KIP-376: Implement AutoClosable on appropriate classes that want to be used in a try-with-resource statement

Status

Current state: *Accepted (2.2.0)*

Discussion thread: *Link*

JIRA: *KAFKA-7402*

Motivation

Various components in Streams have close methods but do not implement AutoCloseable. This means that they can't be used in try-with-resources blocks.

Remedying that would simplify our tests and make life easier for users as well.

KafkaStreams itself is a notable example of this, but we can take the opportunity to look for other components that make sense as AutoCloseable as well.

So these classes can be used in a try-with-resources Statement after implementing AutoCloseable.

From AutoCloseable's Javadoc:

> An object that may hold resources (such as file or socket handles) until it is closed. The {@link #close()} method of an {@code AutoCloseable} object is called automatically when exiting a {@code try}-with-resources block for which the object has been declared in the resource specification header. This construction ensures prompt release, avoiding resource exhaustion exceptions and errors that may otherwise occur.

Public Interfaces

By going over the project, here is a list that I found which can implement AutoCloseable. Suggestions are welcome.

1. org.apache.kafka.streams.KafkaStreams
2. org.apache.kafka.streams.processor.internals.RecordCollector
3. org.apache.kafka.connect.runtime.ConnectMetrics.MetricGroup
4. org.apache.kafka.connect.transforms.TimestampRouter
5. org.apache.kafka.tools.VerifiableProducer
6. org.apache.kafka.connect.runtime.WorkerConnector.ConnectorMetricsGroup
7. org.apache.kafka.common.record.MemoryRecordsBuilder
8. org.apache.kafka.common.metrics.MetricsReporter
9. org.apache.kafka.common.security.oauthbearer.internals.expiring.ExpiringCredentialRefreshingLogin
10. org.apache.kafka.common.network.KafkaChannel
11. org.apache.kafka.clients.consumer.ConsumerInterceptor
12. org.apache.kafka.common.network.Selector.SelectorMetrics
13. org.apache.kafka.clients.consumer.internals.AbstractCoordinator.HeartbeatThread

Proposed Changes

Here are an example of how it would look like:

org.apache.kafka.streams.KafkaStreams

```
public class KafkaStreams implements AutoCloseable{
...
        public void close()
...
}
```

Compatibility, Deprecation, and Migration Plan

- This change will not have backward compatibility issues. It is legal to implement this AutoClosable interface without declaring "throws Exception" on close(), so the close() method signature won't be changed.

Rejected Alternatives

- We decided not to choose Closeable(). Closeable extends AutoCloseable, and AutoCloseable throws a broader exception than Closeable, so we think that AutoCloseable is a more generic and a more compatible option.