1. HTTP State Preservation and Session Management[1][2][3]

HTTP is stateless, each request stands alone with no memory of previous requests. Web applications get around this limitation by using cookies and sessions to remember who you are across multiple page loads. When you log in, the server creates a session record on its side containing your user details and other information, then sends your browser a cookie containing a session ID (recording session ID as well as site identifiers, usually a domain and a Path). Your browser automatically includes this cookie with every subsequent request, allowing the server to look up your session and recognise you without asking for your password again. Django implements this through its session framework: when a request arrives, Django reads the session ID from the cookie, retrieves the matching session data from storage (either a database or cache), and makes that data available to the application so it knows who you are and what you've been doing. This mechanism enables features like staying logged in and preserving user preferences across the inherently stateless HTTP protocol.

2. Django Database Migrations to MariaDB[4]

The usual Django migration process works identically whether using SQLite or MariaDB. Django is able to translate your model definitions into the appropriate SQL commands for the target database whether using SQLite or MariaDB. To migrate a Django application to MariaDB, you first install MariaDB on your server and create a database along with a user account that has permission to access it. You then install the mysqlclient Python library in your Django environment, which provides the database driver Django needs to communicate with MariaDB. In your Django project's settings.py file, you update the DATABASES configuration to point to MariaDB by setting ENGINE to 'django.db.backends.mysql' and filling in the database name, username, password, host address, and port number. Once configured, you run python manage.py makemigrations to create migration files based on your models, followed by python manage.py migrate to execute those migrations and build the database tables in MariaDB.

---

[1] MMDM, nd, Using HTTP cookies. Retrieved from https://developer.mozilla.org/en-US/docs/Web/HTTP/Guides/Cookies

[2] Wikipedia, nd, HTTP cookie. Retrieved from https://en.wikipedia.org/wiki/HTTP_cookie

[3] Zero to Expert, 2024, How does Django Session Middleware work. Retrieved from https://www.zerotoexpert.blog/p/how-does-django-sessionmiddleware

[4] Justin Ellingwood, 2025, How To Use MariaDB with your Django Application on CentOS 7. Website retrieved from https://www.digitalocean.com/community/tutorials/how-to-use-mariadb-with-your-django-application-on-centos-7