fn match_datetime_component

Michael Shoemate

> This proof resides in **"contrib"** because it has not completed the vetting process.

Proves soundness of `match_datetime_component` in `mod.rs` at commit f5bb719 (outdated[1]).

`match_datetime_component` returns the data type and number of possible unique values of a compute operation that retrieves a datetime component.

1 Hoare Triple

Precondition

Compiler-verified

- Argument `temporal_function` of type `TemporalFunction`

Caller-verified

None

Pseudocode

```
1  def match_datetime_component(
2      temporal_function: TemporalFunction,
3  ) -> DataType | None:
4      return ({
5          Millennium: DataType.UInt32,
6          Century: DataType.Int32,
7          Year: DataType.Int32,
8          IsoYear: DataType.Int32,
9          Quarter: DataType.Int8,
10         Month: DataType.Int8,
11         Week: DataType.Int8,
12         WeekDay: DataType.Int8,
13         Day: DataType.Int8,
14         OrdinalDay: DataType.Int16,
15         Hour: DataType.Int8,
16         Minute: DataType.Int8,
17         Second: DataType.Int8,
18         Millisecond: DataType.Int32,
19         Microsecond: DataType.Int32,
20         Nanosecond: DataType.Int32,
21     }).get(temporal_function)
```

---

[1]See new changes with `git diff f5bb719..e4b3962 rust/src/transformations/make_stable_expr/namespace_dt/expr_datetime_component/mod.rs`

Postcondition

Theorem 1.1. For any choice of `temporal_function`, returns the data type of the output if the input is an infallible row-by-row temporal expression, otherwise returns none.

Proof. For each choice of `temporal_function`, the data type is directly verified via testing. Infallible row-by-row is verified by inspection of the Polars source code. □