Install RabbitMQ (Optional if using Docker)

Install **Erlang** (dependency): <https://www.erlang.org/downloads>  
Install **RabbitMQ**:https://www.rabbitmq.com/install-windows.html  
  
rabbitmq-plugins enable rabbitmq\_management  
net start RabbitMQ  
Access UI: <http://localhost:15672>  
  
Default credentials: guest / guest  
Messaging port : 5672  
  
redis  
  
<https://github.com/tporadowski/redis/releases>  
cd C:\Redis  
redis-server.exe  
redis-cli.exe ping

Kubernetes - Install **Chocolatey** (if not installed): Open PowerShell as Admin:  
  
Set-ExecutionPolicy Bypass -Scope Process -Force; [System.Net.ServicePointManager]::SecurityProtocol = [System.Net.ServicePointManager]::SecurityProtocol -bor 3072;

iex ((New-Object System.Net.WebClient).DownloadString('https://community.chocolatey.org/install.ps1'))

1. choco install minikube kubernetes-cli -y  
   minikube start  
     
   kubectl get nodes  
   # Should show one node ready  
     
   minikube dashboard  
     
   cd C:\path\to\rag\_project  
   docker-compose up –build  
     
   Access services:  
     
   Flask app: http://localhost:5000/  
   RabbitMQ UI: <http://localhost:15672>  
   Redis: port 6379  
   Start worker (if not in Docker Compose):

python consumer.py  
  
docker build -t flask-app -f Dockerfile.app .  
docker tag flask-app:latest <your-repo>/flask-app:latest  
docker push <your-repo>/flask-app:latest  
  
kubectl apply -f k8s/rabbitmq-deployment.yaml  
kubectl apply -f k8s/redis-deployment.yaml  
kubectl apply -f k8s/flask-deployment.yaml  
kubectl apply -f k8s/consumer-deployment.yaml  
  
kubectl get svc flask-app

**Other**

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

# 2. Run Redis in Docker  
docker run -d --name my-redis -p 6379:6379 redis

# 3. Run RabbitMQ in Docker (management UI included)  
docker run -d --name my-rabbitmq -p 5672:5672 -p 15672:15672 rabbitmq:3-management

# 4. Initialize database  
py db\_setup.py

# 5. Start consumer (worker)  
py consumer.py

# 6. Start web app

py app.py

# 7. Send a test question via curl  
  
curl -X POST http://127.0.0.1:5000/ask -H "Content-Type: application/json" -d '{"user":"Srikar","question":"Who is Marcus Aurelius Antoninus?","pdf\_ids":["pdfs/sample.pdf"]}'

example to run  
 {

"user": "Srikar",  
"question": "Who is MARCUS AURELIUS ANTONINUS?",   
 "pdf\_ids": ["pdfs/sample.pdf"]

}