# Lesson 01: FastAPI Overview

## Welcome to the Course 🎉

Welcome to this **FastAPI Full Course**.
In this course, you will learn how to build **modern, fast, and production-ready APIs** using **Python and FastAPI**.

This first lesson gives you a **big picture overview** of FastAPI before we start writing real code.

## What You Will Learn in This Lesson

By the end of this lesson, you will understand:

- What FastAPI is
- Why FastAPI is popular
- What FastAPI is used for
- How FastAPI compares to other frameworks
- What we will build in this course

## What is FastAPI?

**FastAPI** is a modern Python web framework used to build **APIs**.

It is designed to be:

- 🚀 Very fast
- 🧠 Easy to learn
- 🔒 Safe with built-in validation
- 📄 Well-documented automatically

FastAPI is commonly used to build:

- REST APIs

- Backend services

- Microservices

- Backend for web & mobile apps

---

# Why is FastAPI So Popular?

FastAPI became popular because it solves many problems developers had with older frameworks.

## Key advantages:

- **High performance**
  Built on Starlette and Pydantic.
  One of the fastest Python frameworks.

- **Easy to write**
  Clean and readable syntax.
  Less boilerplate code.

- **Automatic API documentation**
  Swagger UI and ReDoc are generated automatically.
  No extra setup required.

- **Strong data validation**
  Uses Python type hints.
  Fewer runtime errors.

---

# FastAPI vs Other Python Frameworks

| Framework | Best For |
| --- | --- |
| Flask | Small apps, learning basics |
| Django | Full-stack web applications |
| FastAPI | APIs, microservices, modern backends |

FastAPI is **not a replacement** for everything, but it is an **excellent choice for APIs**.

# How FastAPI Works (High Level)

FastAPI works by:

1. Receiving an HTTP request (GET, POST, PUT, DELETE)
2. Validating the data automatically
3. Executing your Python function
4. Returning a JSON response

You write **normal Python functions**, and FastAPI handles the rest.

# What We Will Build in This Course

Throughout this course, you will learn by building real examples, including:

- Simple APIs
- CRUD APIs
- APIs with validation
- APIs with databases
- Authentication using JWT
- A complete **To-Do List API** (final project)

By the end of this course, you will be able to:

> Build and design real-world FastAPI backends confidently 💪

## Do Not Worry About Setup Yet

In this lesson:

- ❌ No installation
- ❌ No coding
- ❌ No configuration

We focus only on **understanding concepts**.

In the **next lesson**, we will:

- Install FastAPI
- Create our first FastAPI app
- Run a server locally

## Next Lesson 🚀

➡️ **Lesson 02: Installing FastAPI & First API**

## 💡 Tip for Students

If something feels unclear right now, that's normal.
Everything will become clear once we start coding step by step.