Kafka Console Consumer Flags (kafka-console-consumer.sh): Quick Documentation

1 What this tool is

`kafka-console-consumer.sh` is a Kafka CLI utility that reads records from Kafka topics and prints them to standard output. It is commonly used for debugging, validating ingestion, and monitoring streams.

How to get the authoritative flag list for your Kafka version

Flags can differ slightly across Kafka versions/distributions. Always verify on your machine:

```
bin/kafka-console-consumer.sh --help
```

2 Common usage patterns

Consume live (new messages only)

```
bin/kafka-console-consumer.sh --bootstrap-server localhost:9092 --topic
    iot_sensor_data
```

Consume from the beginning (replay)

```
bin/kafka-console-consumer.sh --bootstrap-server localhost:9092 --topic
    iot_sensor_data --from-beginning
```

Consume a fixed number of messages and exit

```
bin/kafka-console-consumer.sh --bootstrap-server localhost:9092 --topic
    iot_sensor_data --from-beginning --max-messages 10
```

3 Flags and explanations (console consumer)

Table 1: Flags supported by `kafka-console-consumer.sh` (typical Apache Kafka distributions).

Flag	**Explanation / When to use**
`--bootstrap-server <host:port>`	Kafka broker endpoint to connect to (required for the modern consumer). Example: `localhost:9092`.
`--topic <topic>`	Consume from a single topic (common case). Example: `iot_sensor_data`.
`--from-beginning`	If there is no committed offset for the consumer group (or no group is used), start at the earliest available record rather than at the end. Useful for replaying data.
`--group <group-id>`	Run as a member of a consumer group; offsets are committed for the group. Re-running with the same group resumes from the last committed offset (unless offsets are reset).
`--partition <int>`	Consume only from a specific partition of the topic. Often used for debugging partition-specific behavior.
`--offset <n|earliest|latest>`	Start offset for consumption (often used with `--partition`). Accepts a non-negative integer offset, or `earliest` / `latest`.
`--max-messages <int>`	Maximum number of records to consume before exiting. Useful for tests and automation.
`--timeout-ms <int>`	Exit if no message is available for consumption for the specified time (milliseconds). Helpful for scripts where you don't want to block forever.
`--consumer.config <file>`	Path to a properties file containing consumer configs (e.g., security, deserializers, group settings).
`--consumer-property key=value`	Provide individual consumer configs on the command line (overrides values from `--consumer.config` for the same key). Can be repeated multiple times.
`--formatter <class>`	Fully-qualified Java class name used to format messages for display. Default is typically `kafka.tools.DefaultMessageFormatter`.
`--property <key=value>`	Properties for the message formatter (not consumer configs). Commonly used to print keys/offsets/timestamps. Can be repeated multiple times.
`--key-deserializer <class>`	Custom key deserializer class (Java). Required when keys are not simple strings/bytes or if you want special handling.
`--value-deserializer <class>`	Custom value deserializer class (Java). Useful for Avro/JSON/protobuf custom decoding in CLI contexts (when configured appropriately).
`--skip-message-on-error`	If an error occurs while processing a record, skip it instead of halting the consumer. Helpful when a few malformed records exist.
`--whitelist <regex>`	Consume from topics matching a whitelist regex. Used when you want to subscribe to multiple topics dynamically by pattern.

Continued on next page

Flag	Explanation / When to use
`--blacklist <regex>`	Exclude topics matching a blacklist regex (often used together with `--whitelist` patterns, depending on version behavior).
`--csv-reporter-enabled`	Enable CSV metrics reporter for the consumer tool (primarily diagnostics/testing).
`--metrics-dir <path>`	If CSV reporter is enabled, write CSV metrics output into this directory.
`--enable-systest-events`	Log lifecycle/system test events in addition to consumed records (mainly used for system tests).
`--new-consumer`	Use the modern consumer implementation (this is the default in most versions).
`--zookeeper <hosts>`	Only required for the legacy/old consumer mode in older Kafka versions. Modern Kafka consumers use `--bootstrap-server`.
`--delete-consumer-offsets`	Legacy/older behavior: delete the consumer offsets path in ZooKeeper when starting (only relevant for old consumer scenarios).

4 Formatter (`--property`) keys you will commonly use

The `--property` flag configures how output is printed (formatter properties), not how the consumer connects.

Table 2: Common formatter properties passed via `--property key=value`.

Property	Effect
`print.key=true`	Print the record key (if present).
`print.value=true`	Print the record value (usually true by default depending on formatter).
`print.partition=true`	Print partition number for each record.
`print.offset=true`	Print offset for each record.
`print.timestamp=true`	Print record timestamp (CreateTime/LogAppendTime).
`key.separator=`	Separator between key and value in output (example: comma).
`line.separator=\n`	Separator between messages/lines (default newline).

Example: print key, partition, offset, timestamp

```
bin/kafka-console-consumer.sh \
  --bootstrap-server localhost:9092 \
  --topic iot_sensor_data \
  --from-beginning \
  --property print.key=true \
  --property print.partition=true \
  --property print.offset=true \
  --property print.timestamp=true
```

5 Notes and common pitfalls

- **Spelling matters**: it is `--from-beginning` (two "n" in *beginning*).
- If you're **inside the Kafka Docker container**, `localhost:9092` typically refers to the broker inside that container.
- If you run from the **host machine** (outside Docker), `localhost` refers to your host, not the container. In that case use the container port mapping or service hostname.
- For secure clusters (SASL/SSL), put security configs in `--consumer.config` (recommended) or via repeated `--consumer-property`.

6 References (optional)

Apache Kafka Quickstart demonstrates console consumption: kafka.apache.org/quickstart/