1️⃣ Create a Migration

For ApplicationDbContext:

```
# Create a new migration in a custom folder
dotnet ef migrations add InitialApplicationMigration `
    --context ApplicationDbContext `
    --output-dir Configurations/Migrations/Application
```

For IdentityDbContext:

```
# Create a new migration in a custom folder
dotnet ef migrations add InitialIdentityMigration `
    --context IdentityDbContext `
    --output-dir Identity/Migrations
```

Notes:

- `--context` specifies which DbContext the migration belongs to.

- `--output-dir` specifies the folder where migration files are stored.

2️⃣ Apply (Update) Database

For ApplicationDbContext:

```
dotnet ef database update `
    --context ApplicationDbContext
```

For IdentityDbContext:

```
dotnet ef database update `
    --context IdentityDbContext
```

Each command will **only apply tables for the specified context**. Safe to run on the same database.

---

3 Rollback to a Specific Migration

If you need to roll back:

For ApplicationDbContext:

```
dotnet ef database update InitialApplicationMigration `
    --context ApplicationDbContext
```

For IdentityDbContext:

```
dotnet ef database update InitialIdentityMigration `
    --context IdentityDbContext
```

> Use the **migration name you want to roll back to**.
> Use 0 to rollback completely (drop all tables for that context).

---

4 Remove the Last Migration

For ApplicationDbContext:

```
dotnet ef migrations remove `
    --context ApplicationDbContext
```

For IdentityDbContext:

```
dotnet ef migrations remove `
    --context IdentityDbContext
```

> Removes the **last migration** (doesn't affect database). Useful if you mess up
> the migration and want to recreate it.

---

5 Drop Database

For ApplicationDbContext:

```
dotnet ef database drop `
    --context ApplicationDbContext `
```

```
    --force
```

For IdentityDbContext:

```
dotnet ef database drop `
    --context IdentityDbContext `
    --force
```

`--force` skips the prompt.
Make sure your connection string points to a database you **can delete**.

6️⃣ Recreate Migrations From Scratch

If you want to **start completely clean**:

```
# Remove migration folder contents manually first
# Then recreate initial migration
dotnet ef migrations add InitialApplicationMigration `
    --context ApplicationDbContext `
    --output-dir Configurations/Migrations/Application

dotnet ef migrations add InitialIdentityMigration `
    --context IdentityDbContext `
    --output-dir Identity/Migrations

# Apply to database
dotnet ef database update --context ApplicationDbContext
dotnet ef database update --context IdentityDbContext
```

7️⃣ List Migrations

```
dotnet ef migrations list --context ApplicationDbContext
dotnet ef migrations list --context IdentityDbContext
```

Shows which migrations exist for each DbContext.