

# West separate logical expressions

```
bool operator==(TaggedAmountClass const& other) const {
    auto result =      this->date() == other.date()
        and this->cents_amount() == other.cents_amount()
        and std::all_of(
            m_tags.begin()
            ,m_tags.end()
            , [&other](Tags::value_type const& entry) {
                return (
                    (entry.first.starts_with("_"))
                    or (
                        other.tags().contains(entry.first)
                        and other.tags().at(entry.first) == entry.second));
            });
    return result;
}
```

# West separate new-line in list print

```
Amount income = get_INK1_Income(model);
```

```
prompt << "\n1: INK1 1.1 Lön, förmåner, sjukpenning m.m. = " << income;
```

```
Amount dividend = get_K10_Dividend(model);
```

```
prompt << "\n2: K10 1.6 Utdelning = " << dividend;
```

```
prompt << "\n3: Continue (Create K10 and INK1)";
```

```
model->prompt_state = PromptState::K10INK1EditOptions;
```