Contents

[Week 1: Welcome to R — Fundamentals & Foundations 2](#_Toc198285665)

[Week 2: Data Wrangling with dplyr — Speak Data Fluently 2](#_Toc198285666)

[Week 3: Data Cleaning, Tidying & Time Handling 3](#_Toc198285667)

[Week 4: Data Visualization with ggplot2 and plotly 4](#_Toc198285668)

[Week 5: Statistics, Hypothesis Testing & Modeling Foundations 6](#_Toc198285669)

[Week 6: Machine Learning with tidymodels, caret, and mlr3 7](#_Toc198285670)

[Week 7: Big Data & Pipelines with sparklyr, arrow, and targets 9](#_Toc198285671)

[Week 8: Advanced Data Wrangling, APIs & Web Scraping 11](#_Toc198285672)

[Week 9: Capstone Project (Part 1) — From Raw Data to Insightful Dashboard 12](#_Toc198285673)

[Week 10: Capstone Project (Part 2) — Finalization, Automation & Dashboard Reporting 14](#_Toc198285674)

[Week 11: Bonus – Portfolio, Job Prep & Advanced R Topics 15](#_Toc198285675)

## Week 1: Welcome to R — Fundamentals & Foundations

**Objective**: Learn R syntax, basic data types, structures, and functions

**Key Topics & Tools:**

* RStudio setup
* Vectors, lists, matrices, data frames
* Basic arithmetic, indexing, functions
* Built-in datasets
* Tools: Base R, RStudio, Swirl

**Retention Techniques:**

* **Story**: R as your data cooking kitchen — vectors are ingredients, functions are recipes.
* **Mnemonic**: *DEF* = Define, Execute, Function
* **Visual**: Lists = “bento box” 🍱, data frame = Excel sheet

|  |  |  |
| --- | --- | --- |
| **Day** | **Focus** | **Activity** |
| 1 | RStudio Setup + Syntax | Install R + RStudio, write your first R script, explore basic math |
| 2 | Vectors | Create and subset vectors, use logical filtering |
| 3 | Lists, Matrices, Factors | Work with structured but varied data |
| 4 | Data Frames | Understand the most-used data structure in R |
| 5 | Indexing & Subsetting | Learn how to retrieve and filter data |
| 6 | Writing Functions | Define and use basic functions |
| 7 | Mini-Project | Create a basic spending tracker + summary table |

**Practice Project:**

* **"Personal Budget Tracker"** using data.frame, basic stats, and plots

## Week 2: Data Wrangling with dplyr — Speak Data Fluently

**Objective**: Use dplyr verbs to manipulate and summarize real data

**Key Topics & Tools:**

* dplyr: filter, select, mutate, summarise, group\_by, arrange, joins
* The %>% pipe operator
* Built-in datasets: mtcars, starwars, gapminder
* Tools: dplyr, readr, janitor

**Retention Techniques:**

* **Mnemonic**: *FSSMJ* = Filter, Select, Summarise, Mutate, Join
* **Story**: Treat data as a restaurant order:
  + **Filter** the menu 🍽
  + **Select** your dishes 🍝
  + **Mutate** ingredients 👨‍🍳
  + **Summarize** your bill 💵
* **Visual**: %>% = "and then..."

|  |  |  |
| --- | --- | --- |
| **Day** | **Focus** | **Activity** |
| 8 | Intro to dplyr + pipe | Learn core verbs and chaining with %>% |
| 9 | Filtering & Selecting | Extract rows and columns based on conditions |
| 10 | Mutating & Arranging | Create new variables, reorder rows |
| 11 | Summarising & Grouping | Group data, compute summaries |
| 12 | Joins | Combine data frames using left\_join, inner\_join, etc. |
| 13 | Real-World Airbnb Dataset | Clean and summarize NYC Airbnb data |
| 14 | Mini Project: Data Doctor | Clean, transform, and summarize any real dataset (Kaggle or built-in) |

**Practice Project:**

* **"Airbnb Insights"** — filter listings, compute stats by neighborhood, apply joins

## Week 3: Data Cleaning, Tidying & Time Handling

**Objective: Clean messy data, reshape it, and handle dates/times like a pro.**

**Key Topics & Tools:**

* Cleaning: janitor, dplyr
* Tidying: tidyr — pivot\_longer(), pivot\_wider(), separate(), unite()
* Dates: lubridate — ymd(), mdy(), floor\_date(), wday(), intervals()
* Real-world messy datasets (e.g., COVID-19, weather, retail)

**Retention Techniques:**

* **Story**: Imagine your dataset is a closet 🧥 — tidying = sorting clothes by type, size, season.
* **Mnemonic**:
  + **CLEAN** = Column names, Logical values, Empty rows, Arrange rows, Nulls
  + **TIDY** = Tidy = *One row per observation, one column per variable*
* **Visual**: pivot\_longer() = *stack columns*, pivot\_wider() = *spread rows into columns*

|  |  |  |
| --- | --- | --- |
| **Day** | **Focus** | **Activity** |
| **15** | Cleaning with janitor | Use clean\_names(), remove empty rows, identify duplicates |
| **16** | Tidy Data Principles | Understand “tidy” format (long vs. wide), use tidyr |
| **17** | Reshaping Data | Use pivot\_longer() and pivot\_wider() on messy health data |
| **18** | Splitting & Combining Columns | Use separate() and unite() for string columns |
| **19** | Working with Dates | Parse and format dates with lubridate, extract day/month/year |
| **20** | Time-Based Features | Create week, month, and duration-based summaries |
| **21** | Project: “Messy to Model-Ready” | Take a messy real dataset (e.g. COVID-19), clean and transform it into tidy format |

**Practice Project:**

**“COVID Trends Cleanup”**

* Dataset: [Johns Hopkins COVID-19 dataset](https://github.com/CSSEGISandData/COVID-19)
* Tasks:
  + Clean column names
  + Tidy the format
  + Parse date columns
  + Summarize daily new cases by country
  + Plot trends over time

**Tools & Resources:**

* **Packages**: tidyr, janitor, lubridate, dplyr
* 📘 [*R for Data Science* — Chapters 12–16](https://r4ds.hadley.nz/)
* 📄 Cheatsheets:
  + [Data Wrangling Cheatsheet](https://posit.co/resources/cheatsheets/)
  + [Lubridate Reference](https://lubridate.tidyverse.org/)
* 🎥 [Tidy Data in R - YouTube](https://www.youtube.com/watch?v=9bUuVqPjm_U)

## Week 4: Data Visualization with ggplot2 and plotly

**Objective: Create clear, attractive, and insightful static and interactive plots using the grammar of graphics.**

**Key Topics & Tools:**

* ggplot2: Grammar of graphics (layers, aesthetics, geoms)
* Plot types: scatter, line, bar, histogram, boxplot, facet
* Themes and customization
* plotly: Interactive plots
* Datasets: gapminder, diamonds, nycflights13, your own cleaned data

**Retention Techniques:**

* **Story**: Think of ggplot2 as LEGO blocks 🧱 — you build plots layer by layer.
* **Mnemonic**:
  + **A.G.G.** = Aesthetics (what to show), Geoms (how to show), Grammar (rules)
* **Analogy**:
  + aes() = like a map legend (what variable goes where)
  + + = like Photoshop layers — add one element at a time
* **Chunking**: Break visual building into 3 key parts:
  + - Data + aesthetics
    - Geometric object
    - Labels, scales, and themes

|  |  |  |
| --- | --- | --- |
| **Day** | **Focus** | **Activity** |
| **22** | Intro to ggplot2 | Understand ggplot(data) + aes(x, y) + geom\_\* structure |
| **23** | Scatter & Line Plots | Visualize relationships and trends over time |
| **24** | Bar, Column & Histograms | Compare categories and distributions |
| **25** | Boxplots & Density Plots | Compare numeric distributions between groups |
| **26** | Facets, Themes & Colors | Add facets (subplots), improve aesthetics, use themes like theme\_minimal() |
| **27** | Interactive Plots with plotly | Convert ggplots to interactive versions, hover & zoom |
| **28** | Project: “Tell a Story with Data” | Use any dataset to tell a compelling story with 3–5 well-designed visualizations |

**Practice Project:**

**"Data Storyboard Challenge"**

* Choose a dataset (e.g. Gapminder, your COVID/Airbnb data)
* Build:
  + 1 line chart showing time trends
  + 1 bar chart comparing categories
  + 1 boxplot of group-wise stats
  + Add tooltips & zoom using plotly

**Tools & Resources:**

* 📦 Packages: ggplot2, plotly, dplyr, viridis, scales
* 📘 Book: [*R for Data Science* — Chapter 3 & 28 (ggplot2)](https://r4ds.hadley.nz/data-visualize.html)
* 📄 Cheatsheet: [ggplot2 Cheatsheet (PDF)](https://posit.co/resources/cheatsheets/)
* 📊 Explore: [Gapminder Dataset](https://www.gapminder.org/data/)
* 🎥 YouTube: *ggplot2 from Scratch* (Data School, StatQuest)

## Week 5: Statistics, Hypothesis Testing & Modeling Foundations

**Objective: Learn statistical thinking, hypothesis testing, and build your first regression models in R.**

**Key Topics & Tools:**

* Descriptive statistics: mean, median, variance, sd, IQR
* Probability distributions: normal, binomial
* Inferential statistics: confidence intervals, p-values
* Hypothesis testing: t-test, chi-squared, ANOVA
* Intro to modeling: Linear regression with lm()
* Tools: Base R, infer, broom, ggpubr, ggstatsplot

**Retention Techniques:**

* **Story**: Statistics is your **detective toolkit** 🕵️ — you collect evidence (data) to support or reject claims.
* **Analogy**:
  + Hypothesis test = *court trial*:
    - H₀ = "Innocent until proven guilty"
    - p-value = strength of evidence
* **Mnemonic**:
  + **CIPT** = Confidence Interval, p-value, T-test, Tidy model
* **Visuals**:
  + Use bell curves to explain normal distributions
  + Show regression lines overlaying scatter plots

|  |  |  |
| --- | --- | --- |
| **Day** | **Focus** | **Activity** |
| **29** | Descriptive Stats | Use summary(), mean(), sd(), quantile() |
| **30** | Distributions & Visuals | Visualize normal vs skewed distributions, use rnorm(), hist() |
| **31** | Hypothesis Testing Basics | Learn p-values, confidence intervals, null vs alternative hypotheses |
| **32** | t-tests, chi-square, ANOVA | Use t.test(), chisq.test(), aov() with visual explanations |
| **33** | Linear Regression | Fit models with lm(), visualize with ggplot2, interpret coefficients |
| **34** | Model Evaluation | Residual plots, R², assumptions of linear regression |
| **35** | Mini Project: “Does Price Affect Sales?” | Use regression and hypothesis tests on a real retail dataset |

**Practice Project:**

📊 **"Retail Analytics Report"**

* Dataset: [Online Retail Dataset](https://archive.ics.uci.edu/ml/datasets/Online+Retail)
* Tasks:
  + Describe key numeric variables (mean, sd, etc.)
  + Run t-tests (e.g., avg spend UK vs France)
  + Model sales quantity ~ price using linear regression
  + Report: interpret R² and coefficients

**Tools & Resources:**

* 📦 Packages: broom, ggpubr, infer, ggstatsplot, tidyverse
* 📘 [StatQuest YouTube Series (Highly Recommended!)](https://www.youtube.com/user/joshstarmer)
* 📘 [*R for Data Science* — Model Basics](https://r4ds.hadley.nz/model-basics.html)
* 📄 Cheatsheet: [Modeling with broom & infer](https://www.rstudio.com/resources/cheatsheets/)

## Week 6: Machine Learning with tidymodels, caret, and mlr3

**Objective: Learn how to build, train, tune, and evaluate machine learning models using R’s best ML ecosystems.**

**Key Topics & Tools:**

* ML pipeline: training, testing, validation
* Supervised learning: classification & regression
* tidymodels: a unified ML framework (recipes, workflows, resampling, metrics)
* caret: classic and versatile ML package
* mlr3: modern ML with high performance
* Model types: linear regression, logistic regression, decision trees, random forests, k-NN
* Tools: tidymodels, caret, mlr3, yardstick, vip, parsnip

**Retention Techniques:**

* **Story**: Think of ML in R as building a **robot chef** 🤖🍳 that learns recipes (patterns) from past dishes (data) to cook better meals (predictions).
* **Mnemonic**:
  + **FREMT** = Formula, Recipe, Engine, Metrics, Tune
* **Analogy**:
  + **Train/Test Split** = like giving a student past exams to study (train), then giving a new one (test)
  + **Overfitting** = student memorizes past questions but fails a new test

|  |  |  |
| --- | --- | --- |
| **Day** | **Focus** | **Activity** |
| **36** | Intro to ML + tidymodels | Build a workflow: recipe() → model\_spec() → workflow() |
| **37** | Classification: Logistic Regression | Classify Titanic survival or loan defaults |
| **38** | Decision Trees & Random Forests | Use rpart and ranger for tree-based models |
| **39** | Model Evaluation Metrics | Use yardstick for accuracy, precision, recall, ROC |
| **40** | Hyperparameter Tuning | Use tune\_grid() and cross-validation |
| **41** | Using caret and mlr3 | Compare classic caret vs modern mlr3 on same dataset |
| **42** | Project: “Churn Prediction Engine” | Build a full ML pipeline to predict customer churn |

**Practice Project:**

**"Customer Churn Predictor"**

* Dataset: Telco Customer Churn (Kaggle)
* Tasks:
  + Preprocess data with recipe()
  + Build and compare logistic regression and random forest
  + Tune hyperparameters
  + Evaluate using accuracy, ROC curve, confusion matrix
  + Present model insights using variable importance

**Tools & Resources:**

* 📦 Packages: tidymodels, parsnip, workflows, recipes, yardstick, caret, mlr3
* 📘 [Tidymodels website](https://www.tidymodels.org/)
* 🎥 [StatQuest ML Playlist](https://www.youtube.com/playlist?list=PLblh5JKOoLUIxGDQs4LFFD--41Vzf-ME1)
* 📄 Cheatsheets:
  + [Tidymodels Cheatsheet (PDF)](https://posit.co/resources/cheatsheets/)
  + [Caret Quickstart](https://topepo.github.io/caret/)

## Week 7: Big Data & Pipelines with sparklyr, arrow, and targets

**Objective: Learn how to process big data efficiently and build scalable, automatable workflows and pipelines in R.**

**Key Topics & Tools:**

* Big data formats: Parquet, Arrow
* Working with large datasets: sparklyr for distributed processing
* Workflow automation and reproducibility: targets, drake
* Lazy evaluation, memory efficiency, pipeline logic
* Practical use cases: ETL pipelines, production-ready data science workflows

**Retention Techniques:**

* **Story**: Imagine you’re building a **data factory** 🏭. Big data needs forklifts (Spark), efficient packaging (Arrow), and conveyor belts (Targets) to run smoothly.
* **Mnemonic**:
  + **SPA** = Spark, Parquet, Arrow — tools to Scale
  + **TARP** = Targets Automates Reproducible Pipelines
* **Visuals**:
  + Pipeline = a flowchart of steps: input → transform → model → report
  + Arrow = compressed fast-shipping boxes 📦
  + Spark = distributed engine with workers 🚜

|  |  |  |
| --- | --- | --- |
| **Day** | **Focus** | **Activity** |
| **43** | Intro to Big Data & arrow | Read/write large Parquet files, memory-efficient I/O |
| **44** | Scalable Processing with sparklyr | Connect to Spark, run dplyr-style code on big data |
| **45** | Working with Remote Data | Load & process data from cloud-like environments |
| **46** | Pipeline Design Concepts | Understand DAGs (Directed Acyclic Graphs), pipeline structure |
| **47** | Build Pipelines with targets | Automate multi-step R workflows: raw → clean → model |
| **48** | Combining Big Data + Pipelines | Use arrow and targets together in a pipeline |
| **49** | Project: “Retail ETL & ML Pipeline” | Build an end-to-end ETL pipeline with Spark & Targets |

**Practice Project:**

🏗️ **“Retail ETL + ML Pipeline”**

* Dataset: Simulate a large CSV (>1M rows) or use [NYC Taxi Data](https://www1.nyc.gov/site/tlc/about/tlc-trip-record-data.page)
* Tasks:
  + Load with arrow or via sparklyr
  + Clean and transform the data
  + Train a model (e.g., predict tip %)
  + Automate everything using targets

**Tools & Resources:**

* 📦 Packages: sparklyr, arrow, targets, fst, data.table, dplyr
* 📘 [Targets Documentation](https://books.ropensci.org/targets/)
* 📘 [sparklyr Guide](https://spark.rstudio.com/)
* 📄 Cheatsheets:
  + [Sparklyr Cheatsheet](https://spark.rstudio.com/cheatsheet/)
  + [Targets Cheatsheet (PDF)](https://github.com/ropensci/targets)
* 🎥 YouTube: *Using targets in R* (RStudio YouTube Channel)

## Week 8: Advanced Data Wrangling, APIs & Web Scraping

**Objective: Master high-performance data transformation, handle messy data, and extract data from APIs & web pages using R.**

**Key Topics & Tools:**

* High-performance data wrangling: data.table, dplyr, janitor
* String cleaning, missing values, date handling
* Working with nested data (JSON, lists, APIs)
* Web scraping with rvest
* Working with REST APIs using httr, jsonlite
* Real-world use cases: live financial data, weather, news, public APIs

**Retention Techniques:**

* **Story**: Think of messy data like a **dirty kitchen** 👨‍🍳 — your job is to clean, prep, and make it ready for use.
* **Mnemonic**:
  + **STICKY** = Strings, Tables, Imputing (missing), Columns, Klean (janitor), Ymd (dates)
  + **HERO** = httr, extract, rvest, organize
* **Analogy**:
  + API = digital waiter: you “order” data, and it “serves” you back
  + Web scraping = treasure hunting on a website 🕵️‍♂️

|  |  |  |
| --- | --- | --- |
| **Day** | **Focus** | **Activity** |
| **50** | High-Performance Wrangling | Use data.table for fast filtering, grouping, joins |
| **51** | String & Text Cleaning | Use stringr, janitor, tidyr, regex for messy text |
| **52** | Missing, Dates, & Nested Data | Handle NA, impute, clean dates (lubridate), unpack JSON |
| **53** | Intro to APIs | Use httr to GET/POST data from public APIs (OpenWeather, etc.) |
| **54** | JSON Handling & Parsing | Use jsonlite::fromJSON() to work with nested structures |
| **55** | Web Scraping with rvest | Extract tables, headlines, or listings from real websites |
| **56** | Project: “Real-World Data Harvester” | Scrape or call API to build a real-time dataset (weather, stock, news, etc.) |

**Practice Project:**

🌐 **"Live Data Harvester"**

* Pick 1:
  + Scrape latest news headlines from [BBC](https://www.bbc.com/news)
  + Pull live weather data from OpenWeatherMap API
  + Extract trending repositories from GitHub using its API
* Tasks:
  + Retrieve and clean the data
  + Transform into tidy format
  + Visualize (bar chart, line chart, etc.)
  + Automate with targets

**Tools & Resources:**

* 📦 Packages: data.table, janitor, lubridate, stringr, rvest, httr, jsonlite
* 📘 [API + Web Scraping with R Tutorial (R-bloggers)](https://www.r-bloggers.com/2019/04/accessing-data-from-apis-in-r/)
* 📘 [rvest Web Scraping Guide](https://rvest.tidyverse.org/)
* 📄 Cheatsheets:
  + [Stringr](https://stringr.tidyverse.org/)
  + [rvest Cheatsheet (PDF)](https://github.com/rstudio/cheatsheets/raw/main/rvest.pdf)

## Week 9: Capstone Project (Part 1) — From Raw Data to Insightful Dashboard

**Objective: Apply all the R skills you’ve learned so far to a real-world, multi-stage data analytics and modeling project with full automation and reporting.**

**What You’ll Build:**

An **end-to-end data product** that:

* Extracts & cleans large data (CSV/API/web)
* Builds visual dashboards (static + interactive)
* Creates ML models to predict or explain outcomes
* Automates everything with pipelines
* Generates dynamic reports or dashboards

**Project Theme (Choose One):**

**Option 1: Real Estate Price Predictor**

* Data: Housing prices (e.g., Kaggle House Prices Dataset)
* Goal: Predict house sale prices, visualize key features, automate the whole process

**Option 2:** **E-Commerce Sales & Churn Dashboard**

* Data: Online retail or simulated customer data
* Goal: Analyze sales patterns, model churn, and build a business dashboard

**Option 3:** **Weather + Energy Use Tracker**

* Data: API (weather) + CSV (electricity usage)
* Goal: Predict energy consumption based on weather; automate data pulls + ML

|  |  |  |
| --- | --- | --- |
| **Day** | **Focus** | **Tasks** |
| **57** | Project Planning | Define objectives, choose project, sketch pipeline |
| **58** | Data Collection | Ingest data (CSV, API, or web scraping) |
| **59** | Data Wrangling | Clean & transform using dplyr, data.table, janitor |
| **60** | EDA & Visualization | Use ggplot2, plotly, ggstatsplot for trends, patterns |
| **61** | Feature Engineering | Transform variables, handle missing values, encode categoricals |
| **62** | Modeling v1 | Build regression/classification model (with tidymodels) |
| **63** | Pipeline Setup | Automate the above using targets or drake |

**Tools You’ll Apply:**

* 📦 readr, data.table, janitor for wrangling
* 📦 ggplot2, plotly, ggthemes for visuals
* 📦 tidymodels, caret, or mlr3 for ML
* 📦 targets, arrow for pipelines and performance
* 📦 rmarkdown, flexdashboard, quarto for final reporting

**Resources:**

* 🎥 [Tidytuesday Datasets](https://github.com/rfordatascience/tidytuesday) – great real datasets
* 📘 [Quarto Dashboard Templates](https://quarto.org/docs/dashboards/)
* 📄 [Target Pipeline Guide (Beginner)](https://books.ropensci.org/targets/)

## Week 10: Capstone Project (Part 2) — Finalization, Automation & Dashboard Reporting

**Objective: Finalize, automate, and present your end-to-end data project using pipelines, dashboards, and reproducible reports.**

**What You’ll Do:**

* Wrap your workflow with targets for automation
* Build a beautiful dashboard with flexdashboard or quarto
* Generate reproducible, shareable RMarkdown reports
* Save your project to GitHub as a portfolio piece
* Present your results as if you're a data consultant

|  |  |  |
| --- | --- | --- |
| **Day** | **Focus** | **Tasks** |
| **64** | Final Model Tuning | Use cross-validation & hyperparameter tuning (tune\_grid()) |
| **65** | Feature Importance & Interpretation | Use vip, SHAP values, or coefficient analysis |
| **66** | Automating with targets | Complete pipeline: data → model → plot → report |
| **67** | Build Static Dashboard | Use flexdashboard or quarto to create a multi-page report |
| **68** | Make it Interactive | Add interactivity with plotly, DT, or filters |
| **69** | Reproducible Reporting | Create dynamic .Rmd or .qmd files with auto-generated plots, tables, insights |
| **70** | Polish, Push, Present | Final polish, push project to GitHub, write README, prepare slides or video walkthrough |

**Deliverables by End of Week 10:**

* ✅ Automated pipeline (targets)
* ✅ Interactive dashboard or report
* ✅ Documented GitHub repo with:
  + Project overview
  + Code + data folder
  + Visuals / metrics
  + Instructions to reproduce
* ✅ Presentation-ready pitch (slides, screencast, or notebook)

**🧠 Storytelling Tips:**

* Think like a consultant: *"What does this mean for the business?"*
* Tell the story of your data: problem → exploration → insight → solution
* Use visuals to highlight your key findings (one chart = 1000 words)

**📘 Tools & Resources:**

* 📦 flexdashboard, quarto, rmarkdown, DT, plotly
* 📘 [Quarto Dashboard Docs](https://quarto.org/docs/dashboards/)
* 📘 [Targets Workflow Book](https://books.ropensci.org/targets/)
* 🎥 [RMarkdown Reporting & Dashboard Tutorials](https://rmarkdown.rstudio.com/lesson-1.html)

## Week 11: Bonus – Portfolio, Job Prep & Advanced R Topics

**Objective: Position yourself as an expert with a data portfolio, prepare for jobs/freelance work, and explore advanced R capabilities.**

**Your 3 Goals for This Week:**

1. **Build Your Data Science Portfolio Website**
2. **Prepare for Job Interviews / Freelancing**
3. **Get Introduced to Advanced R Topics** (so you know what to learn next)

|  |  |  |
| --- | --- | --- |
| **Day** | **Focus** | **Tasks** |
| **71** | Set Up GitHub Pages or Quarto Website | Create a personal data science site or blog (quarto, Netlify) |
| **72** | Showcase Your Projects | Publish capstone project + 2–3 mini projects from earlier weeks |
| **73** | Resume + LinkedIn for Data Jobs | Optimize resume, write R/data-focused LinkedIn profile |
| **74** | Interview Prep – Technical | Review data science R questions (see below) + code challenges |
| **75** | Interview Prep – Case Studies | Practice “Explain your project” and business impact analysis |
| **76** | Explore Advanced R Topics | Get a taste of time series, deep learning, Shiny apps, or geospatial in R |
| **77** | Plan Your Next 30–90 Days | Create a personalized learning/career roadmap beyond this program |

**🌐 Day 71–72: Build Your Data Portfolio Website**

* Tools:
  + 📦 quarto, blogdown, or GitHub Pages
  + Free hosting on [Netlify](https://www.netlify.com/) or [GitHub Pages](https://pages.github.com/)
* What to include:
  + ✨ Hero headline: “Data Scientist with expertise in R, ML, and Pipelines”
  + 💻 Projects: Link to GitHub repos
  + 📊 Visuals: Screenshots from dashboards
  + 📚 Blog: Short write-ups explaining your capstone & process

**Day 73–75: Job Prep & Freelancing**

**📝 Resume Tips:**

* List tools: R, dplyr, ggplot2, tidymodels, sparklyr, targets
* List projects: show scope, problem solved, tools used, outcomes
* Use metrics: “Improved prediction accuracy by 18%...”

**📄 R Interview Topics:**

* Tidyverse vs base R
* Vectorized operations
* Pipe chaining (%>%)
* Grouped operations with dplyr
* Model tuning with tidymodels
* Lazy evaluation (with targets, arrow)
* Data.table vs dplyr

**🎯 Practice:**

* Mock interviews (record yourself or with a peer)
* HackerRank/LeetCode (data questions using R)
* Case: “Build a pipeline to predict sales drop using messy retail data”

**Day 76: Advanced R Topics (Intro)**

Choose 1 or 2 that interest you most for deeper learning:

|  |  |  |
| --- | --- | --- |
| **Area** | **Topics** | **Tools** |
| 🧠 Time Series | Forecasting, seasonality | fable, tsibble, prophet |
| 📍 Geospatial | Maps, GPS, geocoding | sf, leaflet, ggmap |
| 🖥️ Web Apps | Interactive dashboards | shiny, golem, shinydashboard |
| 🤖 Deep Learning | Neural nets in R | keras, torch, tidymodels |
| 📈 Experimentation | A/B testing, uplift | infer, designr, experimentr |

**Day 77: Your Next 90-Day Plan**

You’ve now built a **full-stack R portfolio**. Here's how to go beyond:

**🔁 Next 90-Day Focus Ideas:**

* 💡 **Master Shiny + Dashboards** → to become a dashboard expert
* 📦 **Package Your Code** → turn pipelines into reusable R packages
* 🤖 **Advanced ML & MLOps** → AutoML, model monitoring, CI/CD
* 🌐 **Freelance on Upwork or Toptal** → start applying with your new skills

**Bonus Tools & Resources**

* 📘 [Quarto Portfolio Tutorial](https://quarto.org/docs/websites/)
* 📘 [Data Science Resume Templates](https://www.dataquest.io/blog/data-science-resume/)
* 🎥 [R for Job Interviews Playlist (YouTube)](https://www.youtube.com/c/RProgramming)
* 💬 [r/DataScience](https://reddit.com/r/datascience) + [r/rstats](https://reddit.com/r/rstats) communities

**Final Challenge:**

✅ Present your capstone project (record yourself or do a Zoom with a friend):  
Explain your:

* Problem & objective
* Dataset & wrangling
* Visuals
* Modeling approach
* Business insight or recommendation
* Next steps or improvements