1. We have downloaded kafka from https://www.apache.org/dyn/closer.cgi?path=/kafka/3.7.0/kafka_2.13-3.7.0.tgz
2. We extracted the tgz file to downloads and extracted it.
3. We have to start zookeeper service by going to the kafka folder and running the following command in the terminal :
    a. bin/zookeeper-server-start.sh config/zookeeper.properties

   The above command should run successfully to start the zookeeper.
4. We have to start the kafka broker service by opening a new terminal in the kafka folder and running the following command:
    a. $ bin/kafka-server-start.sh config/server.properties
5. Now as required we need to create two topics on kafka using the following commands(should be run on new terminal in the kafka folder) :
    a. $ bin/kafka-topics.sh --create --topic commentsfromreddit --bootstrap-server localhost:9092
    b. $ bin/kafka-topics.sh --create --topic wordcountfromcomments --bootstrap-server localhost:9092
    c. The above commands creates two kafka topics names commentsfromreddit and wordcountfromcomments.
6. Now, We can send messages to a kafka topic by running the following command in a new terminal opened in the kafka folder .
    a. bin/kafka-console-producer.sh --bootstrap-server localhost:9092 --topic commentsfromreddit
7. We can now read messages from kafka topic by running the following command in a new terminal opened in the kafka folder.
    a. bin/kafka-console-consumer.sh --bootstrap-server localhost:9092 --topic commentsfromreddit --from-beginning
8. We now have to setup elastic search, logstash, kibana.
9. We have downloaded elastic search from the https://www.elastic.co/downloads/elasticsearch
10. We extracted elastic search to a folder and did the following :
    a. we have disabled SSL in the config/elasticsearch.yml file by modifying the following properties as
    b. xpack.security.enabled: false
    c. xpack.security.http.ssl.enabled: false
    d. This change was made as we no longer require SSL encryption for communicatng within Elasticsearch and this simplifies the configuration part .
    e. After the above changes, we need to start Elastic search by opening a new terminal and running the following command: bin/elasticsearch
    f. After running the command, we get a password and configuration token.
    g. We can go to http://localhost:9200 to verify if elasticsearch is installed properly. This should return an output similar to

```
{
  "name" : "Chintas-MBP.lan",
  "cluster_name": "elasticsearch",
  "cluster_uuid" : "uBI6fvxXSKi27zfFvR8u0Q",
  "version" : {
    "number" : "8.13.2",
    "build_flavor" : "default",
    "build_type" : "tar",
    "build_hash" : "16cc90cd2d08a3147ce02b07e50894bc060a4cbf",
    "build_date" : "2024-04-05T14:45:26.420424304Z",
    "build_snapshot" : false,
    "lucene_version" : "9.10.0",
    "minimum_wire_compatibility_version" : "7.17.0",
    "minimum_index_compatibility_version" : "7.0.0"
  },
  "tagline" : "You Know, for Search"
}
```

11. We have downloaded kibana from the https://www.elastic.co/downloads/kibana
12. We extracted kibana to a folder and did the following :
    a. we need to start kibana by opening a new terminal and running the following command: bin/kibana
    b. Once kibana starts, we need to go to http://localhost:5601 and configure kibana using the password and token generated during the elasticsearch generation.
    c. Once, the setup is down we can see elasticsearch and kibana working on the browser for visualization.
13. We have downloaded logstash from https://www.elastic.co/downloads/logstash
14. We extracted logstash to a folder and did the following :
    a. We created a logstash.conf with config details in the same folder (refer to the logstash.conf file from the github repo)
    b. We then ran the below command in a new terminal opened inside the logstash folder
    c. bin/logstash -f logstash.conf
15. We now need to install spark using the following command
    a. pip3 install pyspark==3.5.1 this installs the latest version of pyspark which is 3.5.1
16. To setup spark locally , we have downloaded spark from https://spark.apache.org/downloads.html
17. After extracting spark, we need to go that directory and put the redditproducer and redditconsumer python files in that directory. We also need to create a checkpoint directory in the same directory.
18. After this, we need to execute the following command to run redditconsumer.py file in the spark application.
    a. Go to the spark directory that we have installed and open a new terminal and run the command
    b. spark-submit --packages org.apache.spark:spark-sql-kafka-0-10_2.12:3.5.1 --conf spark.sql.streaming.forceDeleteTempCheckpointLocation=true

redditconsumer.py /tmp/checkpoint localhost:9092 commentsfromreddit wordcountfromcomments

19. After this, we need to execute the following command to run redditproducer.py file in the spark application.
    a. Go to the spark directory that we have installed and open a new terminal and run the command
    b. python3 -u redditproducer.py commentsfromreddit localhost:9092

    We can see the data streaming using commands specified in step 5.
20. We now used kibana to visualize the data.
21. To visualize, we created a Data View by using the appropriate name(we have used wordcountfromcomments) and index pattern in our case it's (wordcountfromcomments*) and the required time stamp.
22. Now, to create a visualization, we selected the index created and words.keyword field on the horizontal axis, count field on the vertical axis, name on vertical axis as Sum of Count, sum as aggregation function.
23. We have set the properties of visualization so that the number of values is set to 10 so that we get the top 10 named entities.
24. Now we can see the visualizations in various formats which are barplots , donuts, Heat map etc.
25. We have chosen bar plots, and donuts and analyzed those for the time intervals after 15, 30, 45, and 60 minutes.