



# Prisma Quick Start



## What is Prisma?

- ORM used by Keystone to interact with Postgres
- CLI accessible by using `yarn keystone prisma <command>`
- Or run with `npx prisma <command>`
- Lots of docs: <https://www.prisma.io/docs/>



# Development

- `prisma migrate dev`
  - Run when pulling down main, starting a new branch, etc.
  - Makes sure your local dev db is set up correctly
- `prisma db push`
  - Use when prototyping/coding new features with schema changes
  - Updates your local db without creating a migration
  - Once code is stable, create and apply a migration:  
`prisma migrate dev --name <name of migration>`
- `prisma migrate dev --create-only`
  - Create a migration *without* applying it
  - Allows editing before migration
  - More on customizing migrations [here](#)



## Development: Seeding the DB

In package.json: {..."prisma": { "seed": "npx ts-node seed.ts"}...}

- Runs `seed.ts` automatically with `prisma migrate dev` or `prisma migrate reset`
- Manually trigger by running `prisma db seed`
- **Recommended for dev and test environments**



# Using interactive commands in Docker

```
docker run --interactive --tty <image name>
```

```
# or
```

```
docker -it <image name>
```

```
# Example usage
```

```
docker run -it node
```

Use interactive commands like `prisma migrate dev` in Docker

Docs: <https://www.prisma.io/docs/concepts/components/prisma-migrate#troubleshooting>



# Production

```
prisma migrate deploy
```

- Use in automated CI/CD pipeline
- Compares applied migrations against migration history
- **Warns** if any migrations have been modified
- Applies pending migrations

**Does not:** detect drift, reset the db, generate artifacts



# Baselining in Prod

- Initializes migration history for db that:
  - Existed before using prisma migrate
  - **Contains data that must be maintained**
- **Q: Do we have to worry about this since our instance is brand-new?**

Docs: <https://www.prisma.io/docs/guides/database/developing-with-prisma-migrate/baselining>