Deployment Plan - Online Appointment Management System

1. Introduction

1.1 Purpose

This document outlines the deployment strategy for the Bus Rental and Revenue Management System, ensuring a smooth transition from development to production while maintaining reliability, security, and scalability.

1.2 Scope

The deployment plan covers the release process, environments, CI/CD pipelines, rollback strategy, monitoring, and security measures for the Bus Rental and Revenue Management System.

1.3 Target Audience

- Developers
- Software developers
- System architects
- Database administrator
- Test engineers

2. Deployment Architecture

2.1 Infrastructure Overview

- **Cloud Provider**: AWS / Google Cloud / Azure
- **Compute Services**: EC2 / Kubernetes / Docker Containers
- **Database**: MySQL
- **Storage**: S3 for static assets, EFS for persistent data
- **Caching**: Redis / Memcached
- **Load Balancing**: Elastic Load Balancer (ELB) / Nginx / Cloudflare
- **Monitoring & Logging**: Prometheus, Grafana, ELK Stack (Elasticsearch, Logstash, Kibana)

2.2 Deployment Environments

Environment	Purpose	Hosting Platform
Development	Ongoing development/testing	Local / Cloud Dev Instance
Staging	Pre-production testing	Cloud Staging Instance
Production	Live system for end users	Cloud Production Instance

3. Deployment Process

3.1 Continuous Integration & Deployment (CI/CD)

Tools Used: GitHub Actions / Jenkins / GitLab CI

Steps:

1. Code is pushed to repository (GitHub / GitLab / Bitbucket).

2. Automated tests are executed.
3. Code is built into Docker images.
4. Artifacts are stored in container registry.
5. Deployment to staging environment for final verification.
6. Manual or automated approval for production release.
7. Deployment to production using blue-green or rolling update strategy.

3.2 Deployment Strategies

Strategy	Description
Blue-Green	Two identical environments; traffic is switched to the new version after verification.
Rolling Update	Gradually deploys new versions, ensuring zero downtime.
Canary Release	Deploys to a small percentage of users first before full rollout.

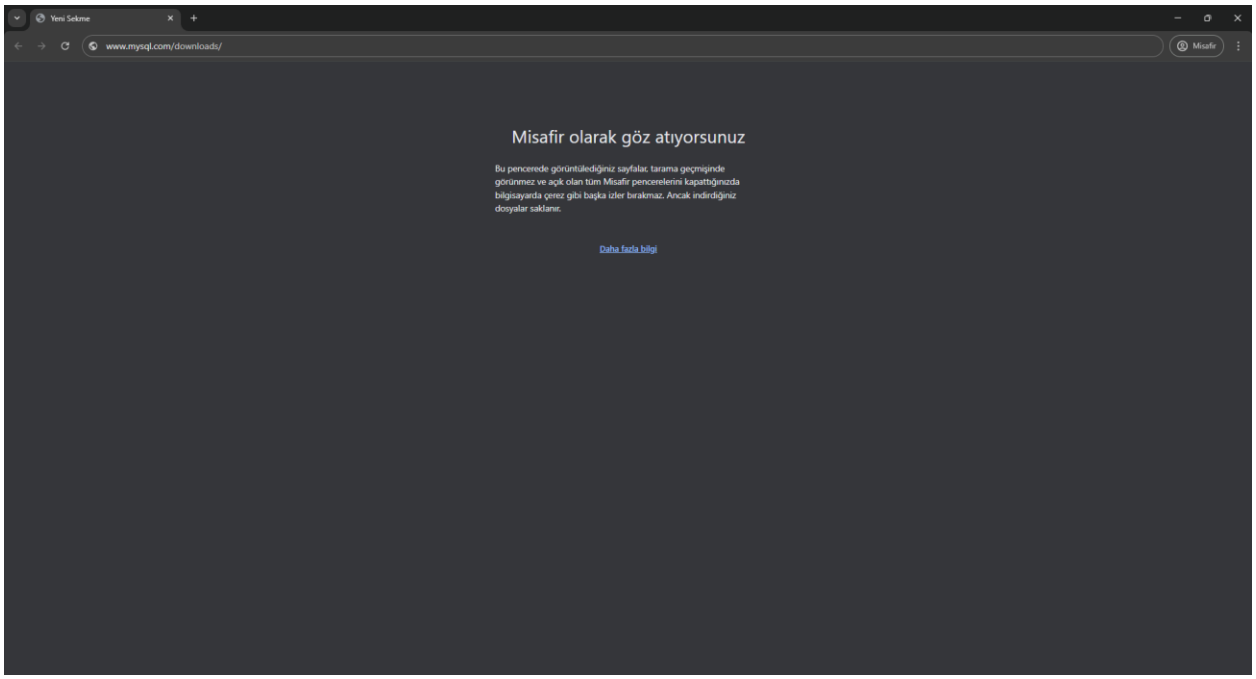
3.3 Rollback Strategy

- **Database Backups**: Automatic backups before deployment.
- **Feature Flags**: Toggle features off if a failure occurs.
- **Versioned Deployments**: Ability to revert to the last stable version.
- **Monitoring Alerts**: Immediate alerts in case of failure.

SETUP

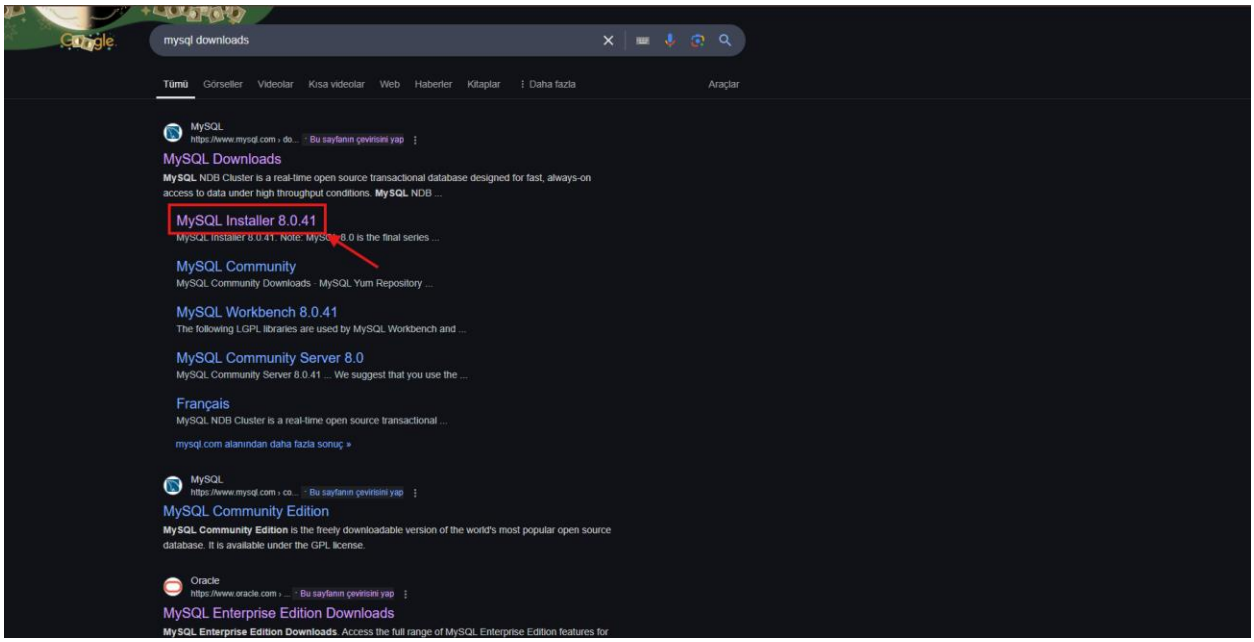
Step - 1

Search by typing 'www.mysql.com/downloads/' in the search bar.



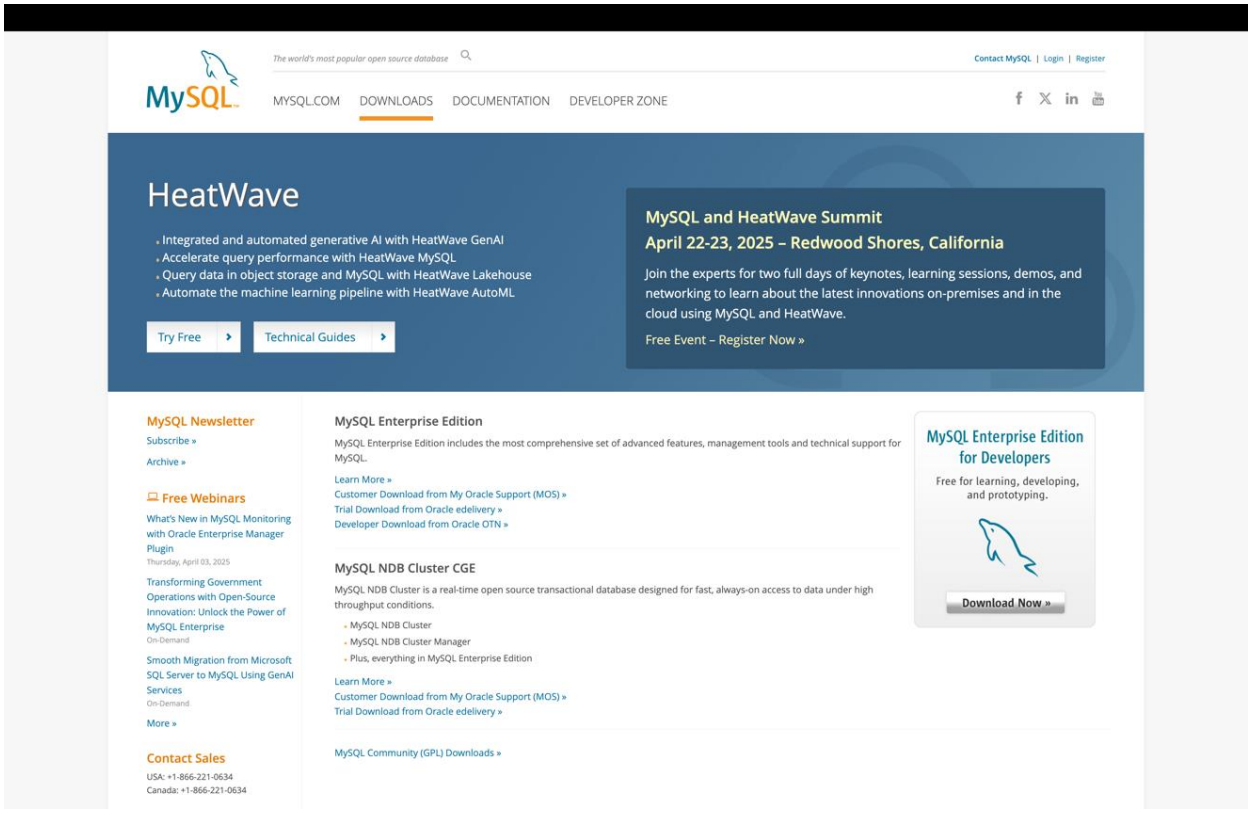
Step – 2

On the page, click on “”MySQL Installer 8.0.41”.



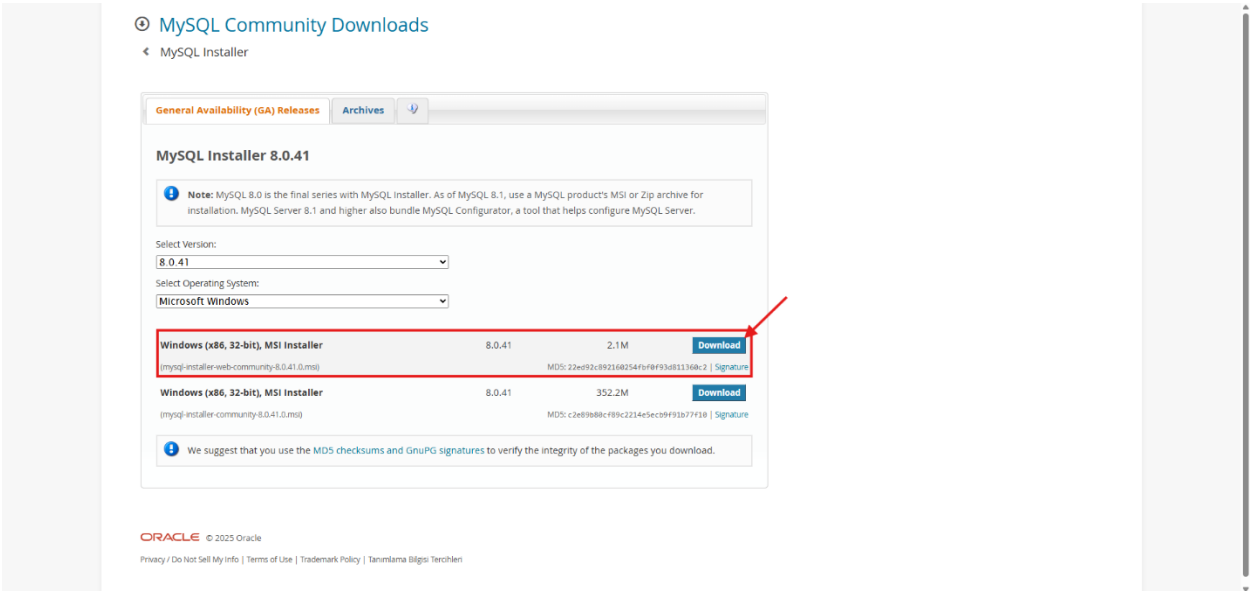
Step – 3

On the page, click on “MySQL Community (GPL) Downloads.”.



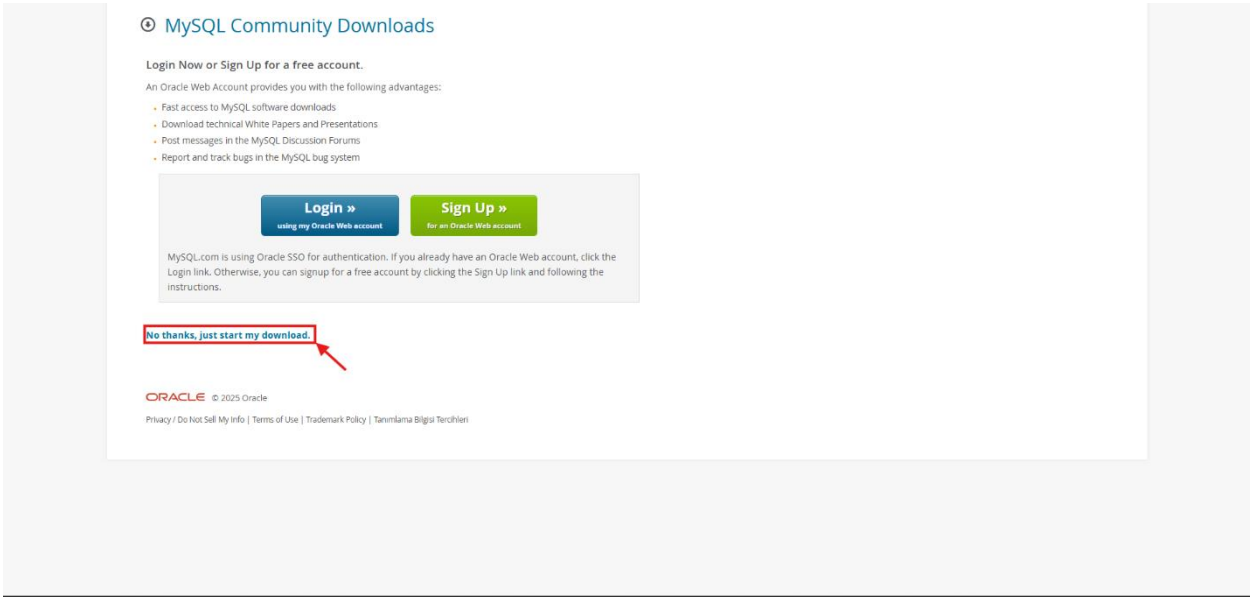
Step – 4

On the page, click on the "Windows(x86, 32-bit), MSI Installer Download.".



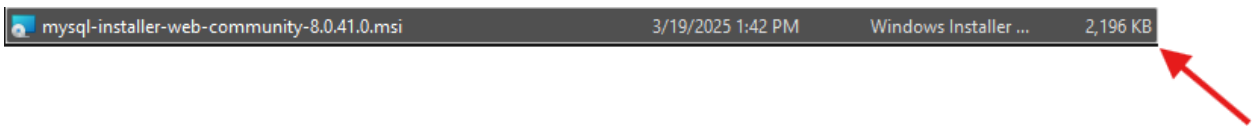
Step – 5

On the page, click on "No thanks, just start my download.".



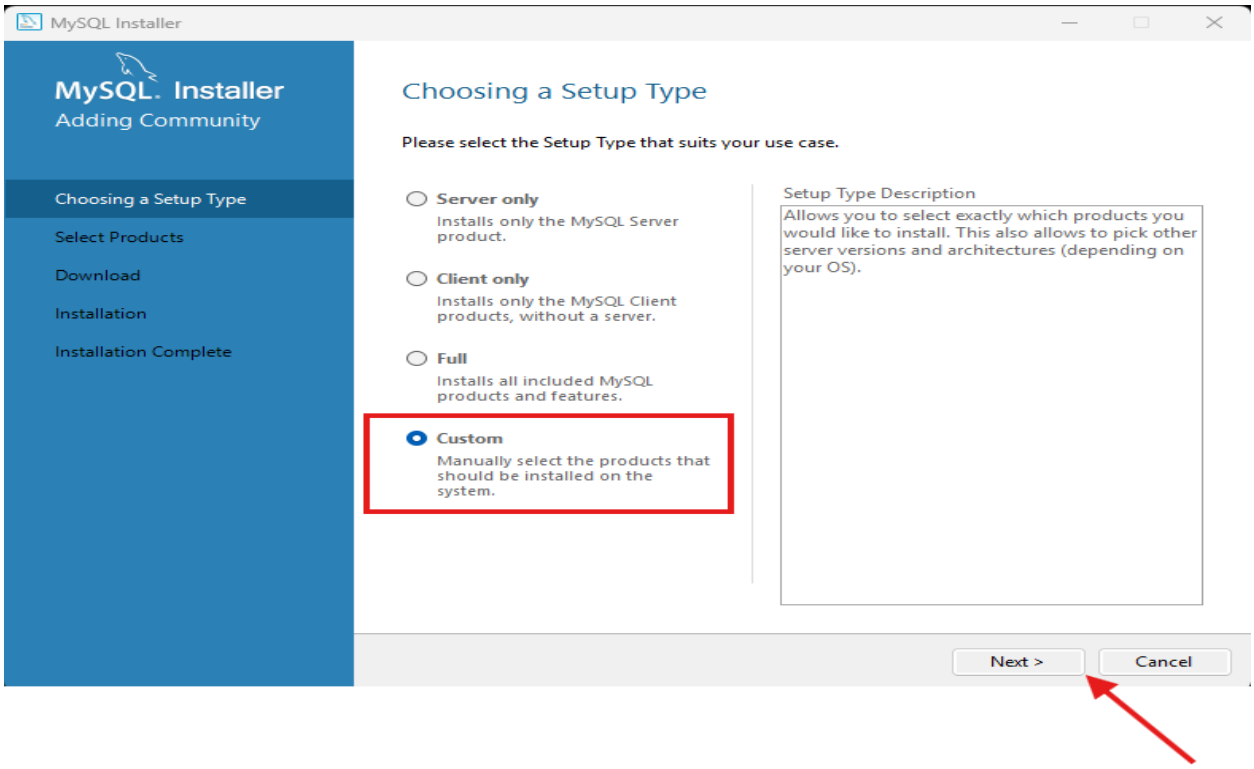
Step – 6

In downloads, we open the incoming file by double-clicking on the icon.



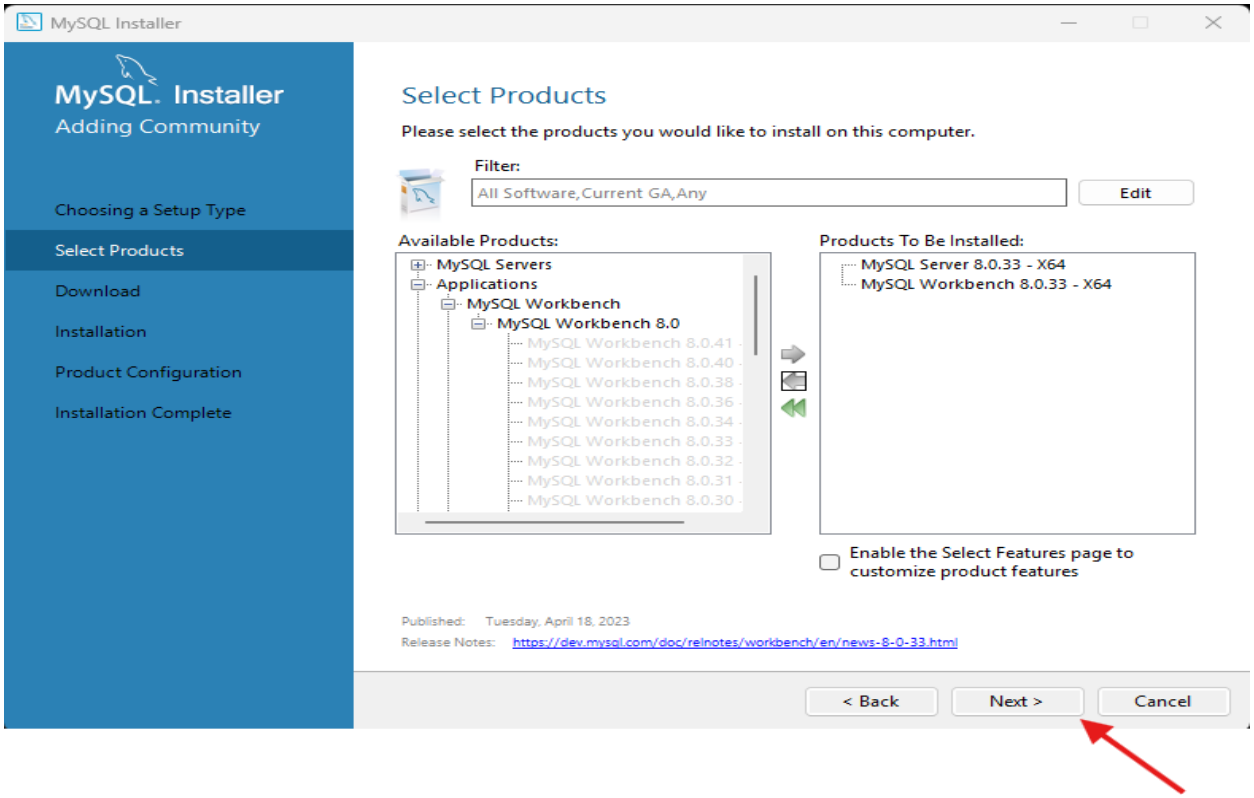
Step – 7

We will choose “Custom” part on the setup type and press Next button.



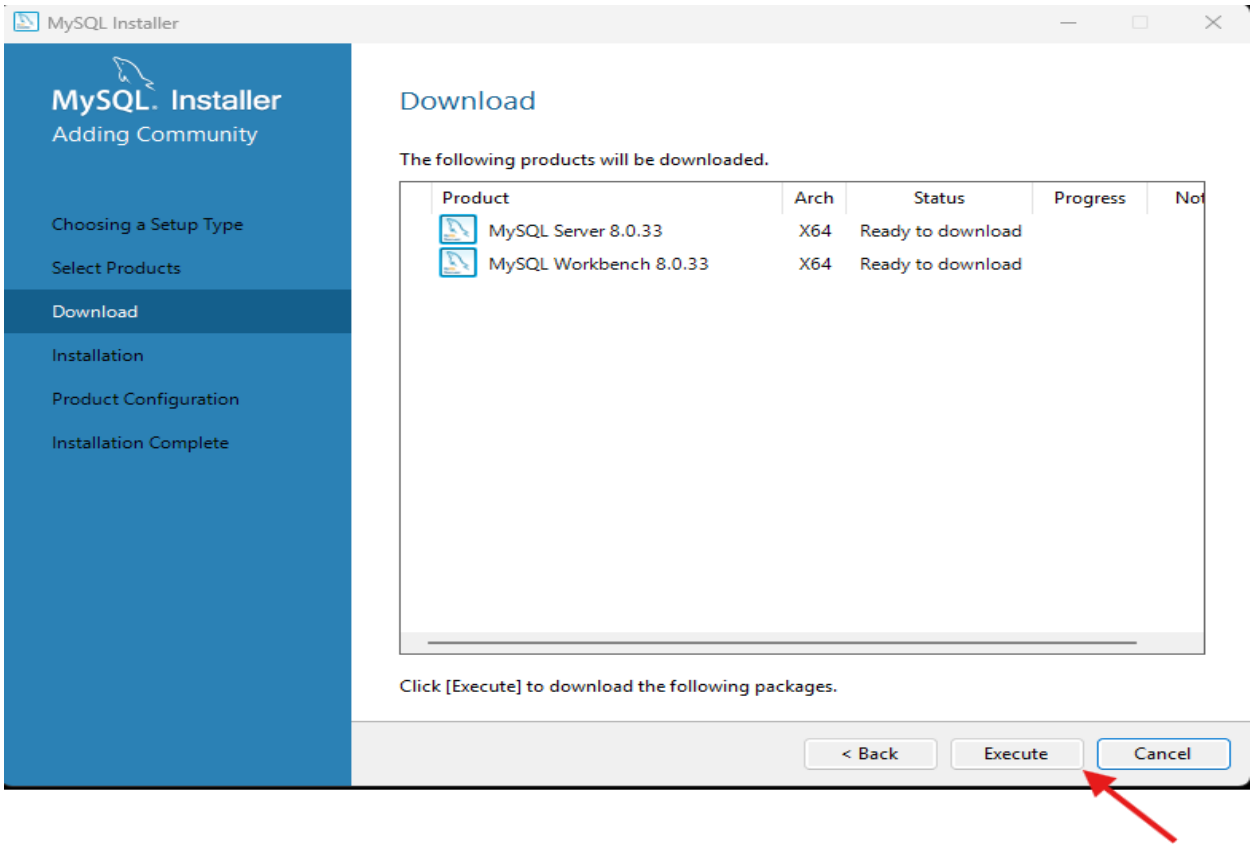
Step - 8

We have to choose application “MySQL Server 8.0.33 – X64 and MySQL Workbench .0.33 – X64 Packages.” and press Next button.



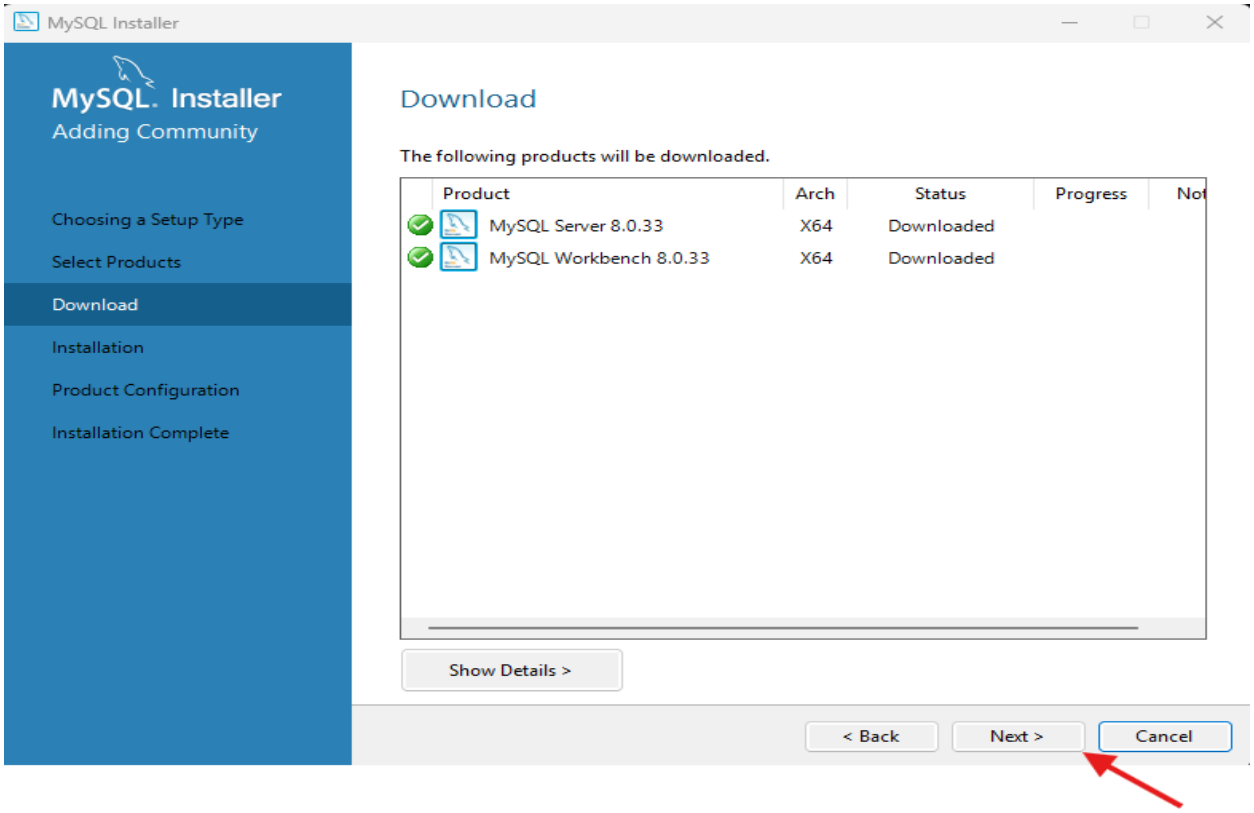
Step – 9

We will click [Execute] button to download the following packages.



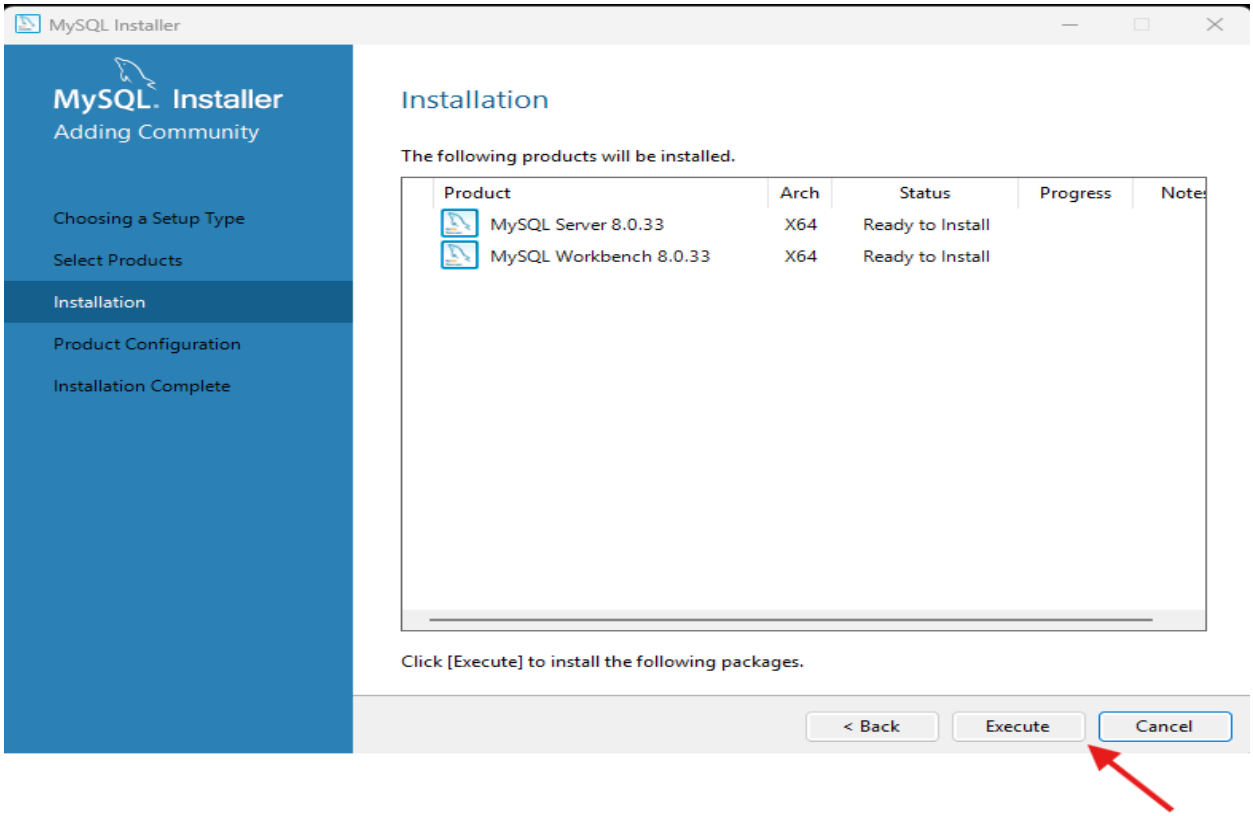
Step – 10

After choosen packages downloaded. Press the Next button.



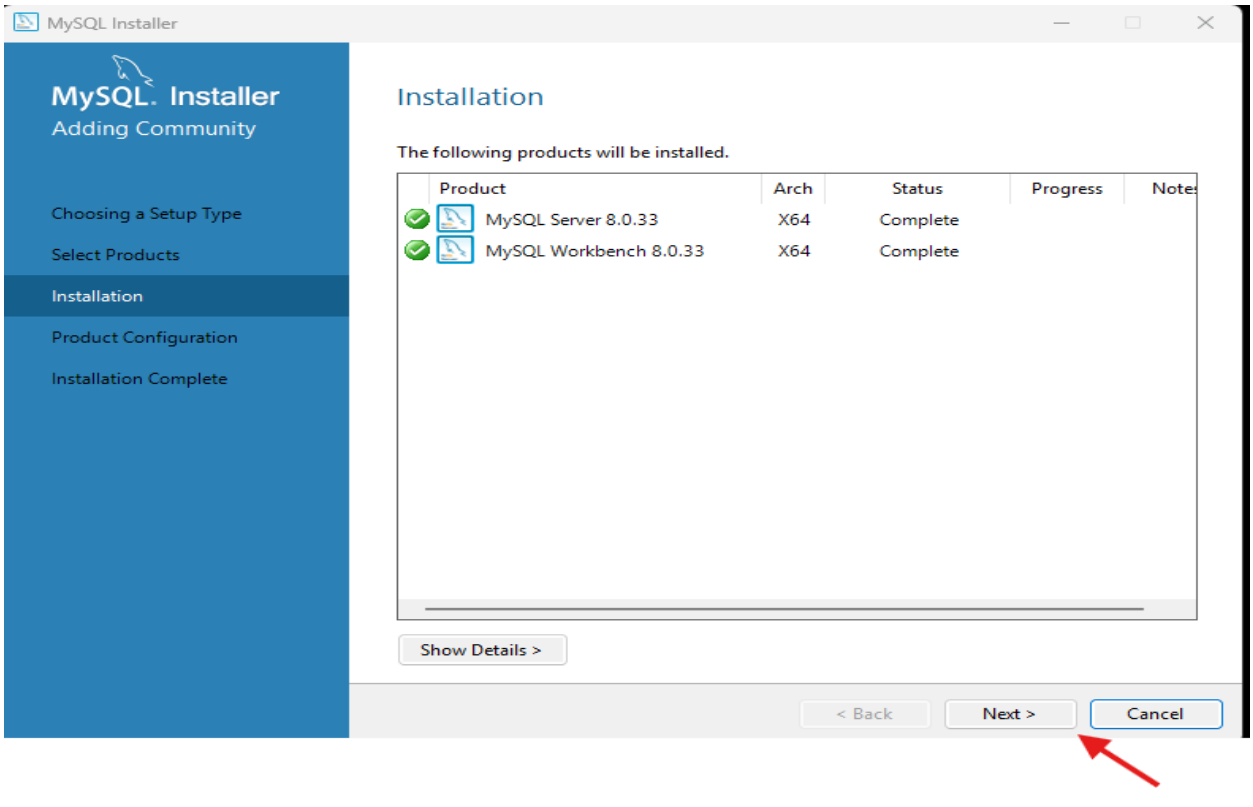
Step – 11

Click [Execute] button to install the following packages.



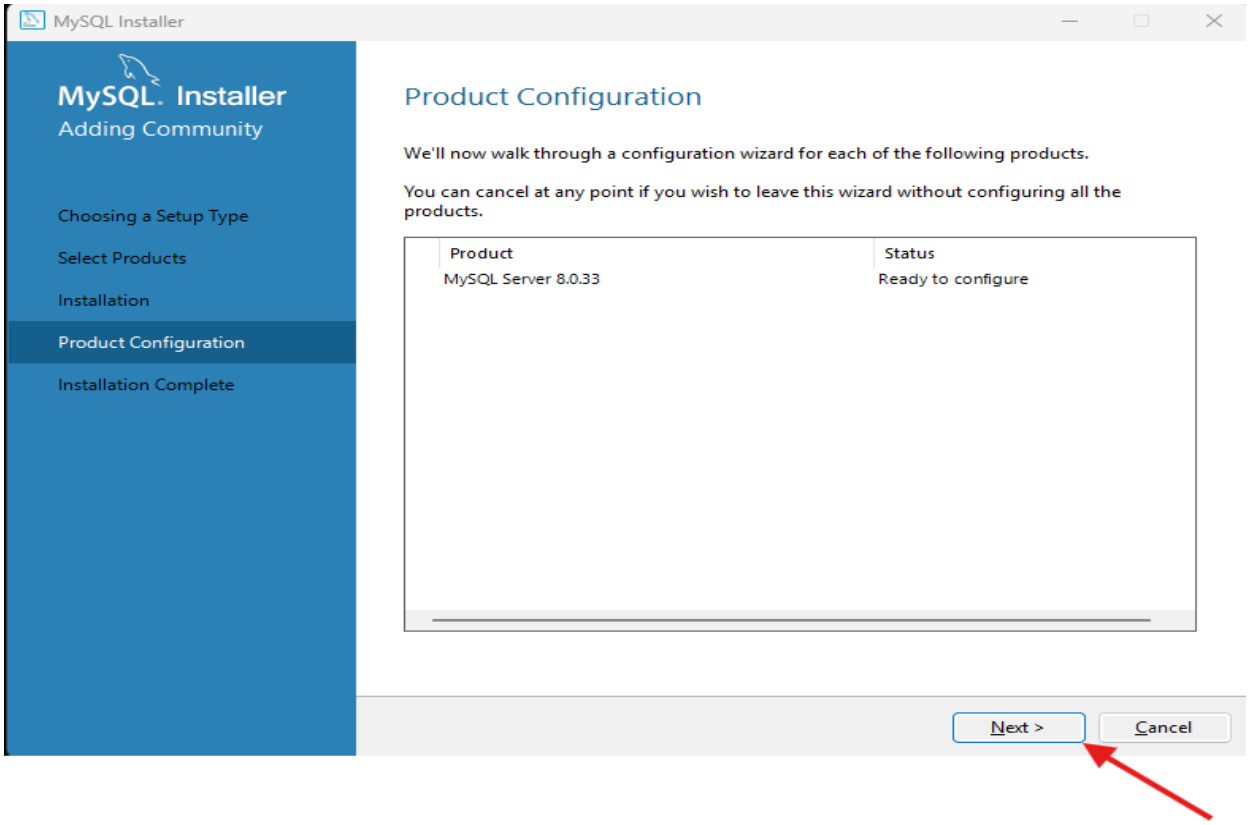
Step – 12

After choosen packages installed. Click Next button.



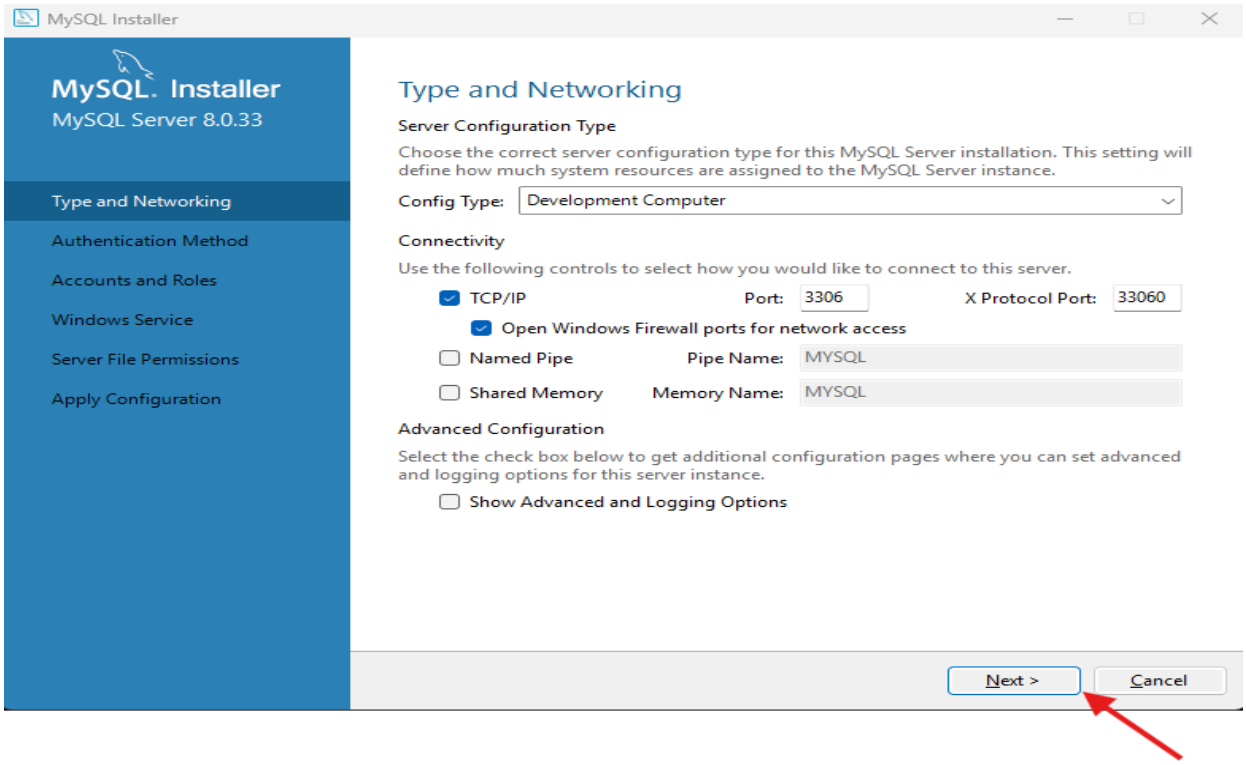
Step -13

MySQL Server 8.0.33 is ready to configure. Click the Next button.



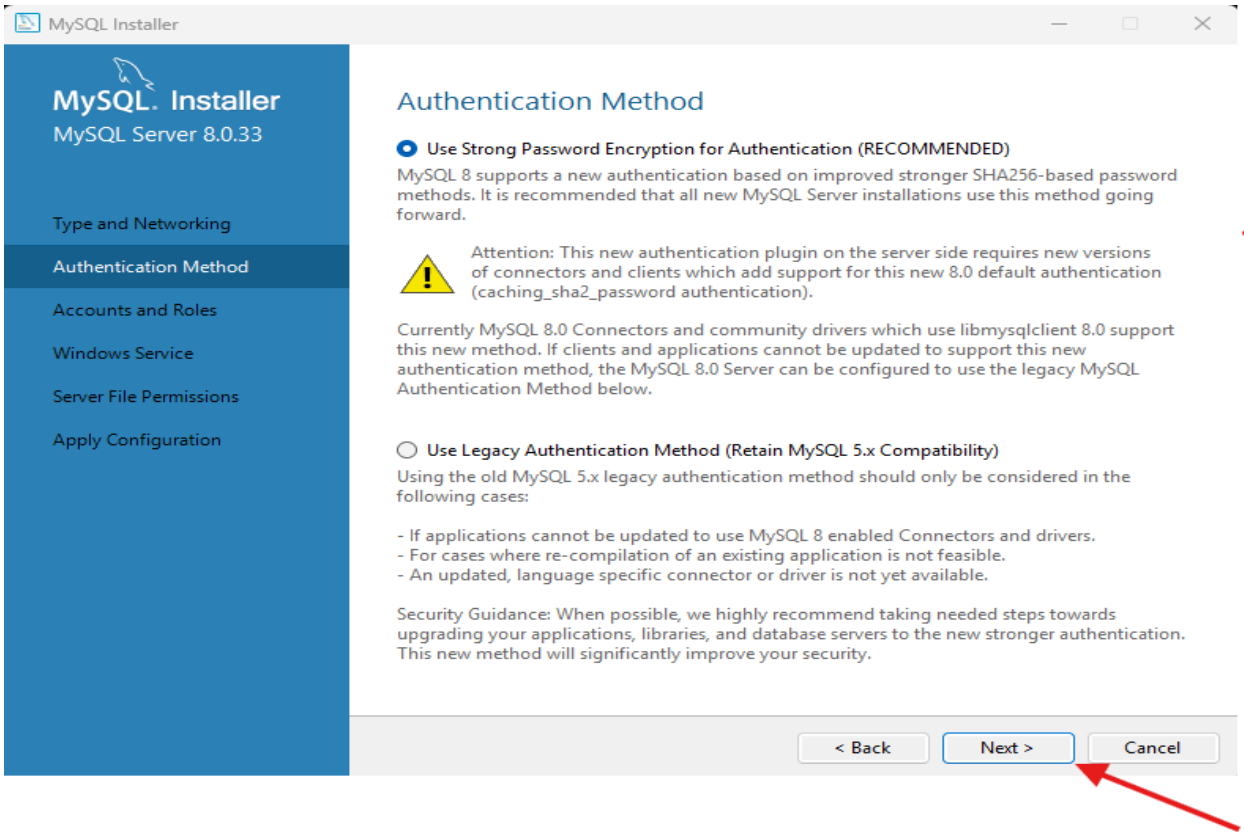
Step – 14

Do not change anything and press the Next button.



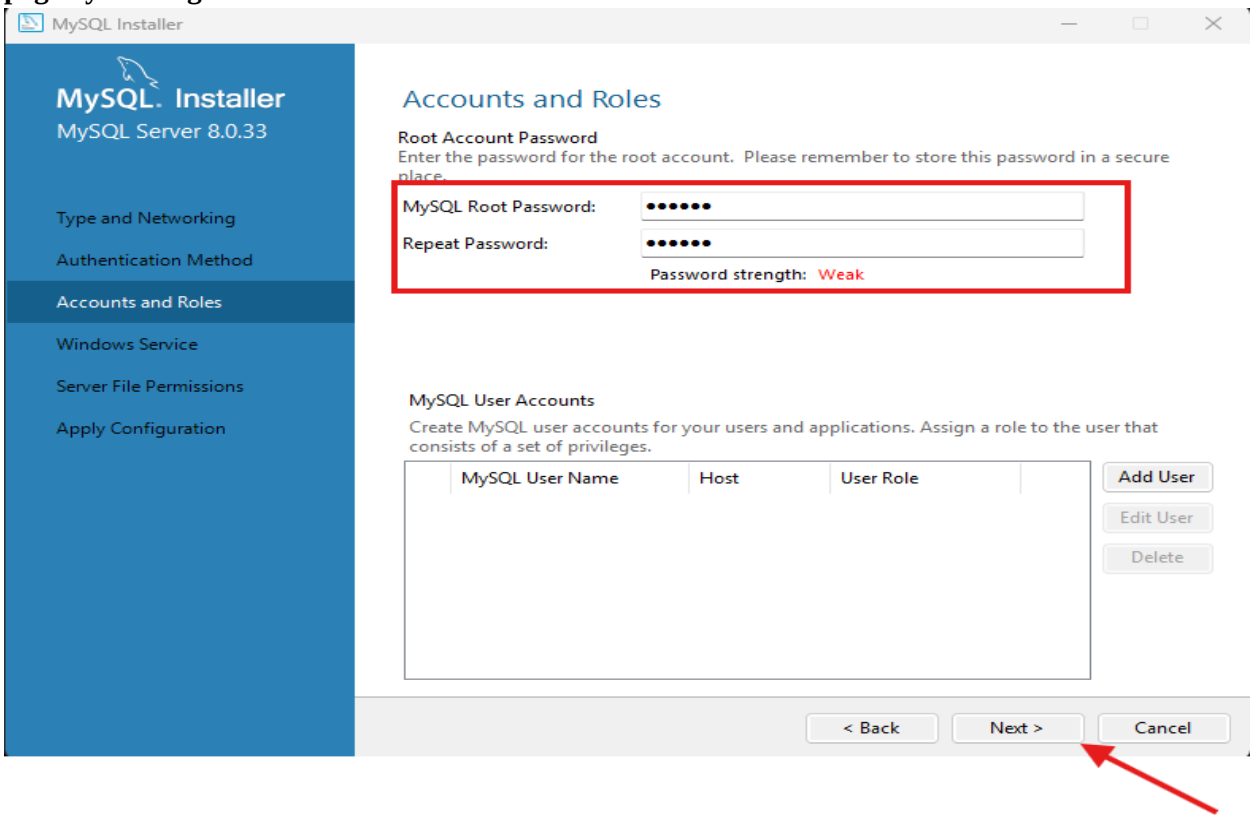
Step – 15

Choose the RECOMMENDED option and press the Next button.



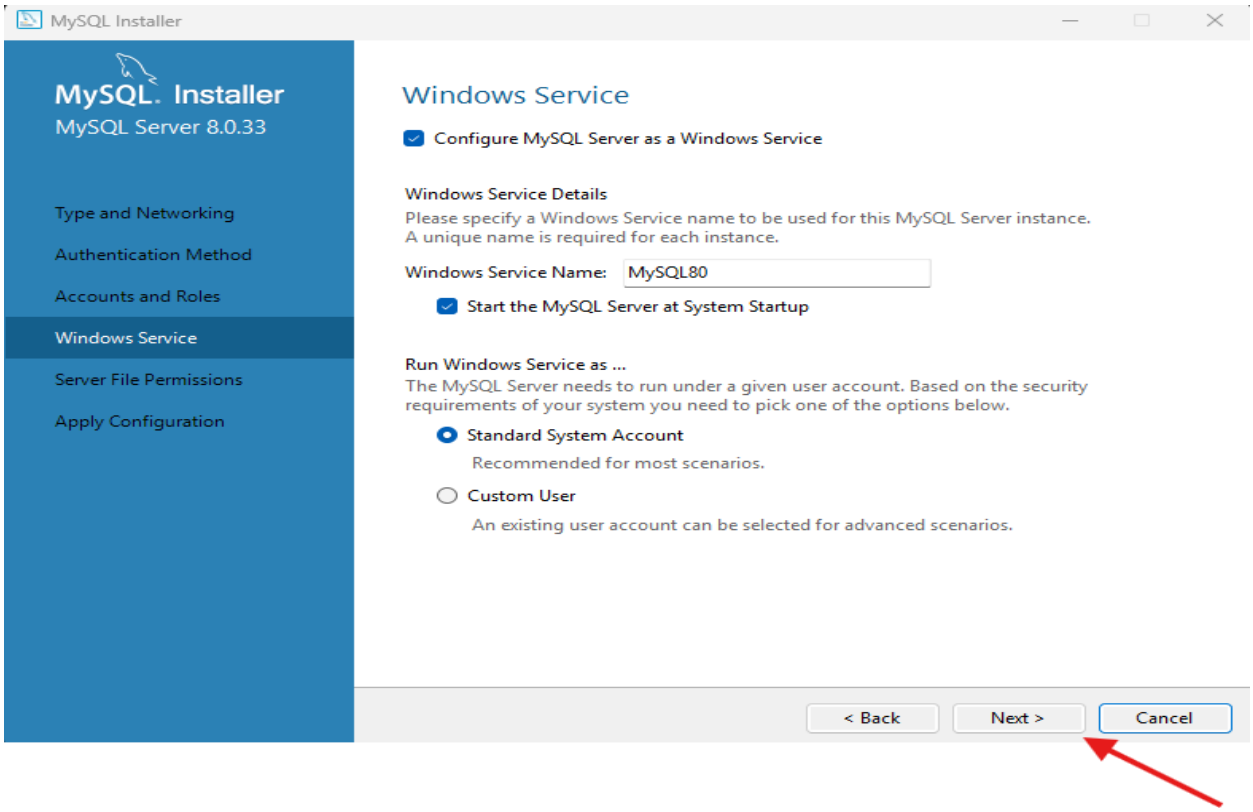
Step – 16

We specify a password for the "root" user in the configuration section. We go to the next page by clicking "finish" in this section.



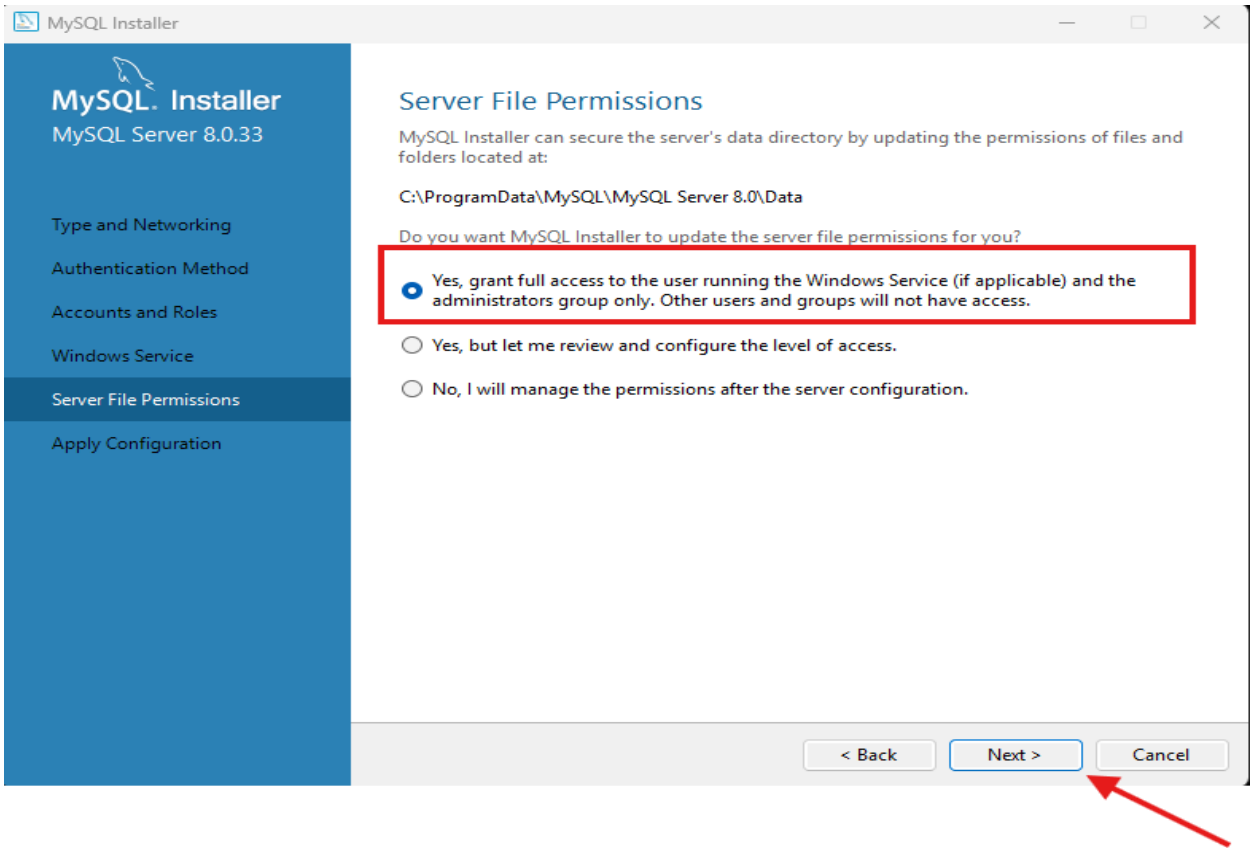
Step – 17

Do not change any options and click the Next button.



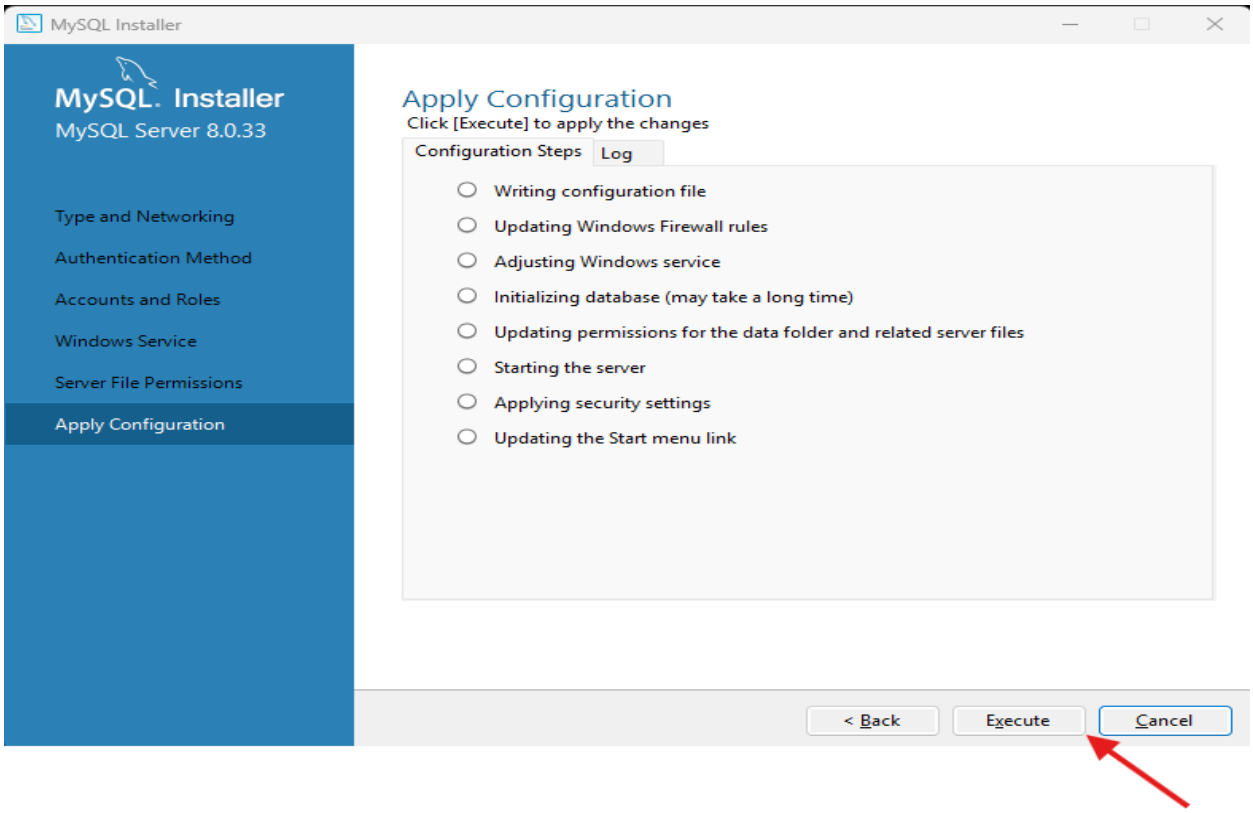
Step – 18

Choose the option in the red box and press Next button.



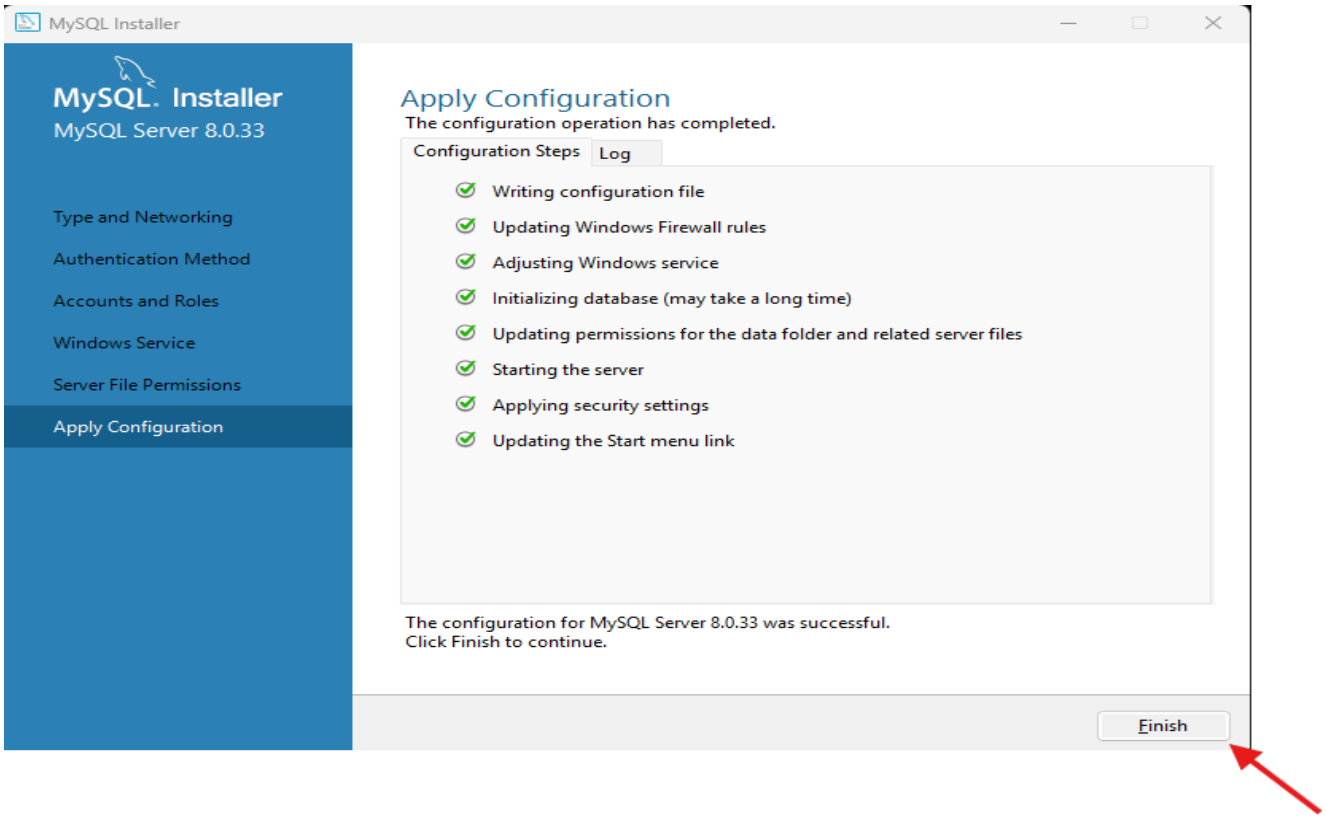
Step – 19

Click the [Execute] button to apply changes.



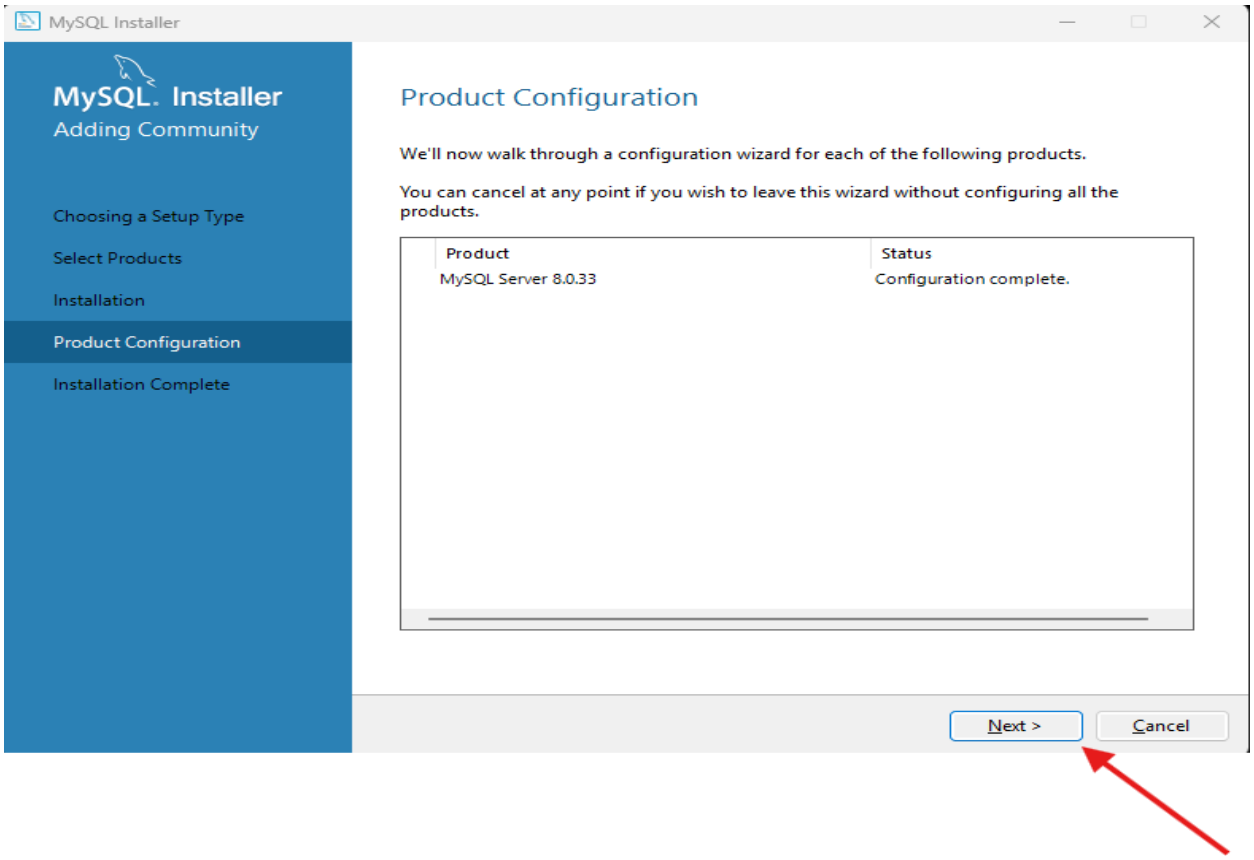
Step – 20

After the configuration has completed press the Finish button.



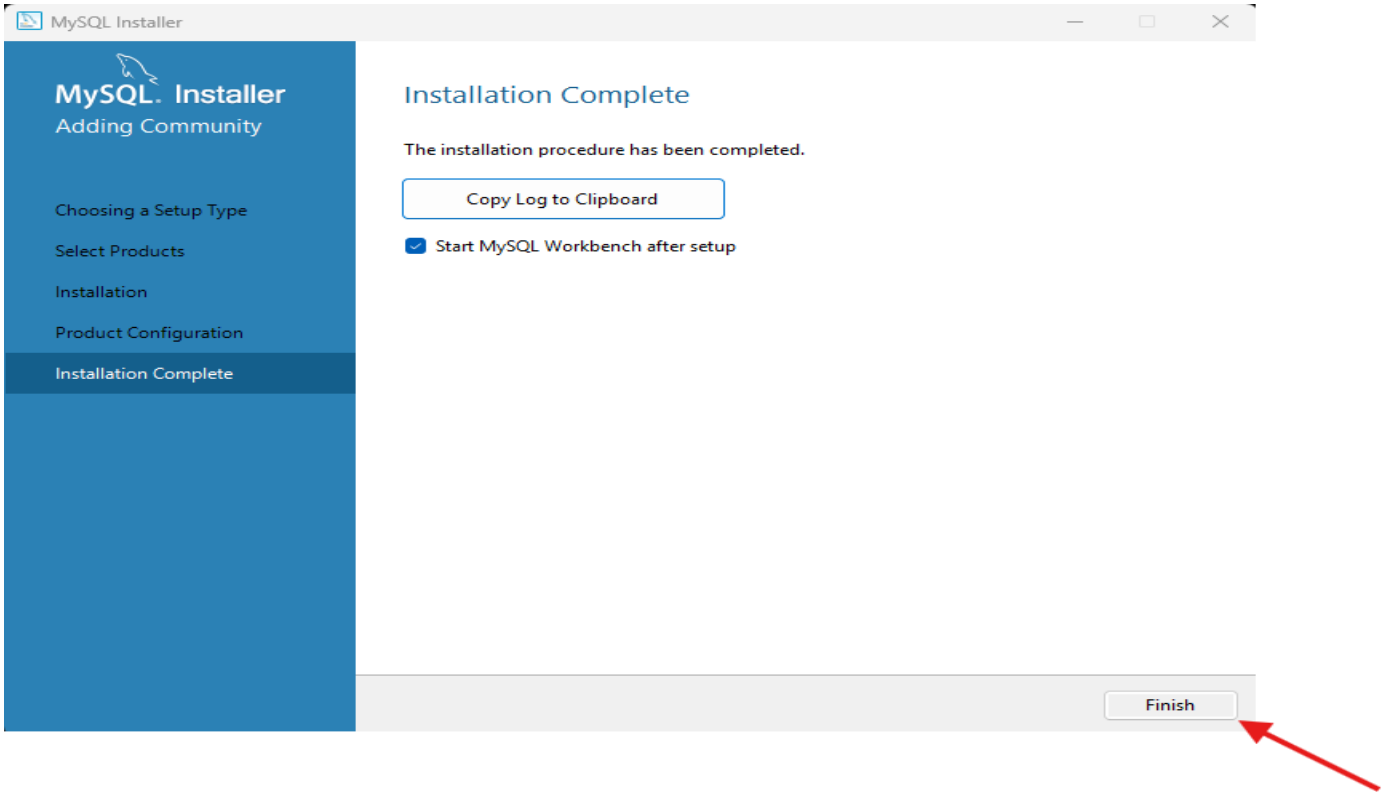
Step – 21

Configuration has completed. Click the Press button.



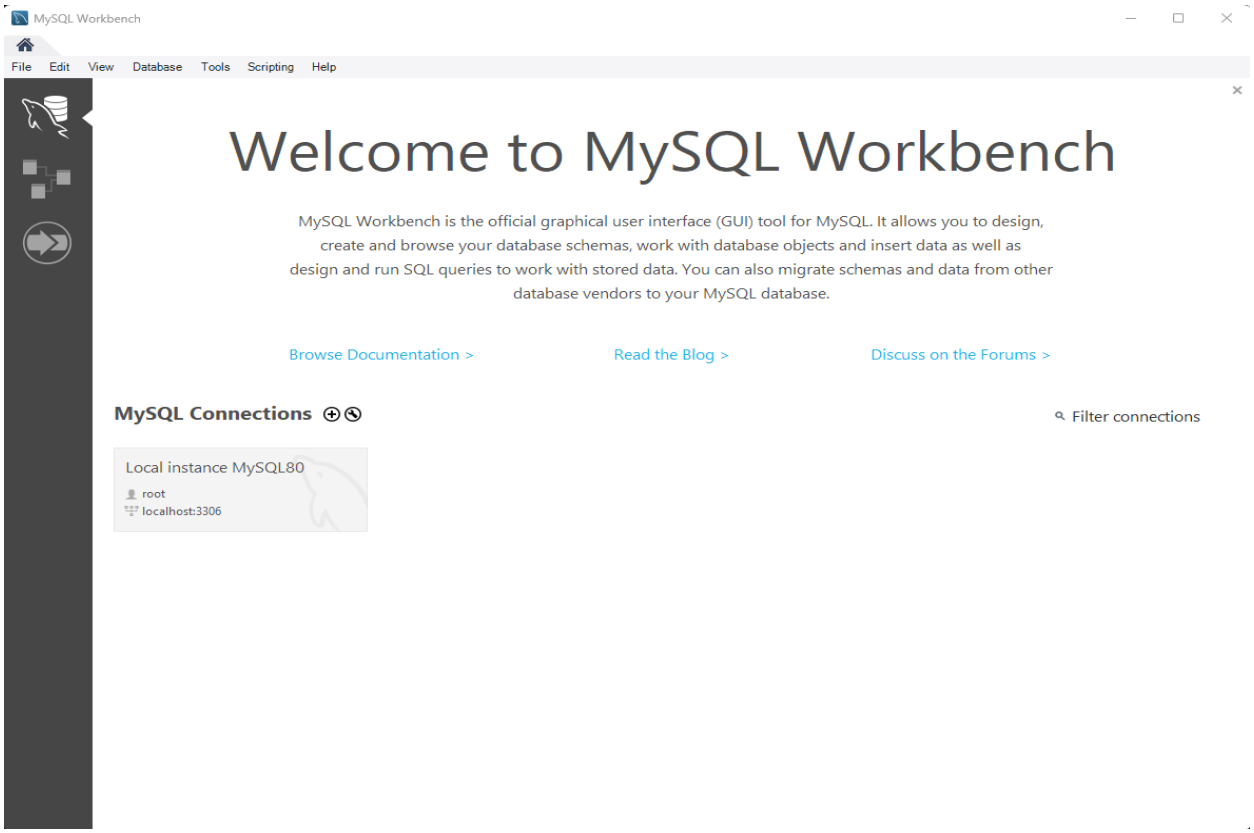
Step – 22

After that the Installation completed. Press the Finish button.



Step – 23

ALL DONE. You are welcome to MySQL Workbench.



4. Monitoring & Logging

The system features advanced monitoring and logging mechanisms to track bus status, rental processes, and financial transactions in real time. The admin dashboard provides detailed analytics on bus location, mileage, maintenance schedules, and rental operations. All rental agreements, payment transactions, bus movements, and system errors are securely recorded. Potential performance issues and security vulnerabilities are regularly analyzed. Additionally, automated notifications alert administrators about payment failures, upcoming maintenance, and anormal fuel consumption, ensuring timely interventions.

5. Security Considerations

5.1 Authentication & Authorization

- JWT-based authentication.
- Role-based access control (RBAC).

5.2 Data Protection

- Encrypted data at rest and in transit (AES-256, TLS 1.2+).
- Regular security audits and vulnerability scanning.

5.3 DDoS Protection

- Cloudflare / AWS Shield for mitigation.
- Rate limiting on API endpoints.

6. Backup & Disaster Recovery Plan

The system implements a robust Backup and Disaster Recovery Plan to ensure data security and business continuity.

As part of the Backup Strategy, the database is protected with daily full backups and hourly incremental backups. Static assets are securely stored using version-controlled backups.

Under the Disaster Recovery Plan, a multi-region deployment ensures high availability. The system is designed to restore services within less than 15 minutes (RTO) and minimize data loss with a recovery point objective (RPO) of under 5 minutes. In case of hardware failures, cyber threats, or unexpected outages, the system maintains operations with minimal disruption.

7. Conclusion

This deployment plan establishes a structured approach to ensure the secure, reliable, and scalable deployment of the bus rental and revenue management system. It guarantees minimal downtime, robust monitoring, and a well-defined rollback strategy to maintain system integrity and operational efficiency