM Scoring
Recency Score =
IF [Days Since Last Order] <= 30 THEN 5
ELSEIF [Days Since Last Order] <= 60 THEN 4
ELSEIF [Days Since Last Order] <= 120 THEN 3
ELSEIF [Days Since Last Order] <= 180 THEN 2
ELSE 1
END

Frequency Score =
IF [Order Frequency] >= 10 THEN 5
ELSEIF [Order Frequency] >= 7 THEN 4
```

```
ELSEIF [Order Frequency] >= 5 THEN 3
ELSEIF [Order Frequency] >= 3 THEN 2
ELSE 1
END

Monetary Score =
IF [Customer Lifetime Value] >= 5000 THEN 5
ELSEIF [Customer Lifetime Value] >= 3000 THEN 4
ELSEIF [Customer Lifetime Value] >= 1500 THEN 3
ELSEIF [Customer Lifetime Value] >= 500 THEN 2
ELSE 1
END

Customer Segment =
IF [Recency Score] >= 4 AND [Frequency Score] >= 4 AND [Monetary Score] >=
4 THEN "Champions"
ELSEIF [Recency Score] >= 3 AND [Frequency Score] >= 3 AND [Monetary
Score] >= 3 THEN "Loyal Customers"
ELSEIF [Recency Score] >= 3 AND [Frequency Score] <= 2 THEN "Potential
Loyalists"
ELSEIF [Recency Score] <= 2 AND [Frequency Score] >= 3 THEN "At Risk"
ELSEIF [Recency Score] <= 2 AND [Frequency Score] <= 2 AND [Monetary
Score] >= 3 THEN "Cannot Lose Them"
ELSE "Others"
END

// Time Intelligence
Previous Year Revenue =
LOOKUP(ZN([Revenue]), -12)

YoY Growth =
([Revenue] - [Previous Year Revenue]) / [Previous Year Revenue]

Quarter to Date =
IF [Order Date] <= TODAY() AND DATEPART('quarter', [Order Date]) =
DATEPART('quarter', TODAY())
THEN [Revenue] END

// Product Analysis
Product Performance Score =
([Revenue] - WINDOW_AVG([Revenue])) / WINDOW_STDEV([Revenue])

ABC Classification =
IF [Product Performance Score] >= 1 THEN "A"
ELSEIF [Product Performance Score] >= 0 THEN "B"
ELSE "C"
END

// Employee Performance
Employee Revenue Rank = RANK([Revenue], 'desc')

Employee Efficiency = [Revenue] / {FIXED [Employee Name] : COUNTD([Order
Id])}
```

Task 1.2: Create Parameters for Interactivity

```
Parameters to create:
1. Date Range Parameter:
   - Name: "Analysis Period"
   - Data type: String
   - List values: "Last 30 Days", "Last 3 Months", "Last 6 Months", "Last
Year", "All Time"

2. Top N Parameter:
   - Name: "Top N Products"
   - Data type: Integer
   - Range: 5 to 20
   - Default: 10

3. Metric Selector Parameter:
   - Name: "KPI Metric"
   - Data type: String
   - List values: "Revenue", "Quantity Sold", "Order Count", "Average
Order Value"

4. Target Revenue Parameter:
   - Name: "Revenue Target"
   - Data type: Float
   - Range: 0 to 1000000
   - Default: 500000
```

Task 1.3: Create Parameter-Driven Calculated Fields

```
// Dynamic date filtering
Dynamic Date Filter =
CASE [Analysis Period]
WHEN "Last 30 Days" THEN [Order Date] >= DATEADD('day', -30, TODAY())
WHEN "Last 3 Months" THEN [Order Date] >= DATEADD('month', -3, TODAY())
WHEN "Last 6 Months" THEN [Order Date] >= DATEADD('month', -6, TODAY())
WHEN "Last Year" THEN [Order Date] >= DATEADD('year', -1, TODAY())
WHEN "All Time" THEN TRUE
END

// Dynamic KPI calculation
Dynamic KPI =
CASE [KPI Metric]
WHEN "Revenue" THEN [Revenue]
WHEN "Quantity Sold" THEN [Quantity]
WHEN "Order Count" THEN 1
WHEN "Average Order Value" THEN [Revenue]
END

// Target achievement
Target Achievement =
```

```
[Revenue] / [Revenue Target]

Performance Status =
IF [Target Achievement] >= 1.1 THEN "Exceeding"
ELSEIF [Target Achievement] >= 0.9 THEN "On Track"
ELSE "Below Target"
END
```

Part 2: Executive KPI Dashboard

Task 2.1: Create KPI Overview Worksheet

```
KPI Cards Layout:
1. Total Revenue Card:
   - Add Dynamic KPI to Text
   - Filter by Dynamic Date Filter = True
   - Format as large text with currency
   - Add colour formatting based on Target Achievement

2. YoY Growth Card:
   - Add YoY Growth to Text
   - Format as percentage
   - Add positive/negative colour encoding
   - Include sparkline trend

3. Customer Count Card:
   - Count distinct Customer Name
   - Apply Dynamic Date Filter
   - Add comparison to previous period

4. Average Order Value Card:
   - Calculate Revenue / Order Count
   - Apply Dynamic Date Filter
   - Format as currency
```

Task 2.2: Advanced Trend Analysis

```
Revenue Trend with Forecast:
1. Create line chart with Order Date (continuous, month) and Revenue
2. Add Analytics → Forecast for next 3 months
3. Include confidence bands
4. Add reference line for revenue target
5. Colour-code actual vs forecast

Decomposition Analysis:
1. Create combo chart showing:
   - Revenue (bars)
   - Customer count (line, secondary axis)
   - Average order value (line, secondary axis)
```

```
2. Enable dual-axis for line charts
3. Synchronize axes where appropriate
```

Task 2.3: Customer Segmentation Analysis

```
RFM Segmentation Scatter Plot:
1. Create scatter plot:
   - X-axis: Recency Score
   - Y-axis: Frequency Score
   - Size: Monetary Score
   - Colour: Customer Segment
2. Add customer count labels
3. Configure tooltip with customer examples

Customer Value Distribution:
1. Create histogram of Customer Lifetime Value
2. Add reference lines for quartiles
3. Colour-code by Customer Segment
4. Include statistical summary
```

Part 3: Product and Sales Analysis

Task 3.1: Product Performance Matrix

```
Product Portfolio Analysis:
1. Create scatter plot:
   - X-axis: Revenue Growth Rate (YoY)
   - Y-axis: Market Share (Revenue / Total Revenue)
   - Size: Total Revenue
   - Colour: ABC Classification
2. Add quadrant reference lines
3. Label high-performing products
4. Create BCG-style matrix interpretation

Category Performance Comparison:
1. Create bullet chart for each category
2. Show actual vs target revenue
3. Include performance zones (good/fair/poor)
4. Rank categories by performance
```

Task 3.2: Sales Team Analysis

```
Employee Performance Dashboard:
1. Create heat map:
   - Rows: Employee Name
   - Columns: Month
   - Colour: Revenue achievement vs target
```

```
2. Add sparklines for trend visualization
3. Include ranking and percentile information

Territory Analysis:
1. Create map visualization:
   - Location: Country/Region
   - Colour: Revenue performance
   - Size: Customer count
2. Add drill-down capability
3. Include market penetration metrics
```

Part 4: Interactive Dashboard Assembly

Task 4.1: Create Executive Dashboard

```
Dashboard Layout:
1. Header section (10% height):
   - Dashboard title
   - Parameter controls
   - Last updated timestamp

2. KPI section (20% height):
   - Four KPI cards in horizontal layout
   - Traffic light indicators

3. Trends section (35% height):
   - Revenue trend chart (left 60%)
   - Customer segmentation (right 40%)

4. Analysis section (35% height):
   - Product performance matrix (left 50%)
   - Geographic analysis (right 50%)
```

Task 4.2: Add Advanced Interactivity

```
Dashboard Actions:
1. Filter Action - Product Focus:
   - Source: Product performance matrix
   - Target: All other charts
   - Clear filter option available

2. Highlight Action - Time Focus:
   - Source: Revenue trend chart
   - Target: KPI cards and other time-based charts
   - Show temporal relationships

3. Parameter Actions:
   - Customer segment selection updates Top N parameter
   - Product selection updates focus filters
```

```
Navigation:
1. Add navigation buttons for different dashboard pages
2. Create bookmark-style navigation
3. Include breadcrumb trail
4. Add reset filters button
```

Task 4.3: Mobile-Responsive Design

```
Device-Specific Layouts:
1. Create phone layout:
   - Single column design
   - Priority KPIs only
   - Simplified interactions

2. Create tablet layout:
   - Two-column design
   - Essential visualizations
   - Touch-friendly controls

3. Test responsiveness:
   - Preview on different device sizes
   - Adjust sizing and positioning
   - Optimize loading performance
```

Part 5: Advanced Analytics Integration

Task 5.1: Statistical Analysis

```
Correlation Analysis:
1. Create correlation matrix for key metrics
2. Show relationships between:
   - Revenue and customer satisfaction
   - Product price and demand
   - Seasonal patterns and sales

Cohort Analysis:
1. Create customer acquisition cohorts by month
2. Track retention rates over time
3. Calculate customer lifetime value by cohort
4. Visualize retention heatmap
```

Task 5.2: Predictive Analytics

```
Sales Forecasting:
1. Use Tableau's forecasting features
2. Create multiple forecast scenarios
```

```
3. Include confidence intervals
4. Compare forecast accuracy

Trend Analysis:
1. Add polynomial trend lines
2. Calculate R-squared values
3. Identify seasonal patterns
4. Predict future performance
```

Part 6: Publishing and Governance

Task 6.1: Prepare for Publication

```
Performance Optimization:
1. Remove unused fields and calculations
2. Optimize extract configuration
3. Test dashboard performance
4. Document data lineage

Quality Assurance:
1. Validate all calculations
2. Test interactive features
3. Check mobile responsiveness
4. Verify security filters
```

Task 6.2: Publish to Tableau Server/Cloud

```
Publication Process:
1. Configure data source authentication
2. Set extract refresh schedule
3. Assign appropriate permissions
4. Create project structure
5. Add metadata and descriptions

User Training Materials:
1. Create user guide documentation
2. Record demonstration videos
3. Schedule training sessions
4. Establish support procedures
```

Task 6.3: Monitoring and Maintenance

```
Usage Analytics:
1. Monitor dashboard usage patterns
2. Track user engagement metrics
3. Identify performance issues
4. Plan improvement iterations
```

```
Governance Framework:
1. Establish data update procedures
2. Create change management process
3. Document calculation definitions
4. Maintain version control
```

Conclusion

This guide provides the foundation for mastering Tableau, from basic CSV connections through advanced enterprise dashboard deployment. The hands-on exercises using Northwind data ensure practical experience with real-world business scenarios.

Key Takeaways

Week 5 Fundamentals:

- Master CSV data import and joining techniques
- Apply data preparation and cleaning methods
- Create meaningful basic visualizations
- Understand Tableau Desktop interface and workflow

Week 6 Advanced Features:

- Design sophisticated interactive dashboards
- Implement advanced analytics and calculations
- Configure enterprise publishing and security
- Establish governance and maintenance procedures

Best Practices Summary

Data Preparation:

- Validate data quality before visualization
- Use appropriate data types and formats
- Document data sources and transformations
- Implement efficient join strategies

Visualization Design:

- Follow visual design principles
- Ensure accessibility and responsiveness
- Implement intuitive navigation
- Optimize for target audience

Dashboard Development:

- Plan information architecture carefully
- Use consistent formatting and branding
- Test thoroughly across devices

- Document functionality and usage

Enterprise Deployment:

- Configure appropriate security measures
- Establish refresh and maintenance schedules
- Monitor usage and performance
- Provide user training and support

Next Steps

1. **Practice Regularly**: Apply techniques to your own datasets
2. **Stay Current**: Follow Tableau's quarterly updates and new features
3. **Join Community**: Participate in Tableau User Groups and forums
4. **Advanced Training**: Explore Tableau Server administration and advanced analytics
5. **Certification**: Consider Tableau Desktop and Server certification paths

Additional Resources

- **Tableau Public**: Free platform for practicing and sharing visualizations
- **Tableau Community**: User forums and knowledge base
- **Tableau Training**: Official training courses and certification programs
- **Tableau Conference**: Annual user conference with advanced sessions
- **Tableau Help**: Documentation and tutorials

Remember: Mastery of Tableau comes through consistent practice and experimentation with different data sources and visualization techniques. Focus on understanding your audience's needs and telling compelling stories with data.