How Docker Works Explained  
  
Docker is a platform that simplifies application development and deployment through containerization.  
  
Here's a brief overview of how it works:  
  
1. Developer: Writes code and prepares a Dockerfile with instructions to build an image.  
2. Client: Uses Docker commands (docker build, docker pull, docker run, docker push) to interact with Docker.  
3. Dockerfile: Script containing instructions to create an image, specifying base images and configurations.  
4. Registry: Stores Docker images, which can be pulled or pushed by developers.  
5. Docker Host: Runs the Docker daemon, managing images and containers.  
6. Docker Daemon: Background service that manages the lifecycle of containers.  
7. Images: Templates for creating containers, containing applications and dependencies.  
8. Containers: Isolated environments where applications run, sharing the host system's kernel.  
  
Workflow:  
- Build: Developer creates an image from a Dockerfile.  
- Push: Image is uploaded to a registry.  
- Pull: Image is downloaded from the registry.  
- Run: Container is created and started from the image.  
  
Docker ensures applications are portable and consistent across different environments, simplifying deployment and scaling.