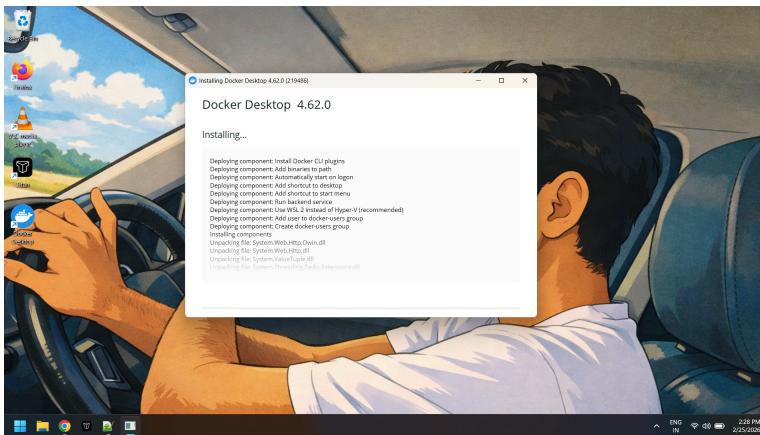


Docker Desktop Setup Procedure:

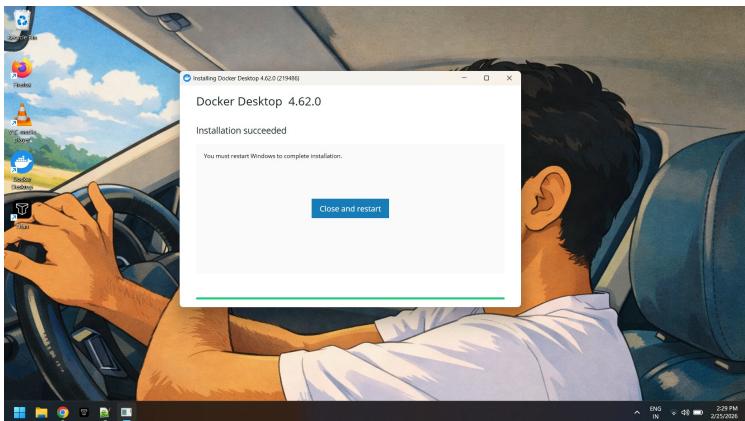
- Download **Docker Desktop** from the following URL:
<https://www.docker.com/products/docker-desktop/>
- Install Docker Desktop.



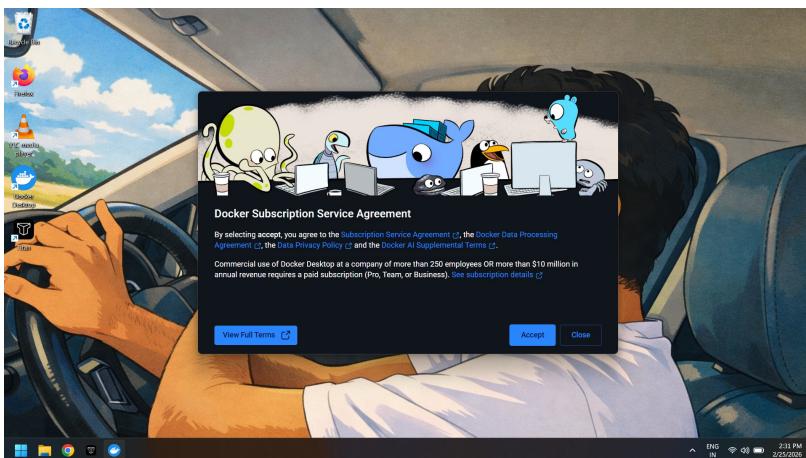
- Wait while the files are unpacked. The installation process will start automatically.



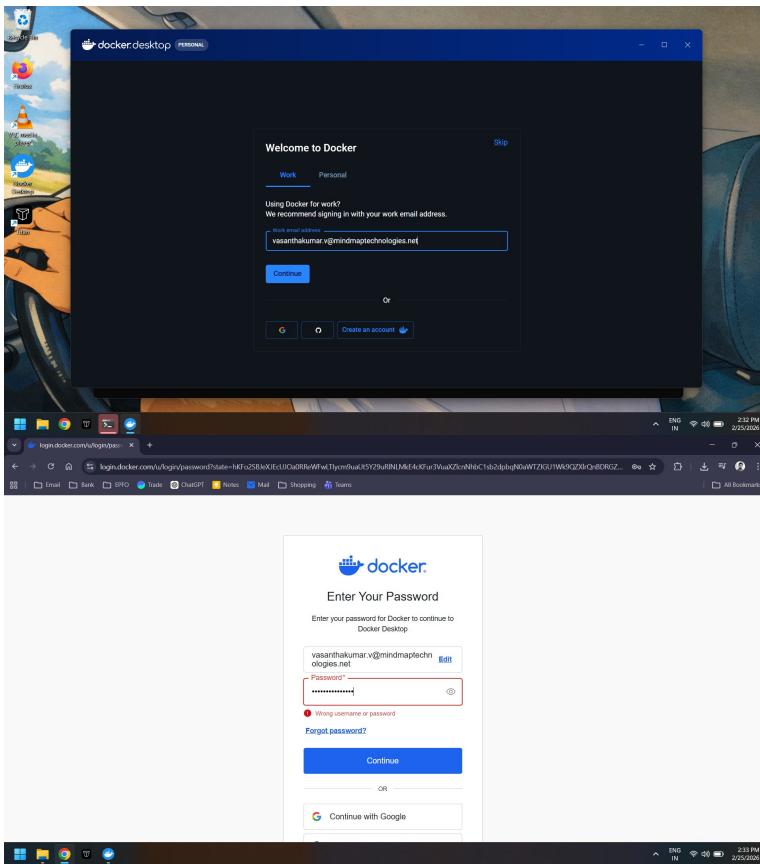
- After the installation is completed successfully, click **Close & Restart**.



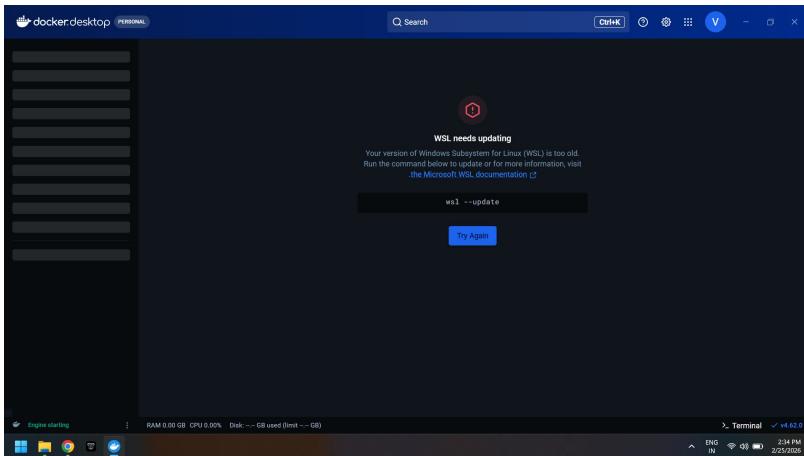
- Open Docker Desktop and accept the terms and conditions.



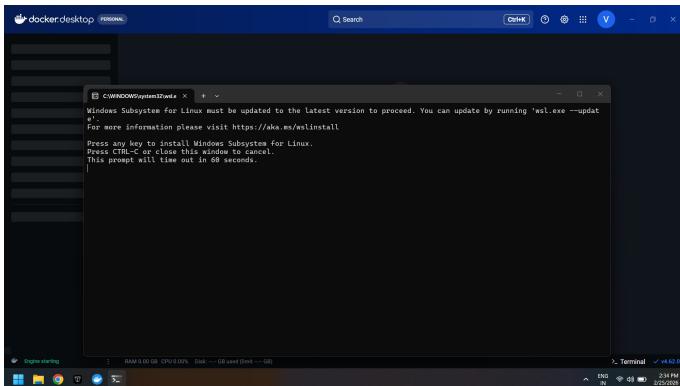
- Enter your email ID and password, then click **Continue**.



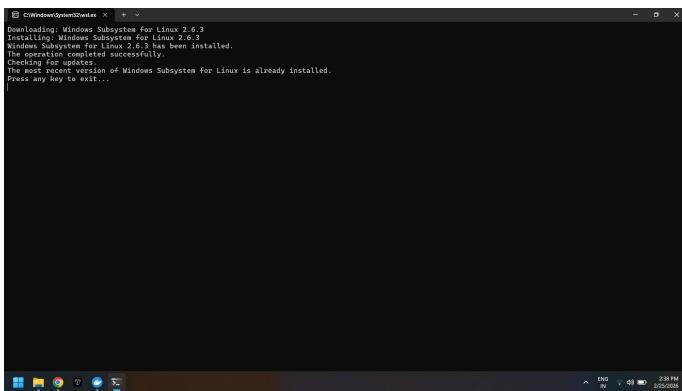
- Once Docker opens, click “Try again” for the WSL update if prompted.



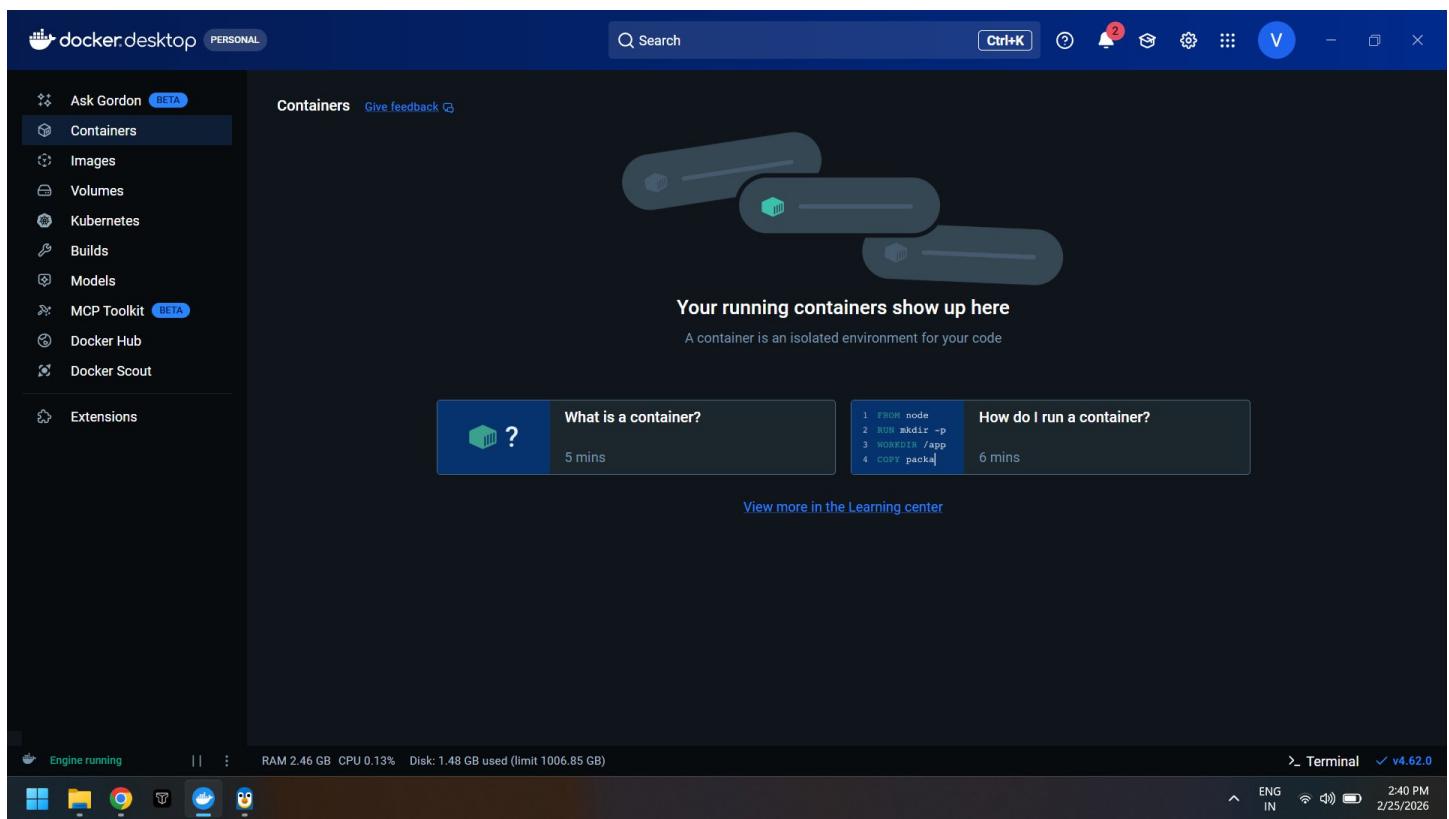
- When the WSL window opens, press any key to install the Windows Subsystem for Linux.



- After the installation is completed, press any key to exit.



- Go back to Docker Desktop, click “Try again” if required, wait for it to refresh, and then navigate to **Containers**.

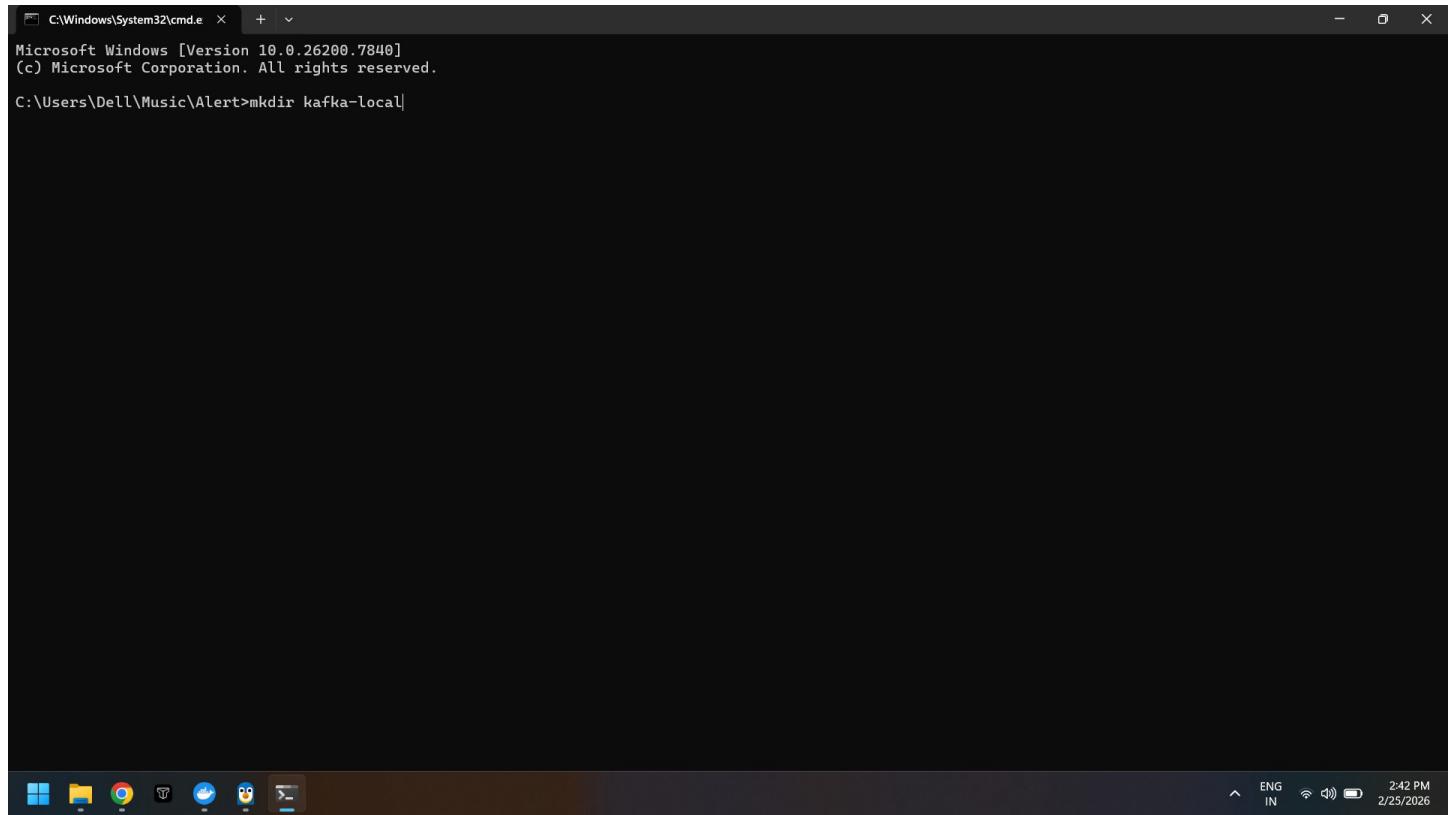


Kafka Local Setup procedure:

Go to any directory on your local system and open **Command Prompt** or **PowerShell**.

Command 1:

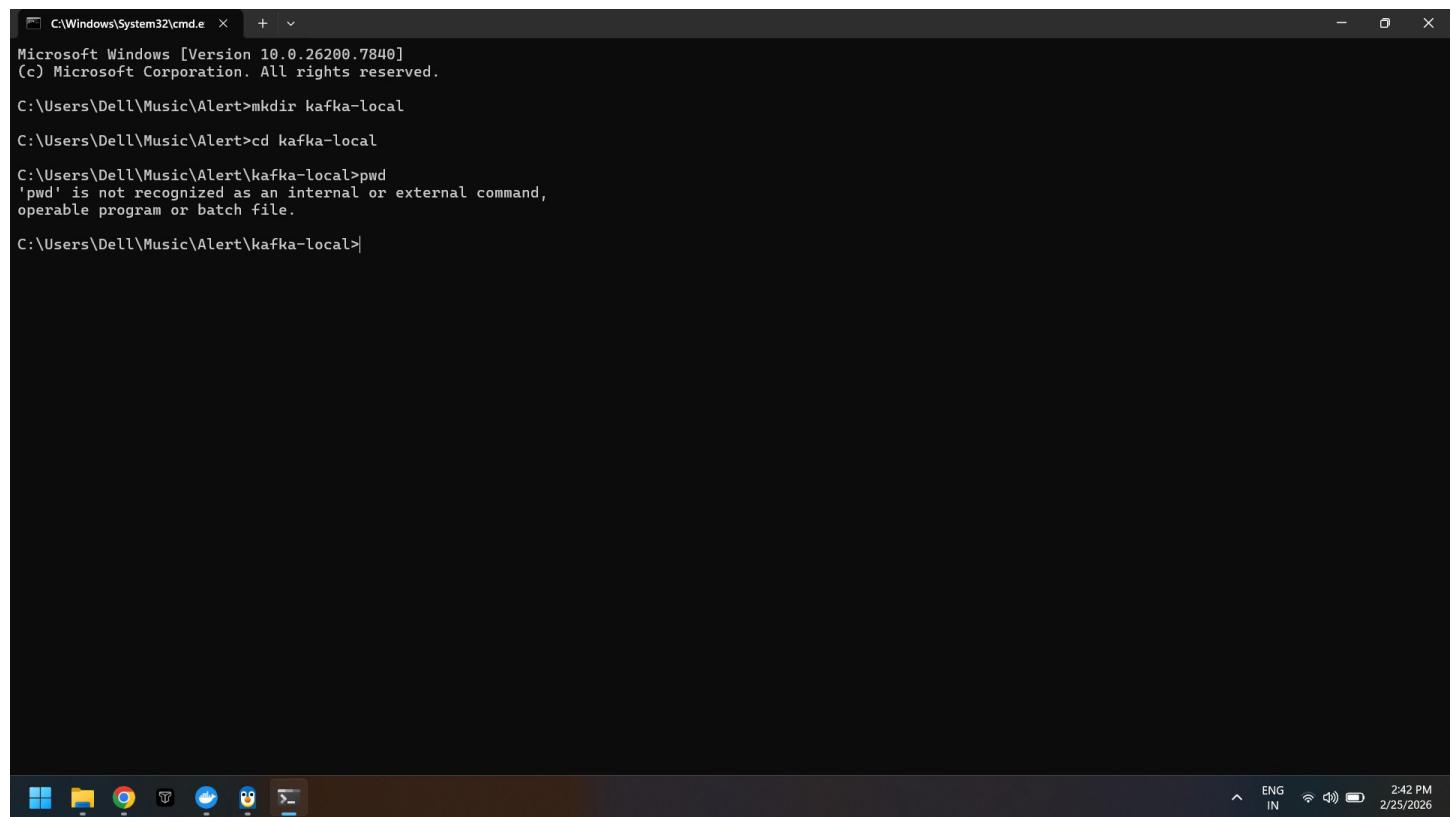
```
mkdir kafka-local
```



A screenshot of a Windows Command Prompt window titled "C:\Windows\System32\cmd.exe". The window shows the following text:
Microsoft Windows [Version 10.0.26200.7840]
(c) Microsoft Corporation. All rights reserved.
C:\Users\DELL\Music\Alert>mkdir kafka-local

Command 2:

```
cd kafka-local
```



A screenshot of a Windows Command Prompt window titled "C:\Windows\System32\cmd.e". The window shows the following text:

```
C:\Windows\System32\cmd.e + 
Microsoft Windows [Version 10.0.26200.7840]
(c) Microsoft Corporation. All rights reserved.

C:\Users\Dell\Music\Alert>mkdir kafka-local
C:\Users\Dell\Music\Alert>cd kafka-local
C:\Users\Dell\Music\Alert\kafka-local>pwd
'pwd' is not recognized as an internal or external command,
operable program or batch file.

C:\Users\Dell\Music\Alert\kafka-local>
```

The window has a dark theme. The taskbar at the bottom shows icons for File Explorer, Task View, Google Chrome, and others. The system tray shows battery status, network connection, and the date and time (2/25/2026, 2:42 PM).

Command 3:

```
notepad docker-compose.yml
```

Once Notepad opens, paste the required Docker Compose content and save the file.

```
C:\Windows\System32\cmd.e > + <
Microsoft Windows [Version 10.0.26200.7840]
(c) Microsoft Corporation. All rights reserved.

C:\Users\DELL\Music\Alert>mkdir kafka-local
C:\Users\DELL\Music\Alert>cd kafka-local
C:\Users\DELL\Music\Alert\kafka-local>pwd
'pwd' is not recognized as an internal or external command,
operable program or batch file.

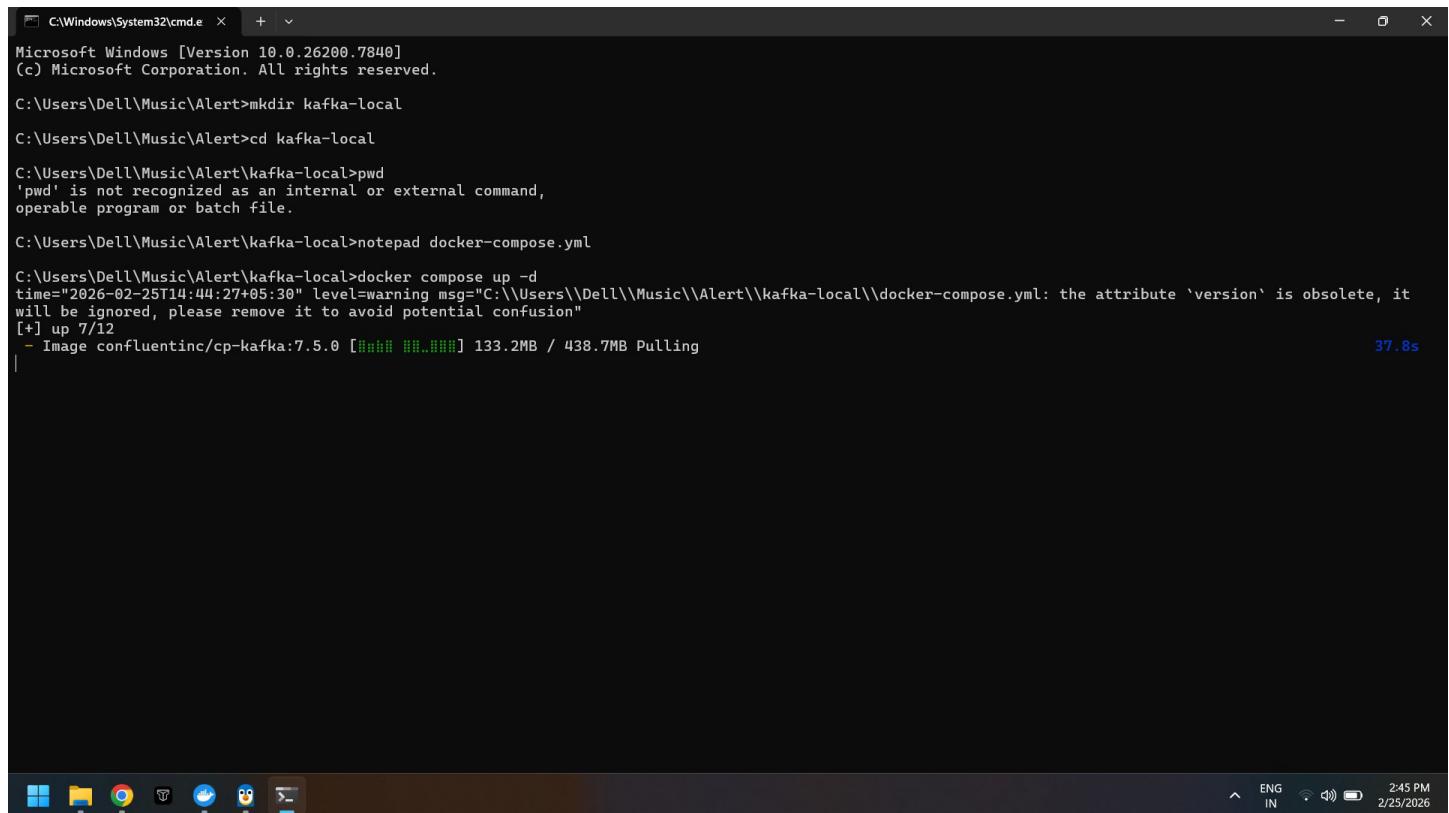
C:\Users\DELL\Music\Alert\kafka-local>notepad docker-compose.yml
C:\Users\DELL\Music\Alert\kafka-local>
```

```
File Edit View
version: '3.8'

services:
  kafka:
    image: confluentinc/cp-kafka:7.5.0
    container_name: kafka-broker
    ports:
      - "9092:9092"
      - "9093:9093"
    environment:
      KAFKA_NODE_ID: 1
      KAFKA_PROCESS_ROLES: broker,controller
      KAFKA_CONTROLLER_QUORUM_VOTERS: 1@kafka:9093
    Kafka_LISTENERS: PLAINTEXT://0.0.0.0:9092,CONTROLLER://0.0.0.0:9093
    Kafka_ADVERTISED_LISTENERS: PLAINTEXT://localhost:9092
    Kafka_LISTENER_SECURITY_PROTOCOL_MAP: CONTROLLER:PLAINTEXT,PLAINTEXT:PLAINTEXT
    Kafka_CONTROLLER_LISTENER_NAMES: CONTROLLER
    Kafka_INTER_BROKER_LISTENER_NAME: PLAINTEXT
    Kafka_NUM_PARTITIONS: 8
    Kafka_AUTO_CREATE_TOPICS_ENABLE: 'false'
    Kafka_OFFSETS_TOPIC_REPLICATION_FACTOR: 1
    Kafka_TRANSACTION_STATE_LOG_REPLICATION_FACTOR: 1
    Kafka_TRANSACTION_STATE_LOG_MIN_ISR: 1
    cluster_id: MKU30EVBNTwNTJENDM2Qk # static cluster id for CRaft
  healthcheck:
    test: ["CMD", "kafka-topics", "--bootstrap-server", "localhost:9092", "--list"]
    interval: 10s
    timeout: 5s
    retries: 5
```

Command 4:

```
docker compose up -d
```



```
C:\Windows\System32\cmd.exe + Microsoft Windows [Version 10.0.26200.7840] (c) Microsoft Corporation. All rights reserved. C:\Users\Del\Music\Alert>mkdir kafka-local C:\Users\Del\Music\Alert>cd kafka-local C:\Users\Del\Music\Alert\kafka-local>pwd 'pwd' is not recognized as an internal or external command, operable program or batch file. C:\Users\Del\Music\Alert\kafka-local>notepad docker-compose.yml C:\Users\Del\Music\Alert\kafka-local>docker compose up -d time="2026-02-25T14:44:27+05:30" level=warning msg="C:\\\\Users\\\\Del\\\\Music\\\\Alert\\\\kafka-local\\\\docker-compose.yml: the attribute 'version' is obsolete, it will be ignored, please remove it to avoid potential confusion" [+]: up 7/12 - Image confluentinc/cp-kafka:7.5.0 [==== ##..##] 133.2MB / 438.7MB Pulling 37.8s |
```

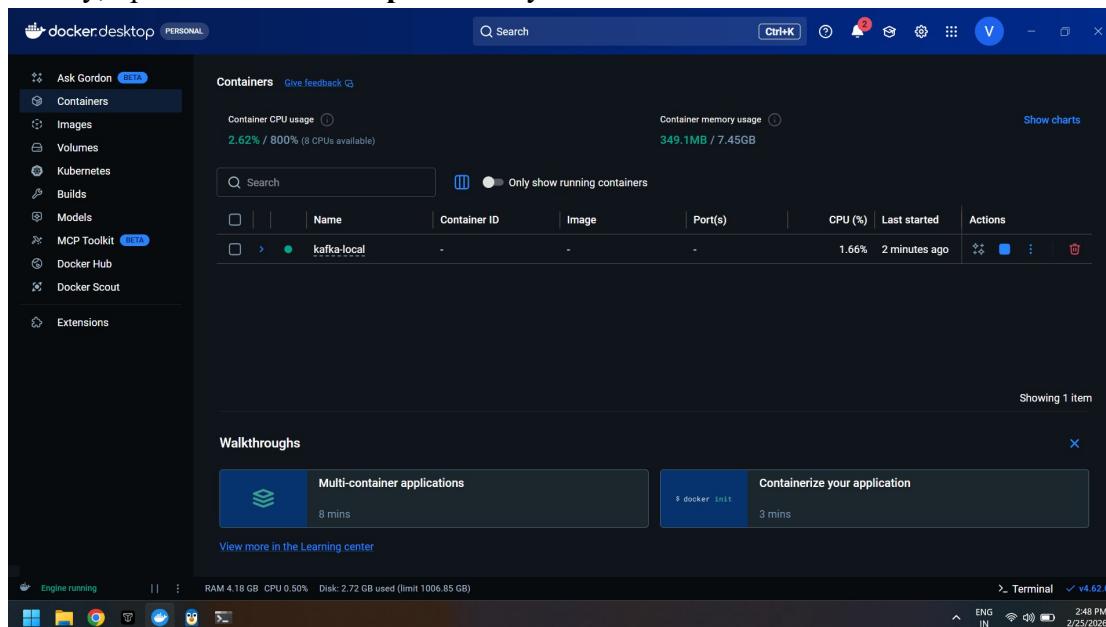
Command 5 (Optional – to check logs):

```
docker logs -f kafka-broker
```

Command 6:

```
docker ps
```

Finally, open **Docker Desktop** and verify whether the Kafka broker container is running.



```
C:\Windows\System32\cmd.e > + <
Microsoft Windows [Version 10.0.26200.7840]
(c) Microsoft Corporation. All rights reserved.

C:\Users\Del\Music\Alert>mkdir kafka-local
C:\Users\Del\Music\Alert>cd kafka-local
C:\Users\Del\Music\Alert\kafka-local>pwd
'pwd' is not recognized as an internal or external command,
operable program or batch file.

C:\Users\Del\Music\Alert\kafka-local>notepad docker-compose.yml

C:\Users\Del\Music\Alert\kafka-local>docker compose up -d
time="2026-02-25T14:44:27+05:30" level=warning msg="C:\\\\Users\\\\Del\\\\Music\\\\Alert\\\\kafka-local\\\\docker-compose.yml: the attribute 'version' is obsolete, it will be ignored, please remove it to avoid potential confusion"
[+] up 14/14
  ✓ Image confluentinc/cp-kafka:7.5.0 Pulled
  ✓ Network kafka-local_default Created
  ✓ Container kafka-broker Created
161.0s
0.1s
0.7s

C:\Users\Del\Music\Alert\kafka-local>docker ps
CONTAINER ID IMAGE COMMAND CREATED STATUS PORTS
100011ceba4b confluentinc/cp-kafka:7.5.0 "/etc/confluent/dock..." About a minute ago Up About a minute (healthy) 0.0.0.0:9092-9093->9092-9093/tcp, [::]:9092-9093->9092-9093/tcp kafka-broker

C:\Users\Del\Music\Alert\kafka-local>
```

The screenshot shows the Docker Desktop interface with the 'Compose file viewer' open. The left sidebar includes options like Ask Gordon, Containers, Images, Volumes, Kubernetes, Builds, Models, MCP Toolkit, Docker Hub, Docker Scout, and Extensions. The main area displays the contents of the docker-compose.yml file:

```
version: '3.8'
services:
  kafka:
    image: confluentinc/cp-kafka:7.5.0
    container_name: kafka-broker
    ports:
      - "9092:9092"
      - "9093:9093"
    environment:
      KAFKA_NODE_ID: 1
      KAFKA_PROCESS_ROLES: broker,controller
      KAFKA_CONTROLLER_QUORUM_VOTERS: 1@kafka:9093
      KAFKA_LISTENERS: PLAINTEXT://0.0.0.0:9092,CONTROLLER://0.0.0.0:9093
      KAFKA_ADVERTISED_LISTENERS: PLAINTEXT://localhost:9092
      KAFKA_LISTENER_SECURITY_PROTOCOL_MAP: CONTROLLER:PLAINTEXT,PLAINTEXT:PLAINTEXT
      KAFKA_CONTROLLER_LISTENER_NAMES: CONTROLLER
      KAFKA_INTER_BROKER_LISTENER_NAME: PLAINTEXT
      KAFKA_NUM_PARTITIONS: 8
      KAFKA_AUTO_CREATE_TOPICS_ENABLE: 'false'
      KAFKA_OFFSETS_TOPIC_REPLICATION_FACTOR: 1
      KAFKA_TRANSACTION_STATE_LOG_REPLICATION_FACTOR: 1
      KAFKA_TRANSACTION_STATE_LOG_MIN_ISR: 1
      CLUSTER_ID: MkU3OEVBNtCwNTJENDM2Qk # static cluster id for KRaft
```

The status bar at the bottom indicates 'Engine running', system resources (RAM 4.19 GB, CPU 0.13%, Disk 2.72 GB used), and the Docker version v4.62.0.

Kafka POC – Step-By-Step Guide

Project: Notification Event System

Topic: notifications.events

Partitions: 8

Consumer Groups:

- notification-cg-sms
- notification-cg-email

Environment:

- Windows 10
- Docker Desktop
- Confluent Kafka 7.5.0 (KRaft mode)
- Kafka UI (Provectus)

📋 Kafka POC – Step-By-Step Guide

Project: Notification Event System

Topic: notifications.events

Partitions: 8

Consumer Groups:

- notification-cg-sms
- notification-cg-email

Environment:

- Windows 10
- Docker Desktop
- Confluent Kafka 7.5.0 (KRaft mode)
- Kafka UI (Provectus)

✓ STEP 1 — Setup Docker Compose

Use docker-compose with:

- Kafka (KRaft mode)
- Kafka UI
- 8 partitions default
- Auto-topic creation disabled
- Persistent volume

Start services:

```
docker compose down -v  
docker compose up -d
```

Verify:

```
docker ps
```

You should see:

- kafka-broker
- kafka-ui

```
D:\Task1\kafka-local>docker ps
CONTAINER ID IMAGE COMMAND CREATED STATUS PORTS
08687aa28800 provectuslabs/kafka-ui:latest "/bin/sh -c 'java --..." 23 minutes ago Up 23 minutes 0.0.0.0:8081->8080
7c11249c5cda confluentinc/cp-kafka:7.5.0 "/etc/confluent/dock..." 23 minutes ago Up 23 minutes (healthy) 0.0.0.0:9092->9092
/tcp, [::]:9092->9092/tcp kafka-broker
```

<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	kafka-local	-	-	-	51.14%	24 minut	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	kafka-broker	7c11249c5cda	confluentinc/cp-kafka:7.5.0	"/etc/confluent/dock..."	50.87%	24 minut	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	kafka-ui	08687aa28800	provectuslabs/kafka-ui:latest	"/bin/sh -c 'java --..."	0.27%	24 minut	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>

✓ STEP 2 — Open Kafka UI

Open browser:

<http://localhost:8081>

Verify:

- Cluster: local-alert-cluster
- Brokers: 1
- Status: Online

The screenshot shows the Apache Kafka UI dashboard for the 'local-alert-cluster'. The left sidebar lists 'Dashboard', 'local-alert-clu...', 'Brokers', 'Topics', and 'Consumers'. The main dashboard area displays the following information:

- Brokers:** 1 clusters (Online)
- Topics:** 2
- Production:** 0 Bytes
- Consumption:** 0 Bytes

At the bottom of the dashboard, there is a table with columns: Cluster name, Version, Brokers count, Partitions, Topics, Production, and Consumption. The data for the single broker is:

Cluster name	Version	Brokers count	Partitions	Topics	Production	Consumption
local-alert-cluster	3.5-IV2	1	24	2	0 Bytes	0 Bytes

STEP 3 — Create Topic

Since auto-topic creation is disabled:

```
docker exec -it kafka-broker kafka-topics --bootstrap-server localhost:9092 --create --topic notifications.events --partitions 8 --replication-factor 1
```

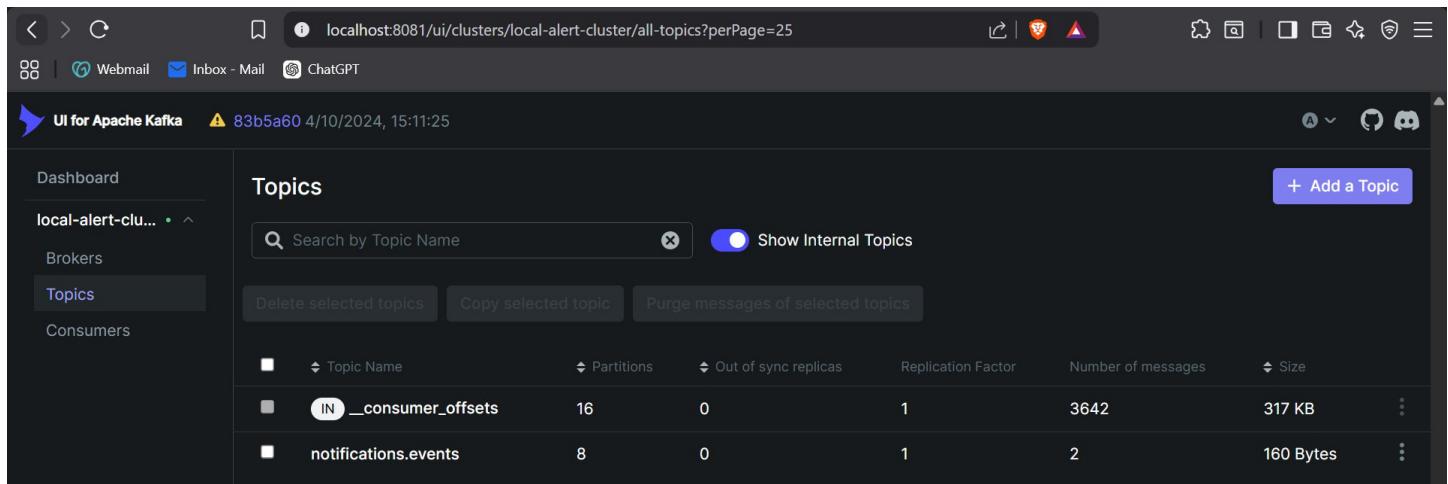
Verify:

```
docker exec -it kafka-broker kafka-topics --bootstrap-server localhost:9092 --describe --topic notifications.events
```

Confirm:

PartitionCount: 8

Refresh UI → Topics → confirm topic visible.



Topic Name	Partitions	Out of sync replicas	Replication Factor	Number of messages	Size
_consumer_offsets	16	0	1	3642	317 KB
notifications.events	8	0	1	2	160 Bytes

✓ STEP 4 — Start Consumer Group Simulation

Terminal 1 – SMS Consumer

```
docker exec -it kafka-broker kafka-console-consumer --bootstrap-server localhost:9092 --topic notifications.events --group notification-cg-sms --property print.partition=true --property print.key=true
```

```
D:\Task1\kafka-local>docker exec -it kafka-broker kafka-console-consumer --bootstrap-server localhost:9092 --topic notifications.events --group notification-cg-sms --property print.partition=true --property print.key=true
Partition:0      1001      sms_test
|
```

Terminal 2 – Email Consumer

```
docker exec -it kafka-broker kafka-console-consumer --bootstrap-server localhost:9092 --topic notifications.events --group notification-cg-email --property print.partition=true --property print.key=true
```

```
D:\Task1\kafka-local>docker exec -it kafka-broker kafka-console-consumer --bootstrap-server localhost:9092 --topic notifications.events --group notification-cg-email --property print.partition=true --property print.key=true
Partition:0      1001      sms_test
```

Refresh UI → Consumers.

You should see:

Group	Members	State
notification-cg-sms	1	STABLE
notification-cg-email	1	STABLE

The screenshot shows the Apache Kafka UI interface. The top navigation bar includes links for 'Webmail', 'Inbox - Mail', and 'ChatGPT'. The main content area is titled 'Consumers' and displays two consumer groups: 'notification-cg-email' and 'notification-cg-sms'. Each group has one member and is in a 'STABLE' state. The UI features a search bar and various filter and sorting options.

✓ STEP 5 — Produce Test Messages

Option A — From CLI

```
docker exec -it kafka-broker kafka-console-producer --bootstrap-server localhost:9092 --topic notifications.events --property parse.key=true --property key.separator=:
```

Send:

```
1001:sms_customer_created
1001:email_customer_created
1002:sms_loan_disbursed
1003:email_past_due
```

```
D:\Task1\kafka-local>docker exec -it kafka-broker kafka-console-producer --bootstrap-server localhost:9092 --topic notifications.events --property parse.key=true --property key.separator=:
>1001:sms_customer_created
>1001:email_customer_created
>1002:sms_loan_disbursed
>1003:email_past_due
>1004:sms_tph_transfer
```

Option B — From UI

Topics → notifications.events → Produce Message

Key:

1001

Value:

sms_test

The screenshot shows two instances of the UI for Apache Kafka interface.

Top Window (Topic Overview):

- Header:** UI for Apache Kafka, 83b5a60 4/10/2024, 15:11:25
- Left Sidebar:** Dashboard, local-alert-clu..., Brokers, **Topics** (selected), Consumers.
- Current Topic:** Topics / notifications.events
- Overview Tab:** Partitions (8), Replication Factor (1), URP (0), In Sync Replicas (8 of 8), Type (External), Segment Size (160 Bytes).
- Segment Count:** 8, Clean Up Policy (DELETE), Message Count (2).
- Partition Data:** Shows partitions 0-4 with their respective replicas, first offsets, next offsets, and message counts.

Bottom Window (Produce Message Dialog):

- Header:** UI for Apache Kafka, 83b5a60 4/10/2024, 15:11:25
- Left Sidebar:** Dashboard, local-alert-clu..., Brokers, **Topics** (selected), Consumers.
- Topic Selection:** Topics / notifications.events
- Produce Message Dialog:**
 - Partition:** Partition #0
 - Key Serde:** String
 - Value Serde:** String
 - Content:** Key: 1 1001, Value: 1 SMS_Testing
 - Headers:** 1 [{}]
 - Buttons:** Produce Message, Close

✓ STEP 6 — Validate Partition Behavior

Observe in consumer terminals:

Partition: X Key: 1002

Important Validation:

- Same key → same partition
- Both groups consume all partitions
- Offsets are independent per group

```
D:\Task1\kafka-local>docker exec -it kafka-broker kafka-console-consumer --bootstrap-server localhost:9092 --topic notifications.events --group notification-cg-sms --property print.partition=true --property print.key=true
Partition:0    1001    sms_test
Partition:0    1002    SMS_Testing
```

✓ STEP 7 — Check Consumer Lag

UI → Consumers → click group

You will see:

| Partition | Current Offset | Log End Offset | Lag |

Lag formula:

$\text{Lag} = \text{LogEndOffset} - \text{CurrentOffset}$

When consumers are active → Lag = 0

Consumer Group ID	Active Consumers	Consumer Lag	Coordinator	State
notification-cg-sms	1	0	1	STABLE
notification-cg-email	1	0	1	STABLE

✓ STEP 8 — Simulate Lag

1. Stop SMS consumer (close terminal).
2. Produce 10 messages.
3. Refresh UI → Consumers → notification-cg-sms

You will see:

Lag > 0

1. Restart SMS consumer.

Lag will reduce back to 0.

This validates:

- Offset persistence
- Lag calculation
- Consumer recovery

✓ STEP 9 — Test Rebalance

Open second SMS consumer:

```
docker exec -it kafka-broker kafka-console-consumer --bootstrap-server localhost:9092 --topic notifications.events --group notification-cg-sms
```

Refresh UI → Consumers.

Now you will see:

- Members = 2
- Partitions redistributed

Max consumers per group = 8 (because 8 partitions).

```
D:\Task1\kafka-local>docker exec -it kafka-broker kafka-console-consumer --bootstrap-server localhost:9092 --topic notifications.events --group notification-cg-sms
```

Consumer Group ID	Active Consumers	Consumer Lag	Coordinator	State
notification-cg-sms	2	0	1	STABLE
notification-cg-email	1	0	1	STABLE

✓ STEP 10 — Architecture Validation Summary

Your POC now confirms:

- ✓ 8 partitions parallelism
- ✓ 2 independent consumer groups
- ✓ Key-based routing (customerId)
- ✓ Offset isolation
- ✓ Rebalance behavior
- ✓ Lag monitoring
- ✓ Recovery behavior
- ✓ UI monitoring

🌐 Important Kafka Concepts Learned

1 Consumer Group Isolation

Each group maintains separate offsets in:

`_consumer_offsets`

2 Partition Parallelism

Max consumers per group = number of partitions.

3 Rebalance Triggered When:

- Consumer joins
- Consumer leaves
- Consumer crashes

4 Lag Appears When:

- Messages exist
- Consumer not processing

Below is a **clean, structured installation guide** for:

- ✓ Java (JDK)
- ✓ Maven
- ✓ Spring Boot (via Spring Initializr / CLI)
- ✓ Windows
- ✓ macOS

This is written as a proper document you can reuse.

Java (JDK) Installation Guide

Recommended Version: **Java 17 or Java 21 (LTS)**

(Spring Boot 3.x requires Java 17+)

A. Install Java on Windows

Step 1 — Download JDK

Download from: [JDK download](#)

Choose:

- Version: 21
- OS: Windows
- Architecture: x64
- Package Type: MSI

Step 2 — Install

- Double-click .msi
- Follow installer

Step 3 — Set Environment Variables

Open:

Control Panel → System → Advanced System Settings → Environment Variables

Add New System Variable:

Variable Name: JAVA_HOME

Variable Value: C:\Program Files\Java\jdk-21.0.10 (Note : use your system path)

Edit PATH → Add:

```
%JAVA_HOME%\bin
```

Click OK.

Step 4 — Verify

Open CMD:

```
java -version
```

You should see:

```
openjdk version "21"
```

B. Install Java on macOS

Step 1 — Install via Homebrew (Recommended)

If Homebrew not installed:

```
/bin/bash -c "$(curl -fsSL  
https://raw.githubusercontent.com/Homebrew/install/HEAD/install.sh)"
```

Install Java:

```
brew install openjdk@21
```

Step 2 — Set JAVA_HOME

Add to:

```
~/.zshrc      (or ~/.bash_profile)
```

Add:

```
export JAVA_HOME=$(/usr/libexec/java_home -v 21)  
export PATH=$JAVA_HOME/bin:$PATH
```

Reload:

```
source ~/.zshrc
```

Step 3 — Verify

```
java -version
```

Maven Installation Guide

Recommended Version: 3.9+

A. Install Maven on Windows

Step 1 — Download Maven

Download from:

<https://maven.apache.org/download.cgi>

Download:

Binary zip archive

Step 2 — Extract

Extract to:

C:\apache-maven-3.9.9

Step 3 — Set Environment Variables

Add:

Variable Name: MAVEN_HOME

Variable Value: C:\apache-maven-3.9.9

Edit PATH → Add:

%MAVEN_HOME%\bin

Step 4 — Verify

```
mvn -version
```

Expected:

```
Apache Maven 3.9.x
```

```
Java version: 21
```

macOS

• B. Install Maven on macOS

Option 1 — Using Homebrew (Recommended)

```
brew install maven
```

Verify

```
mvn -version
```

Spring Boot Setup

You do NOT install Spring Boot manually in most cases.

You use:

- Spring Initializr
- Maven dependency
- Or Spring Boot CLI

Option A — Using Spring Initializr (Recommended)

Go to:

<https://start.spring.io>

Choose:

- Project: Maven
- Language: Java
- Spring Boot: Latest stable
- Java: 17 or 21

- Packaging: Jar
- Dependencies:
 - Spring Web
 - Spring Data JPA
 - PostgreSQL Driver
 - Spring for Apache Kafka (if needed)

Click:

Generate

Extract project.

Run:

```
mvn spring-boot:run
```

Option B — Install Spring Boot CLI (Optional)

Windows

Download from:

<https://github.com/spring-projects/spring-boot/releases>

Extract.

Add to PATH.

Verify:

```
spring --version
```

macOS (Homebrew)

```
brew install spring-boot
```

Verify:

```
spring --version
```

IDE Setup (Recommended)

Windows & macOS

Recommended:

- IntelliJ IDEA
- Spring Tool Suite (STS)
- VS Code with Java Extensions

Quick Verification Checklist

Run:

```
java -version
```

```
mvn -version
```

Create sample app:

```
mvn archetype:generate
```

Run Spring Boot app:

```
mvn spring-boot:run
```

Open:

<http://localhost:8080>

Recommended Versions Summary

Tool	Recommended
Java	17 or 21 LTS
Maven	3.9+
Spring Boot	3.x
IDE	STS / IntelliJ

Production-Ready Stack (Your Case)

Since you work with:

- Kafka
- Spring Boot
- PostgreSQL
- Microservices

Recommended:

Java 21

Spring Boot 3.2+

Maven 3.9+

Docker