KIP-527: Add VoidSerde to Serdes

- Status
- Motivation
- Public Interfaces
- Proposed Changes
- Compatibility, Deprecation, and Migration Plan
- Rejected Alternatives

Status

Current state: *Adopted*

Discussion thread: *here*

JIRA: *KAFKA-8455*

Motivation

Often, when reading an input topic, the key is expected to be null, but there's actually no way to represent this fact in Consumed, leading to confusing type signatures down the topology. For example, you might use the BytesSerde, but then you have a KStream<Bytes,...>. When maintaining a large application, this becomes a hazard, since you'd need to "be really careful" not to try and dereference the key at any point, since you actually know it's always null. One might use the BytesSerde, but then you have a KStream<Bytes,...>. When maintaining a large application, this becomes a hazard, since you'd need to "be really careful" not to try and dereference the key at any point, since you actually know it's always null.

Public Interfaces

I propose to add the following new SerDe:

NothingSerde

```
static public final class VoidSerde extends WrapperSerde<Void> {
}

public class VoidSerializer implements Serializer<Void> {
}

public class VoidDeserializer implements Deserializer<Void> {
}
```

If not null parameters passed then an `java.lang.IllegalArgumentException` will be thrown.

Proposed Changes

I want to add `VoidSerde` to `Serdes` class.

Compatibility, Deprecation, and Migration Plan

There are no compatibility issues here.

Rejected Alternatives

`NullSerdes`, `NothingSerdes` naming was rejected during KIP discussion.