Database

Environment Variable

Database

Redis

-JS Object with key, value pairs

-in memory (RAM not disk?), useful for caching

Document (Mongo, Firestorm, DynamoDB)

Collection

-group of Documents

Documents

-no schema

-has id

-container for key, value pairs inside

Relational (MySql, postgres)

-SQL: Structured Query Language

-Foreign keys: because they reference data in different table

-requires a schema

-ACID (Atomicity: all or nothing, Consistency: data is valid, Isolation: multiple transaction at the same time, Durability: committed data is never lost)

Environment Variable

-UPPER\_SNAKE\_CASE

-Provide different values for different environments

-Store secrets, such as database URLs and credentials or API keys

- NODE\_ENV (common env variable)

Ex

-In a prod env, may want additional logging and analytics compared to dev env

-Building an API connected to a db but want to use a separate test db. Can pass a test db’url and cred when develop locally

Multiple Ways to load env variable

NODE\_ENV=prod VIDEO\_URL="https://www.youtube.com/watch?v=X2CYWg9-2N0" node index.js

-Gets cumbersome quickly,

export NODE\_ENV=prod VIDEO\_URL="https://www.youtube.com/watch?v=X2CYWg9-2N0"

-shell command which saves env var to current shell session (new shell would not have access)

printenv

-view all env var but shows too much

.env

-can’t push sensitive data in package.json and add to .gitignore to hide sensitive info

-import .env, team members should have synced access to same env var