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Proves soundness of `match_datetime_component` in `mod.rs` at commit f5bb719 (outdated[1]).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

```
def match_datetime_component(
    temporal_function: TemporalFunction,
) -> DataType | None:
    return ({
        Millennium: DataType.UInt32,
        Century: DataType.Int32,
        Year: DataType.Int32,
        IsoYear: DataType.Int32,
        Quarter: DataType.Int8,
        Month: DataType.Int8,
        Week: DataType.Int8,
        WeekDay: DataType.Int8,
        Day: DataType.Int8,
        OrdinalDay: DataType.Int16,
        Hour: DataType.Int8,
        Minute: DataType.Int8,
        Second: DataType.Int8,
        Millisecond: DataType.Int32,
        Microsecond: DataType.Int32,
        Nanosecond: DataType.Int32,
    }).get(temporal_function)
```

[1] See new changes with `git diff f5bb719..db56d1a 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. □