**Kafka Integration with C#:**

**Q1: Create a Chat Application which uses Kafka as a streaming platform and consume the chat messages in the command prompt.**

**Start Zookeeper:**

cd C:\kafka\_2.13-3.6.0

.\bin\windows\zookeeper-server-start.bat .\config\zookeeper.properties

**Start Kafka Broker:**

cd C:\Users\soham\Desktop\kafka\_2.13-3.6.0

.\bin\windows\kafka-server-start.bat .\config\server.properties

**Create Kafka Topic:**

cd C:\Users\soham\Desktop\kafka\_2.13-3.6.0

.\bin\windows\kafka-topics.bat --create --topic chat-topic --bootstrap-server localhost:9092 --partitions 1 --replication-factor 1

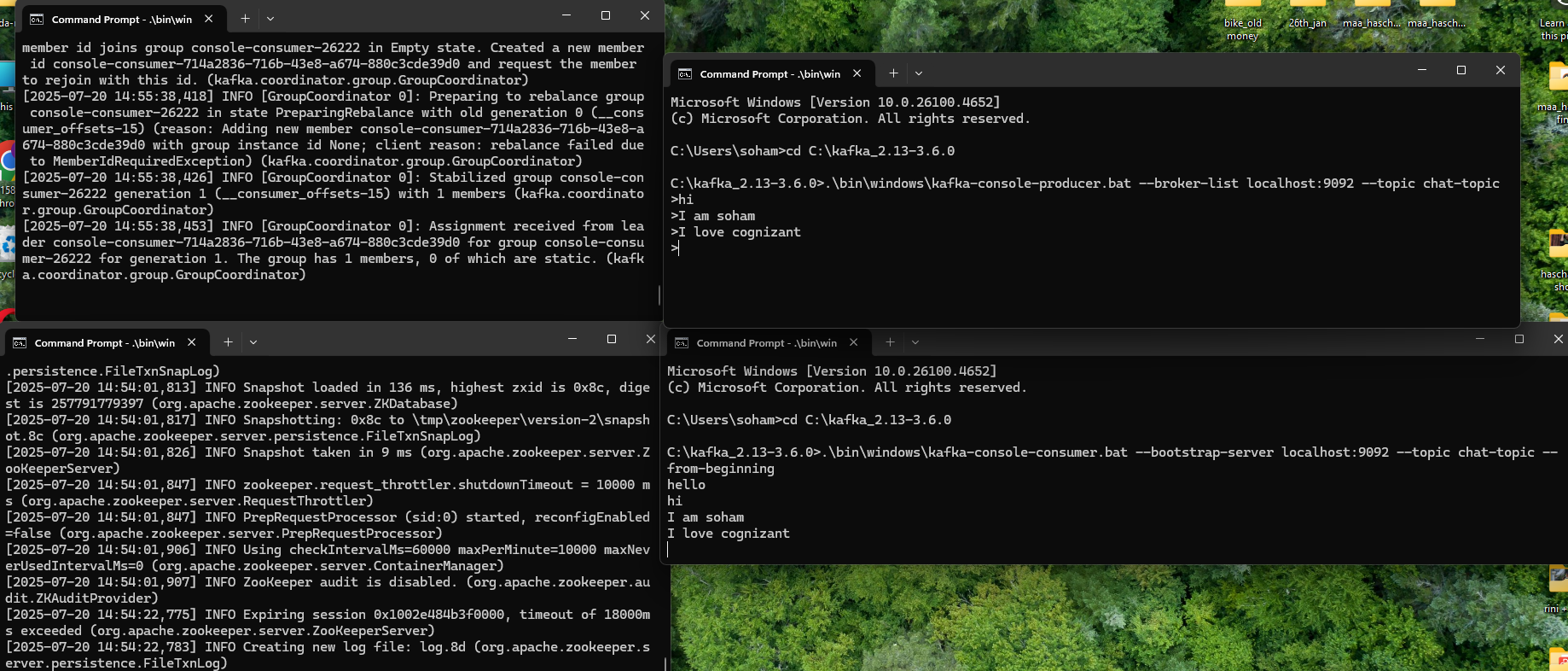
**Test Publisher:**

.\bin\windows\kafka-console-producer.bat --broker-list localhost:9092 --topic chat-topic

**Test Consumer:**

.\bin\windows\kafka-console-consumer.bat --bootstrap-server localhost:9092 --topic chat-topic --from-beginning

**OUTPUT:**

****

**Q2: Create a Chat Application using C# Windows Application using Kafka and consume the message in different client applications.**

**#Form1.Designer.cs**

namespace KafkaChatApp

{

partial class Form1

{

/// <summary>

/// Required designer variable.

/// </summary>

private System.ComponentModel.IContainer components = null;

/// <summary>

/// Clean up any resources being used.

/// </summary>

/// <param name="disposing">true if managed resources should be disposed; otherwise, false.</param>

protected override void Dispose(bool disposing)

{

if (disposing && (components != null))

{

components.Dispose();

}

base.Dispose(disposing);

}

#region Windows Form Designer generated code

/// <summary>

/// Required method for Designer support - do not modify

/// the contents of this method with the code editor.

/// </summary>

private void InitializeComponent()

{

this.button1 = new System.Windows.Forms.Button();

this.txtMessage = new System.Windows.Forms.TextBox();

this.lstChat = new System.Windows.Forms.ListBox();

this.Load += new System.EventHandler(this.Form1\_Load);

this.SuspendLayout();

//

// button1

//

this.button1.Location = new System.Drawing.Point(343, 190);

this.button1.Name = "button1";

this.button1.Size = new System.Drawing.Size(123, 45);

this.button1.TabIndex = 0;

this.button1.Text = "btnSend";

this.button1.UseVisualStyleBackColor = true;

this.button1.Click += new System.EventHandler(this.btnSend\_Click);

// txtMessage

//

this.txtMessage.Location = new System.Drawing.Point(102, 199);

this.txtMessage.Name = "txtMessage";

this.txtMessage.Size = new System.Drawing.Size(235, 26);

this.txtMessage.TabIndex = 1;

//

// lstChat

//

this.lstChat.FormattingEnabled = true;

this.lstChat.ItemHeight = 20;

this.lstChat.Location = new System.Drawing.Point(102, 49);

this.lstChat.Name = "lstChat";

this.lstChat.Size = new System.Drawing.Size(355, 124);

this.lstChat.TabIndex = 2;

//

// Form1

//

this.AutoScaleDimensions = new System.Drawing.SizeF(9F, 20F);

this.AutoScaleMode = System.Windows.Forms.AutoScaleMode.Font;

this.ClientSize = new System.Drawing.Size(800, 450);

this.Controls.Add(this.lstChat);

this.Controls.Add(this.txtMessage);

this.Controls.Add(this.button1);

this.Name = "Form1";

this.Text = "Form1";

this.ResumeLayout(false);

this.PerformLayout();

}

#endregion

private System.Windows.Forms.Button button1;

private System.Windows.Forms.TextBox txtMessage;

private System.Windows.Forms.ListBox lstChat;

}

}

**#From1.cs**

using System;

using System.Windows.Forms;

using System.Threading;

using System.Threading.Tasks;

using Confluent.Kafka;

namespace KafkaChatApp

{

public partial class Form1 : Form

{

public Form1()

{

InitializeComponent();

}

private async void btnSend\_Click(object sender, EventArgs e)

{

var config = new ProducerConfig

{

BootstrapServers = "localhost:9092"

};

using (var producer = new ProducerBuilder<Null, string>(config).Build())

{

var message = txtMessage.Text.Trim();

if (!string.IsNullOrEmpty(message))

{

await producer.ProduceAsync("chat", new Message<Null, string> { Value = message });

lstChat.Items.Add($"You: {message}");

txtMessage.Clear();

}

}

}

private void Form1\_Load(object sender, EventArgs e)

{

Task.Run(() => ConsumeMessages());

}

private void ConsumeMessages()

{

var config = new ConsumerConfig

{

BootstrapServers = "localhost:9092",

GroupId = Guid.NewGuid().ToString(),

AutoOffsetReset = AutoOffsetReset.Earliest

};

using (var consumer = new ConsumerBuilder<Ignore, string>(config).Build())

{

consumer.Subscribe("chat");

CancellationTokenSource cts = new CancellationTokenSource();

try

{

MessageBox.Show("Consumer started");

while (true)

{

var cr = consumer.Consume(cts.Token);

this.Invoke(new Action(() =>

{

lstChat.Items.Add($"Friend: {cr.Message.Value}");

}));

}

}

catch (Exception ex)

{

MessageBox.Show("Consumer error: " + ex.Message);

consumer.Close();

}

}

}

}

}

**#Program.cs**

using KafkaChatApp;

using System;

using System.Collections.Generic;

using System.Linq;

using System.Threading.Tasks;

using System.Windows.Forms;

namespace KafkaChatAppl

{

internal static class Program

{

/// <summary>

/// The main entry point for the application.

/// </summary>

[STAThread]

static void Main()

{

Application.EnableVisualStyles();

Application.SetCompatibleTextRenderingDefault(false);

Application.Run(new Form1());

}

}

}

**OUTPUT:**

