🧠 Online Quiz Platform

An interactive and user-friendly web application where users can take quizzes, view results, and improve their skills. Admins can add questions, manage quizzes, and track performance. Built with modern web technologies to ensure responsiveness, security, and scalability.

🚀 Features

👤 For Students

- Register and log in securely
- Browse and take quizzes
- Instant scoring and detailed feedback
- View past quiz attempts and scores

🛠️ For Admins

- Dashboard to manage quizzes and questions
- Add, edit, and delete multiple-choice questions
- Categorize quizzes by topic and difficulty
- View student scores and analytics

🏗️ Tech Stack

- **Frontend:** HTML, CSS, JavaScript (or React if applicable)
- **Backend:** FastAPI / Django / Flask (replace based on your stack)
- **Database:** SQLite / MySQL / PostgreSQL
- **Authentication:** JWT / Session-based
- **Templating:** Jinja2 (if using Flask or FastAPI templates)
- **Hosting:** Render / Vercel / Heroku / Docker (optional)

📸 Screenshots

Add UI screenshots here (login page, quiz interface, admin dashboard, etc.)

⚙️ Installation & Setup

```
# Clone the repository
git clone https://github.com/yourusername/online-quiz-platform.git
cd online-quiz-platform

# Create and activate a virtual environment
python -m venv venv
source venv/bin/activate  # on Windows use `venv\Scripts\activate`
```

```
# Install dependencies
pip install -r requirements.txt

# Run the application
uvicorn app.main:app --reload  # for FastAPI
```

🧪 Usage

- Visit http://localhost:8000
- Register/Login as a student or admin
- Take quizzes or manage them via the dashboard
- View results and performance reports

🔐 Admin Access

Default admin credentials (change in production):

```
Username: admin
Password: admin123
```

📁 Project Structure (Example)

```
online-quiz-platform/
│
├── app/
│   ├── routers/          # Route definitions (student, admin)
│   ├── templates/        # HTML templates (Jinja2)
│   ├── static/           # Static files (CSS, JS)
│   ├── models.py         # Database models
│   ├── main.py           # FastAPI app entry point
│   └── ...
│
├── requirements.txt
└── README.md
```

🛡️ Security Features

- Input validation and sanitization
- Role-based access (admin vs student)
- Secure authentication
- CSRF protection (if applicable)

📈 Future Improvements

- Timer for quizzes
- Leaderboard system
- Export results as PDF/Excel
- Add support for images in questions
- Dark mode for better accessibility

🤝 Contributing

Pull requests are welcome! For major changes, please open an issue first to discuss what you would like to change.

📄 License

This project is licensed under the MIT License - see the LICENSE file for details.