

⇒ A FastAPI can easily be deployed at the platform railway.app

Connecting to, and using CLI
for project config & deploy:-

① First, deploy the project to Railway, by using Postgres Provision and from github repository where project is deployed.
Follow the article (saved as PDF in Deploy folder), in my computer.

② Set the environment variables for the project, as an environment file (just as we used .env file) by going to our deployed project in railway, then :-

Variables tab → {{ RAW Editor.

make the same variables here, as we made in our local .env file.

The details of database will remain here in the Postgres named section.

③ Now, the project is successfully built and started. We can see it by Generating domain in the Settings section, and going to that URL.

④ Now, project is successfully up.

But, as we are using Alembic to handle our database migrations, our tables are not automatically created.

We need to create it, using Alembic commands from terminal.

⑤ To use alembic commands, we need to link our local system Windows cmd, to our Railway project, such that we can execute commands in our local machine cmd, and it will also get executed in the cloud.

NOTE THAT:

To execute some commands using some tools and frameworks, we need correct path and framework & virtual environment setting in our local system's cmd/terminal.

On successful execution of command in our local machine, the result/output of the command will be reflected in the railway cloud environment using railway up command.

⑥ Open cmd in our local machine/system:

Install npm, if not present.

↳ Install railway cli using following command:

[npm i -g @railway/cli

⑦ In our local cmd, set our local project directory, as current working directory.

Ex:- [E:\Programs & Codes\FastAPI >

⑧ After setting the current working dir as project directory, we need to link it with our railway app cloud project, using command :-
→ project id in cloud.

[railway link 0b701af81-...-fb257]

This command can be obtained by going to our account dashboard, and clicking on the [Set up your project locally] option at the bottom-left corner.

⑨ Check status, as to which project & environment we are currently linked to :-

[railway status]

⑩ Then, we will execute all the required commands locally, and will pass the output to the railway cloud deployment as follows :-

[railway run <cmd to be run>
[railway up]

Ex:- Suppose, we activate the venv in the cloud as follows:-

[railway run .\venv\Scripts\activate] ①

[railway up] ②

These commands will actually be run locally (in our system) first, and if successful, will then be passed to cloud deployment, using ② command.

⑪ First move the project folder's alembic folder & alembic.ini file to a separate folder for temporary purpose.

Then run the command :-

[railway run alembic init alembic]
[railway up]

⑫ In the project folder, bring the original alembic folder & alembic.ini file, that we just moved & stored at other place.

Then, run following command :-

[railway run alembic upgrade head]
[railway up.]

This will create all the database tables using alembic, in the cloud. Project is now ready!