**Case Study:**

A company would like to setup a messaging system that automatically broadcasts to all new employees **in a specific order**.

The order of the messages is:

1. Welcome to Company XYZ!
2. Please remember to collect your laptop from the mobile clinic!
3. ANNOUNCEMENT: Stay safe and wash your hands.

Create a Kafka environment that simulates the following:

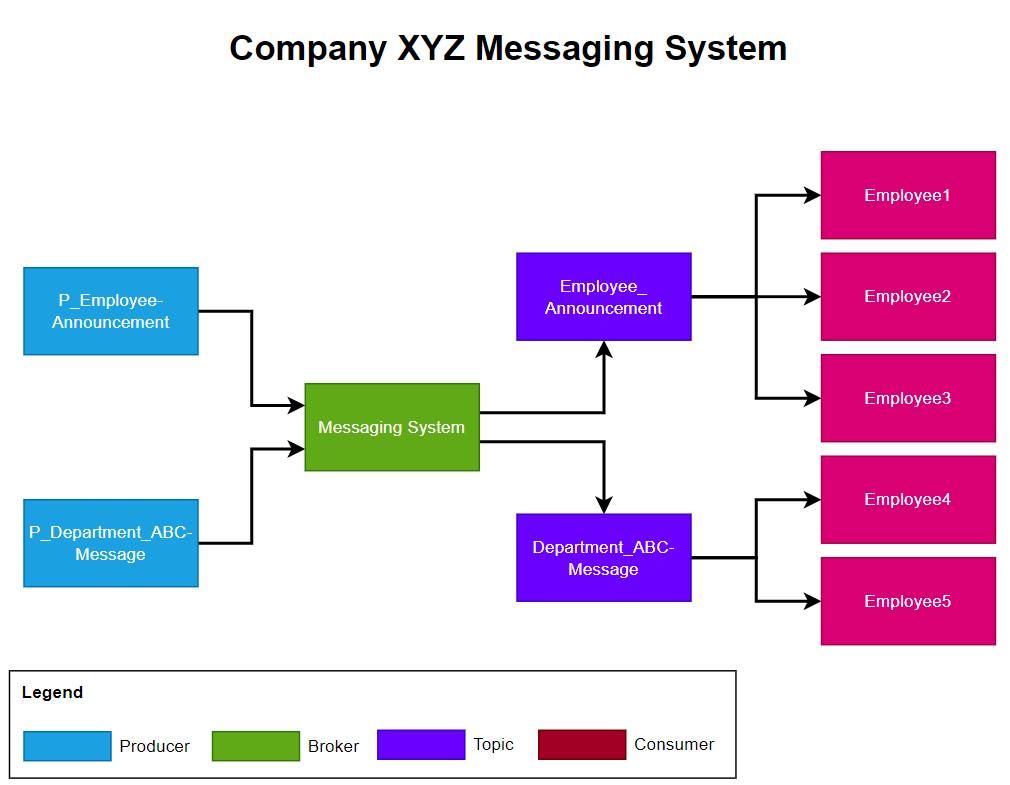
Kafka Broker = Messaging System

Consumers = Employees

All new employees to the company should receive the announcement messages in the specified order above. **Whenever there is a new message, employees should only receive messages that they have never received before.**

**Optional Task (Bonus points):** Create another group message that sends a different set of messages to a specific department of people.

# Answer



*Logical Diagram: Messaging System*

**Environment Readiness**

1. Installation of Kafka application
2. Create a folder in Drive C as below

C:\kafka

1. In C:\kafka directory, download and extract kafka 3.6 from below link

<https://www.apache.org/dyn/closer.cgi?path=/kafka/3.6.0/kafka_2.13-3.6.0.tgz>

1. Create **kafka-logs** and **kafkazookeeper** folder and configure both server and zookeeper configuration files to point to new folders.
2. Create Kafka environment - ***Messaging System***
3. To start Kafka Zookeeper, navigate to kafka installation directory and execute below

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

1. To start Kafka Broker, execute below

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

1. Once the Kafka Broker is up, ensure the broker listening port is listening

*netstat -an | findstr 9092*



1. Create 2 Topics as below
2. **Employee-Announcement**

bin\windows\kafka-topics.bat --create --topic Announcement --bootstrap-server localhost:9092

1. **Department\_ABC-Message**

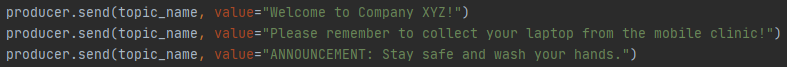
bin\windows\kafka-topics.bat --create --topic Department\_ABC --bootstrap-server localhost:9092

1. Create 5 Kafka Consumer – ***Employee*** (Python - KafkaConsumer)
2. 3x Consumer – ***Employee*** for **Employee-Announcement** Topic
3. 2x Consumer – ***Employee*** for **Department\_ABC-Message** Topic
4. Create 2 Kafka Producer – Producer (Python – KafkaProducer) for below Topics
5. 1x Producer for **P\_Employee-Announcement**
6. 1x Producer for **P\_Department\_ABC-Message**

**Use Case Simulation**

**For Use Case 1**

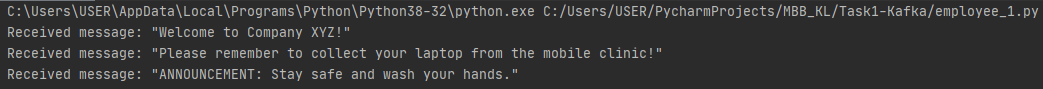
1. Producer **P\_Employee-Announcement** will produce a message to the Topic **Employee-Announcement**



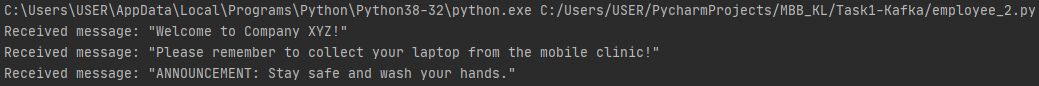
1. The **Consumer – Employee 1, 2 and 3** will receive Welcome Message from the Messaging System which is configured earlier.
2. The **Consumer – Employee** are configured with **auto\_offset\_reset='earliest'** parameters to ensure it received the messages which was not received earlier.
3. Thereafter to ensure only latest messages should be received by employee, the Consumer – Employee are configured with manual commit method.

**Example Output**

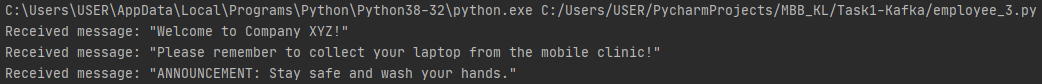
Employee1 – First Attempt



Employee2 - First Attempt



Employee3 - First Attempt



Employee3 – Second Attempt (Only receive the latest message)

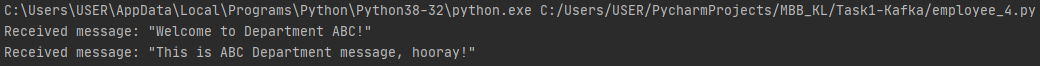


**For Bonus Use Case**

1. Producer – **P\_Department\_ABC-Message** will produce a message to the Topic **Department\_ABC-Message**
2. The **Consumer – Employee 4 and 5** will receive **Department ABC** related messages from the Messaging System which is configured earlier.
3. The **Consumer – Employee** are configured with **auto\_offset\_reset='earliest'** parameters to ensure it received the messages which was not received earlier.
4. Thereafter to ensure only latest messages should be received by employee, the **Consumer – Employee** are configured with manual commit method.

**Example Output**

Employee4 – First Attempt



Employee5 – First Attempt



Employee5 – Second Attempt (Only receive the latest message)



Py\_Scripts Folder Contains below

|  |  |  |
| --- | --- | --- |
| No | Script Name | Usage |
| 1 | producer\_anouncement.py | Produce default message to Employee-Announcement Topic |
| 2 | producer\_department-msg.py | Produce default message to Department\_ABC-Message Topic |
| 3 | employee\_1.py, employee\_2.py, employee\_3.py | Consumers for Use Case 1 |
| 4 | employee\_4.py, employee\_5.py | Consumers for Bonus Use Case |
| 5 | producer\_annoucement.bat - Shortcut | To produce new messages in Employee-Announcement Topic |
| 6 | producer\_department-msg.bat - Shortcut | To produce new messages in Department\_ABC-Message Topic |