Your First 30 Days in Data Engineering

She Ships Data · Free starter kit

How this plan works

In 30 days you'll go from "what does a data engineer actually do?" to running your own small data pipeline. Plan on about 5 hours a week: one hour on most weekdays, or two longer sessions on the weekend.

Each week has one goal, a short list of things to learn and one thing to build. Tick off the boxes as you go. If a week takes you longer, that's fine. Finishing slowly still counts as finishing.

Pick your dataset on day 1. You'll use one public dataset all month, so by day 30 you have a portfolio project that's yours, not a copy of everyone else's. Good options: NYC taxi trip records, your city's open data portal, or any public dataset on a topic you care about. Choose one with at least two related tables or files, such as trips and zones.

Watching the She Ships Data episodes alongside this plan is a great way to see each idea in a real story. The episode exercises use Ripple's data, so they're extra practice, not a replacement for your project.

Before day 1, follow the setup guide in the She Ships Data wiki so your tools are ready.

By day 30 you'll have:

- Working SQL and Python for everyday data tasks
- A clear picture of how data moves from source systems into reports
- One small end-to-end pipeline on GitHub that you can show people

Week 1 (days 1–7): Speak SQL

Goal: query a real dataset and answer business questions with SQL. SQL is the language you'll use most as a data engineer, so it comes first.

Learn

- [ ] What a database, a table, a row and a column are
- [ ] `SELECT`, `WHERE`, `ORDER BY` and `LIMIT`
- [ ] Aggregations: `COUNT`, `SUM`, `AVG` with `GROUP BY` and `HAVING`

- [ ] Joins: `INNER JOIN` and `LEFT JOIN`, and why row counts change after a join
- [ ] Finding duplicate rows with `GROUP BY ... HAVING COUNT(*) > 1`

Build

- [ ] Load your dataset into DuckDB and write five queries that answer real questions, such as busiest month, top 10 categories and average value by group
- [ ] Check it for duplicate rows and write down what you find

Week 2 (days 8–14): Python for data

Goal: use Python to read messy files, clean them and write them back out. This is the glue work most pipelines are made of.

Learn

- [ ] Variables, lists, dictionaries, loops and functions (just enough, not a whole course)
- [ ] Reading and writing CSV and JSON files
- [ ] Working with tables in Python using pandas or Polars
- [ ] Cleaning: fixing data types, trimming text, handling missing values, removing duplicates
- [ ] Why Parquet is better than CSV for analytics, and how to write a Parquet file

Build

- [ ] Write a script that reads your raw dataset files, cleans them and saves clean Parquet files
- [ ] Commit the script to a GitHub repo with a short README

Week 3 (days 15–21): Model the data

Goal: design tables that make reporting easy and correct. Good modeling is what turns raw data into numbers people trust.

Learn

- [ ] ETL vs ELT, and why most modern teams load first and transform in the warehouse
- [ ] Data warehouse, data lake and lakehouse: what each one is and when teams use it
- [ ] Fact tables and dimension tables, and the star schema
- [ ] Grain: deciding what one row in a table represents

- [ ] Primary keys and foreign keys, and how they prevent duplicates

Build

- [ ] Sketch a star schema for your dataset: one fact table for the main events plus two or three dimension tables
- [ ] Write the SQL that builds those tables from your clean Week 2 data

Week 4 (days 22–30): Ship a pipeline

Goal: connect everything into one pipeline that runs on its own and checks its own work.

Learn

- [ ] What a data pipeline is: extract, load, transform, then serve
- [ ] Idempotency: why running a pipeline twice must not double your data
- [ ] Data quality checks: not-null, unique, accepted values and row-count checks
- [ ] Orchestration basics: scheduling, dependencies and what a DAG is
- [ ] Batch vs streaming, and why most pipelines start as batch

Build

- [ ] Combine your Week 2 script and Week 3 SQL into one pipeline you can run with a single command
- [ ] Add at least three data quality checks that fail loudly when something is wrong
- [ ] Run it twice and confirm your row counts don't change
- [ ] Update your GitHub README: what the pipeline does, how to run it and a diagram of the flow

After day 30

You've shipped your first pipeline. Here's where to go next:

- **Keep watching.** Each new She Ships Data episode adds a skill, and its Ripple exercise gives you something new to practice on. Subscribe on YouTube.
- **Follow the full roadmap.** Concepts on Databricks, AWS, Azure and Google Cloud, code practice and end-to-end projects, all step by step. Start free at prep.sheshipsdata.com.
- **Keep the glossary handy.** When a term in an episode or a job post trips you up, look it up in your **Data Engineering Glossary**.

Questions? Email support@sheshipsdata.com.