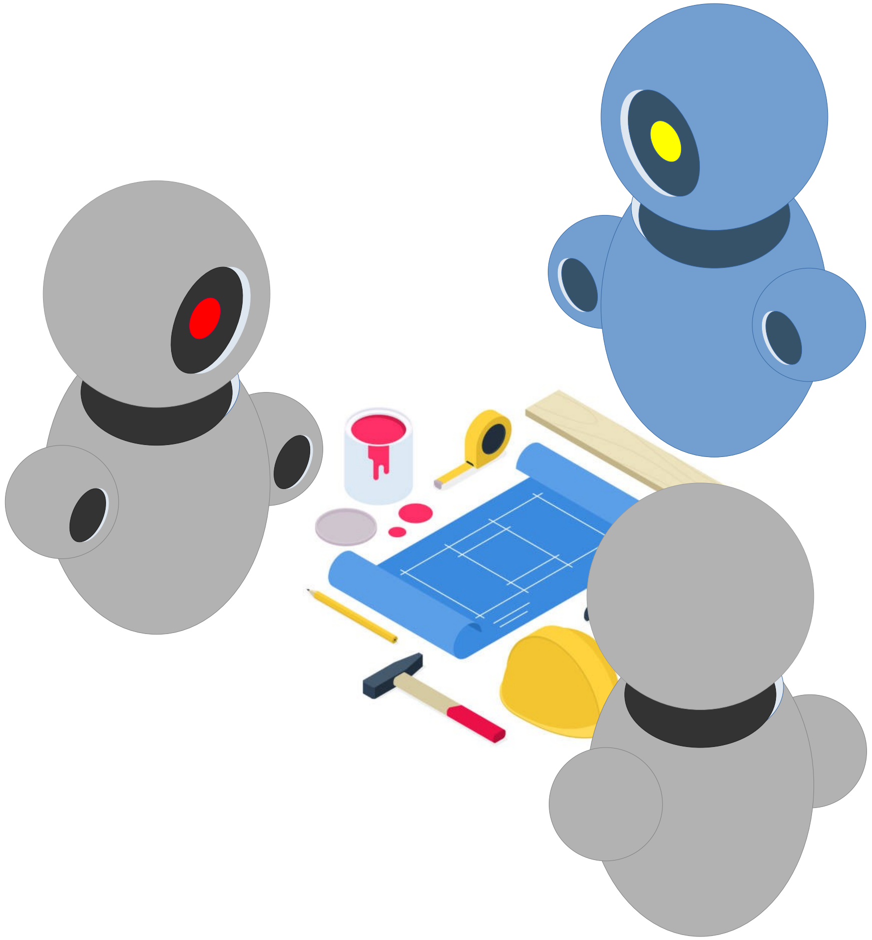
Introduction to Distributed and Embedded Multi-agent Systems

Carlos Eduardo Pantoja[1]
Nilson Mori Lazarin[1,2]

1. Centro Federal de Educação Tecnológica (CEFET/RJ) - 2. Universidade Federal Fluminense (UFF), Brasil

Março, 2024

GOALS AND PLANS



Jason Framework: Goals

Em Jason, os **goals** (objetivos) representam os estados do mundo em que o agente deseja atingir.

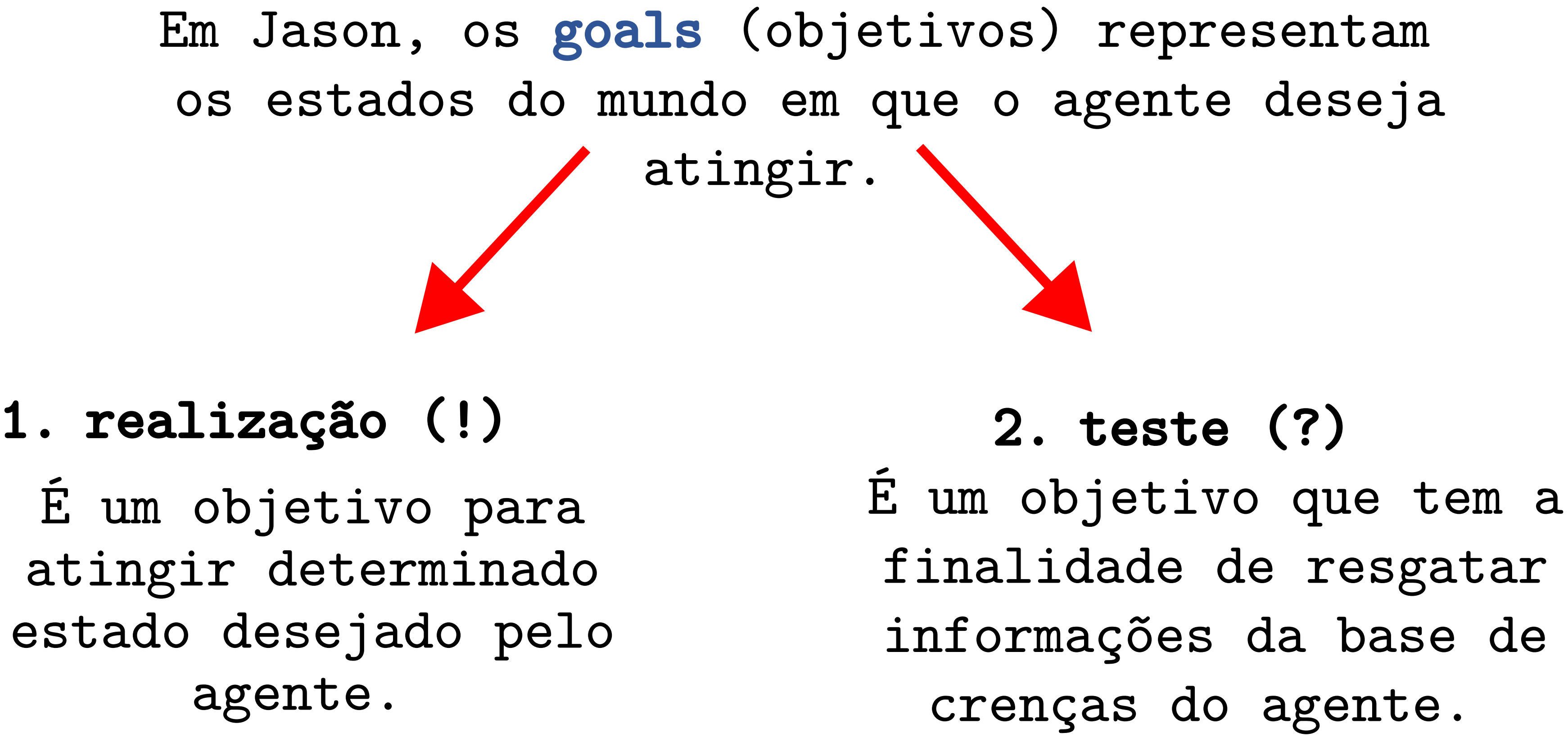
Jason Framework: Goals

Em Jason, os **goals** (objetivos) representam os estados do mundo em que o agente deseja atingir.

1. **realização (!)**

É um objetivo para atingir determinado estado desejado pelo agente.

Jason Framework: Goals



Em Jason, os **goals** (objetivos) representam os estados do mundo em que o agente deseja atingir.

1. realização (!)

É um objetivo para atingir determinado estado desejado pelo agente.

2. teste (?)

É um objetivo que tem a finalidade de resgatar informações da base de crenças do agente.

Goals: Format

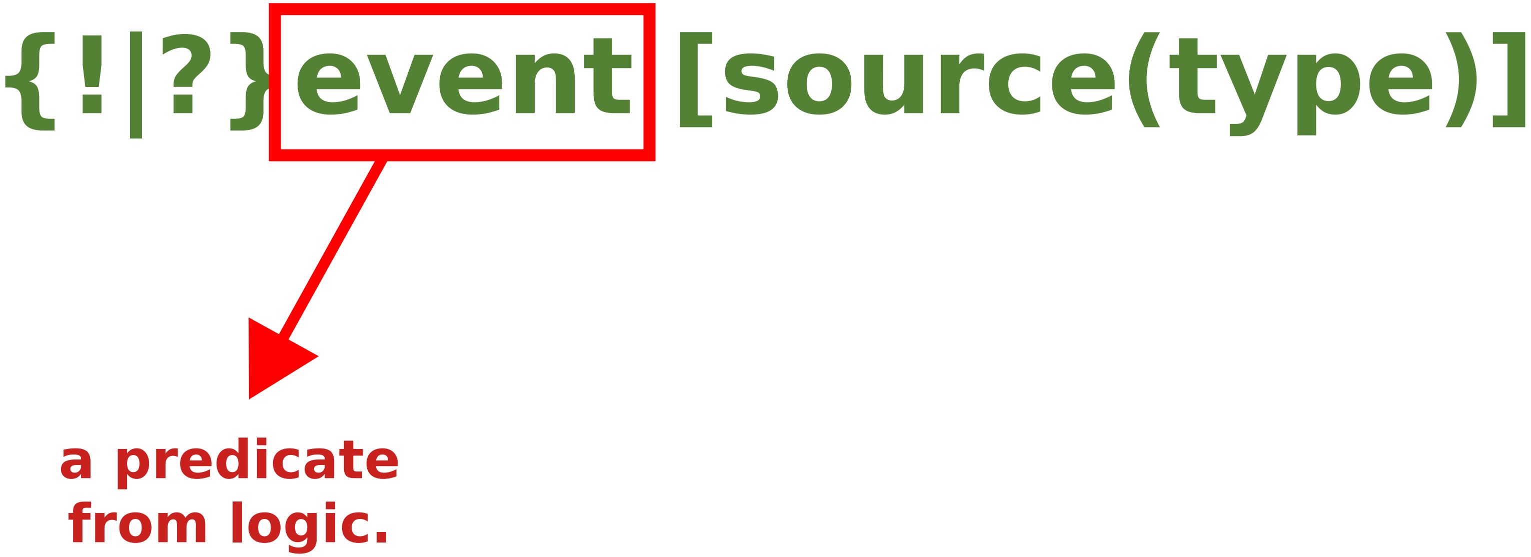
{!|?}event [source(type)]

Goals: Format

{!|?}event [source(type)]

the goal type

Goals: Format



{!|?}event [source(type)]

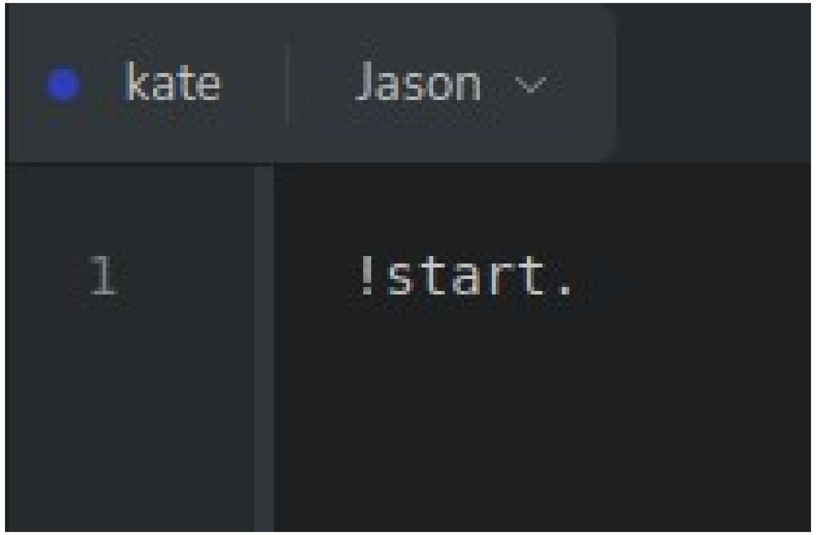
a predicate from logic.

Goals: Format

{!|?}event [source(type)]

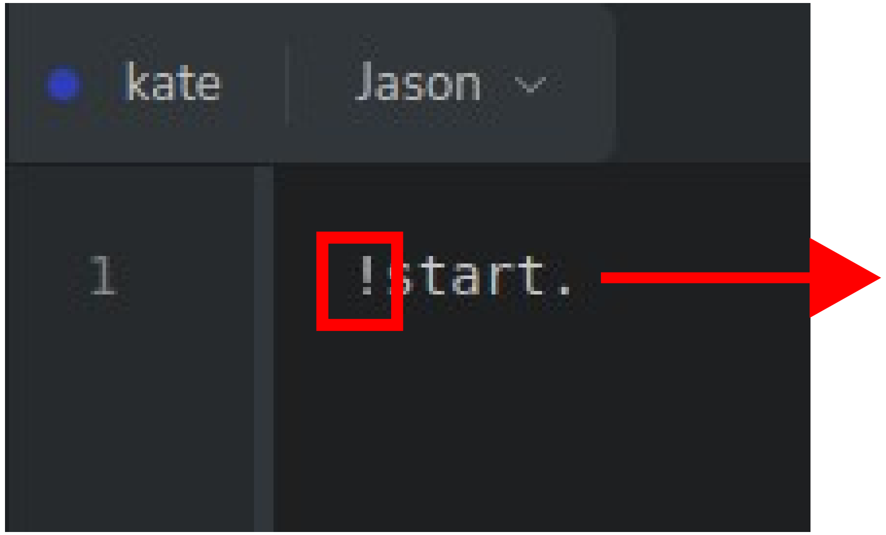
the source of the belief.

Goals: Initial Goal



```
!start.
```

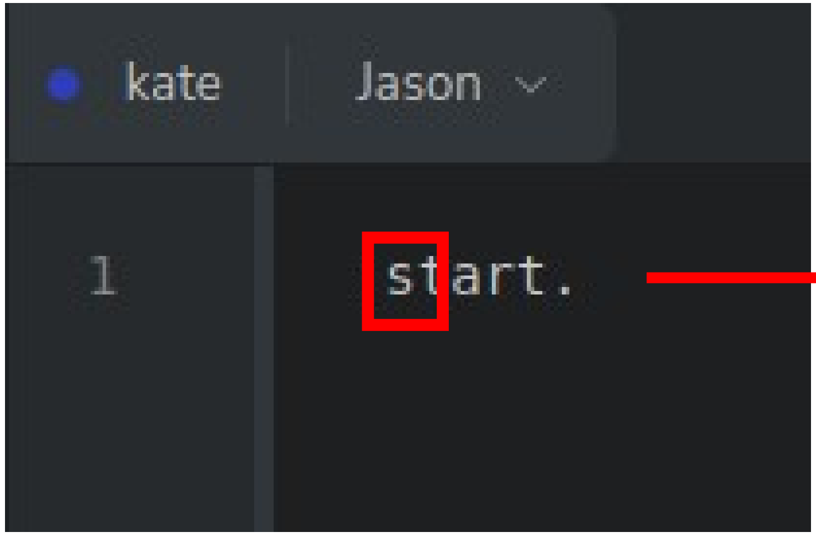
Goals: Initial Goal



```
!start.
```

every goal must start with an **exclamation point**.

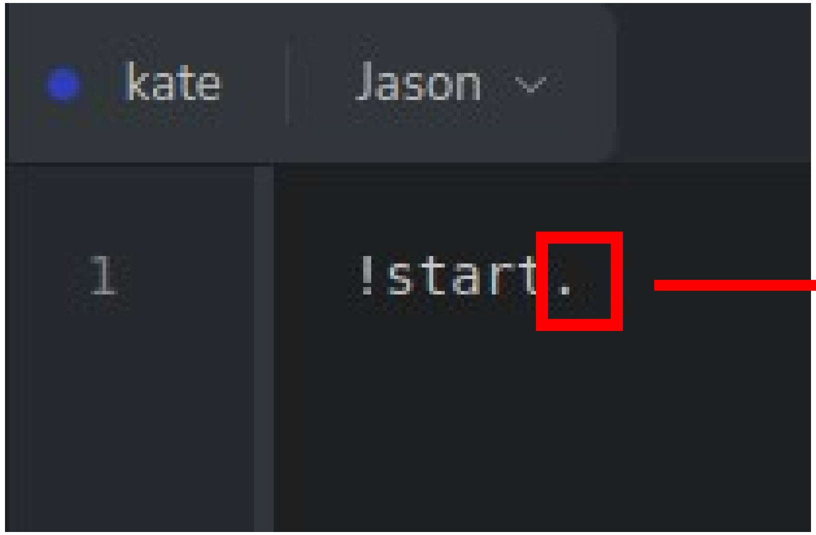
Goals: Initial Goal



```
start.
```

every goal's predicate must start with a **<u>lowercase</u>** letter.

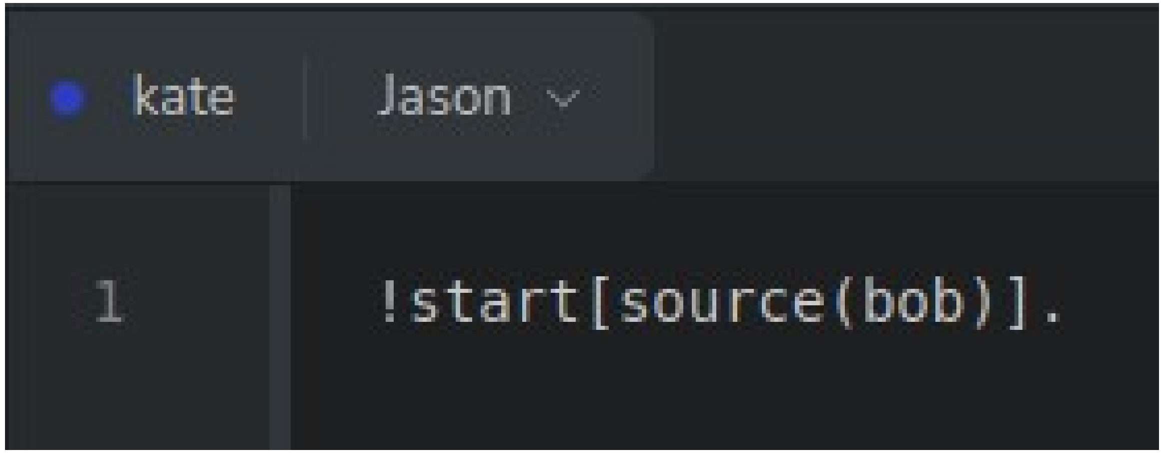
Goals: Initial Goal



```
!start.
```

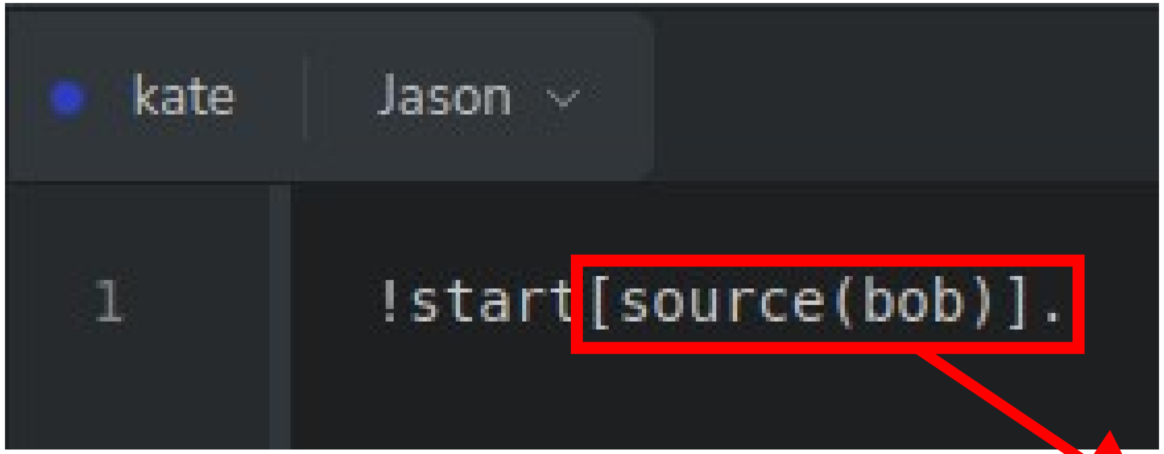
and it must end with a **period**.

Goals: Initial Goal with a Source



```
!start[source(bob)].
```

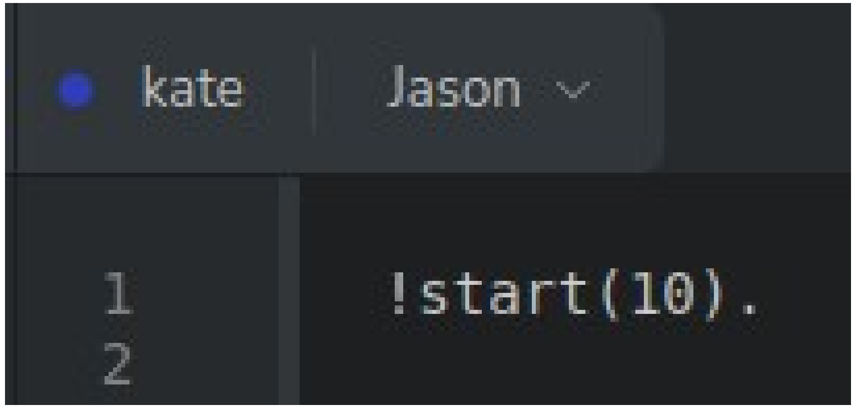
Goals: Initial Goal with a Source



```
!start[source(bob)].
```

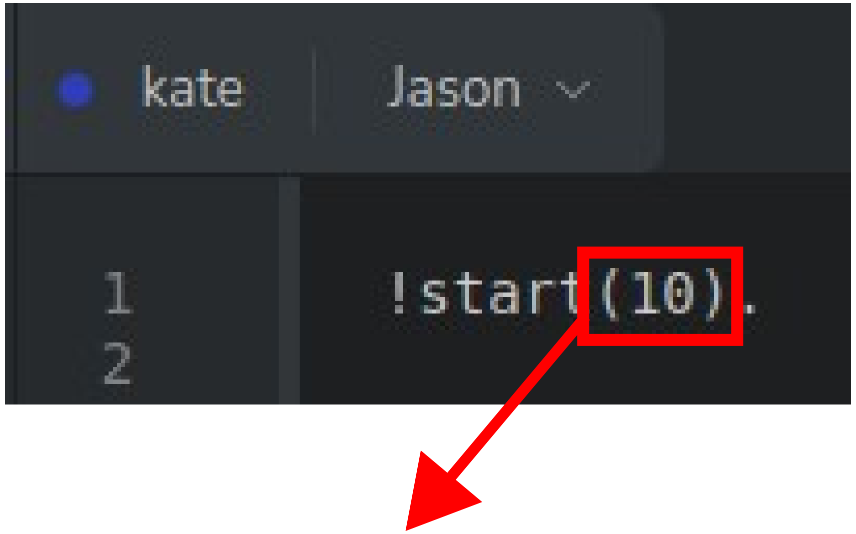
one can define a **source**.

Goals: Initial Goal with Predicate



```
!start(10).
```

Goals: Initial Goal with Predicate



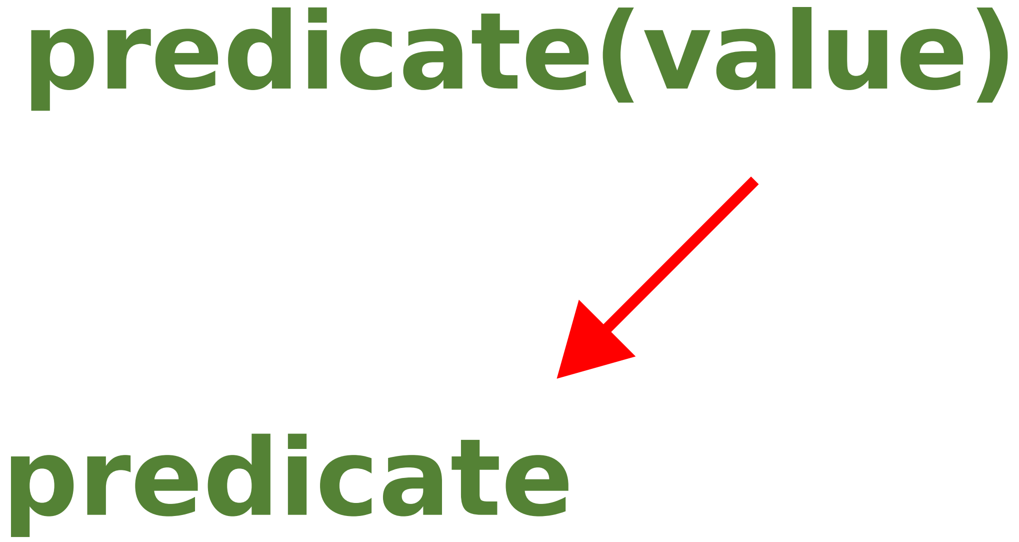
```
!start(10).

```

predicate(int)

Goals: Initial Goal with Predicate

predicate(value)

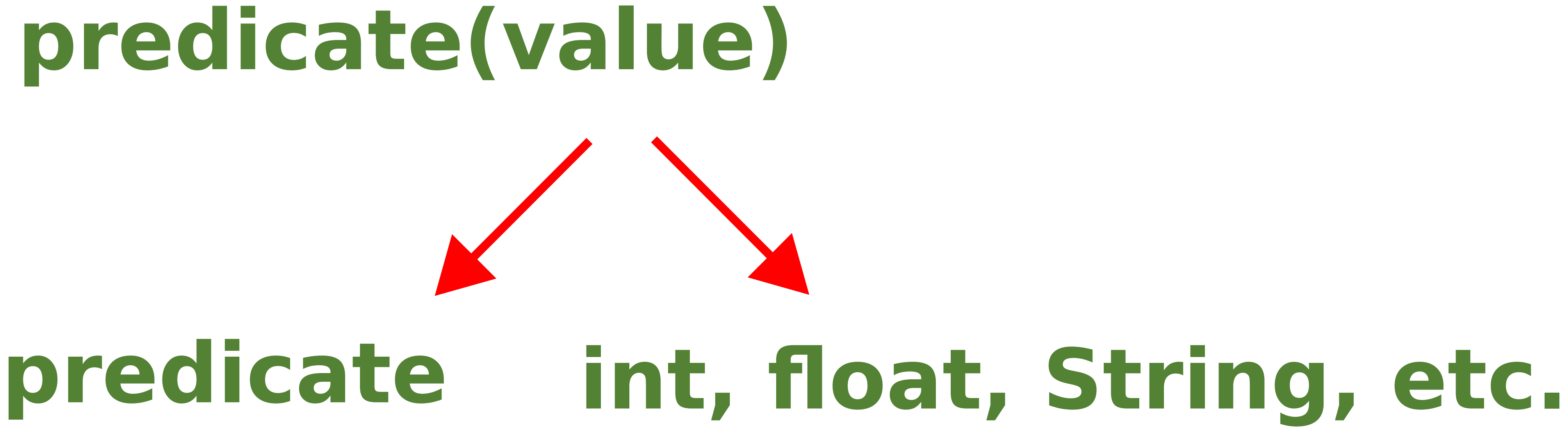
Goals: Initial Goal with Predicate



predicate(value)

predicate

Goals: Initial Goal with Predicate



predicate(value)

predicate

int, float, String, etc.

Goals: Format

predicate(predicate)

Goals: Format

predicate(predicate)

Goals: Format

predicate(predicate(predicate))

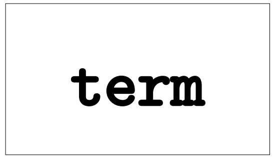
Goals: Format

predicate(predicate(predicate))

Goals: Format

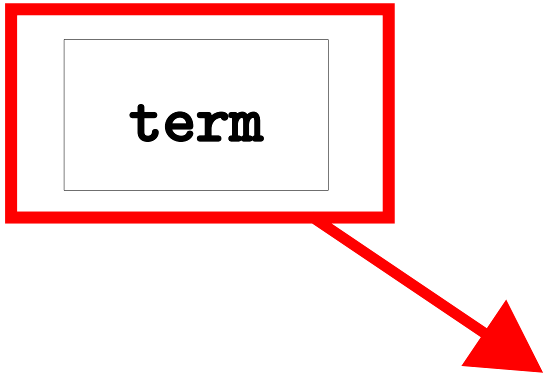
predicate(predicate(predicate(...)))

Logic-Based Programming



term

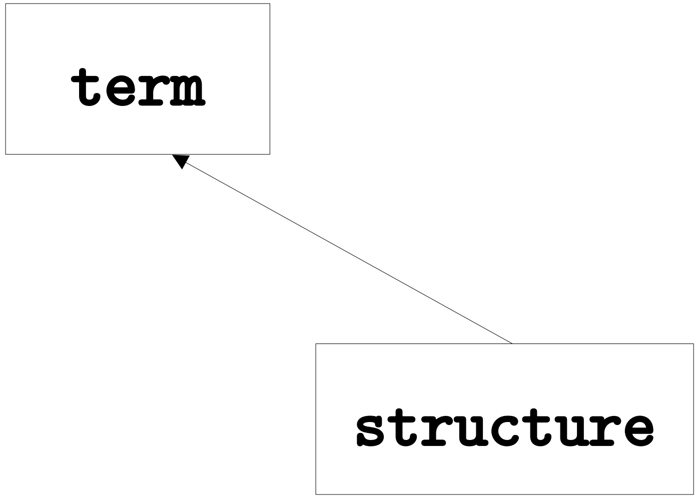
Logic-Based Programming



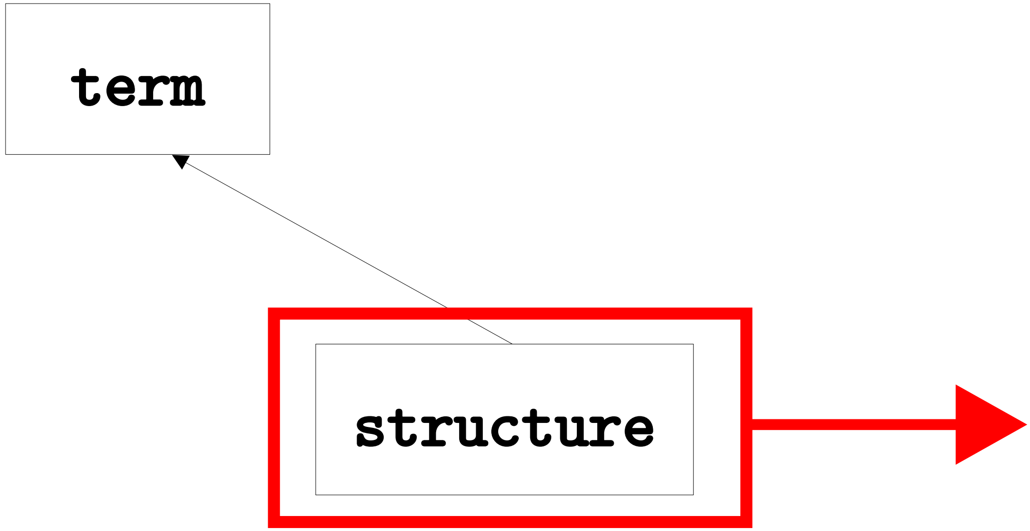
term

Every construction in Jason is a **term**.

Logic-Based Programming

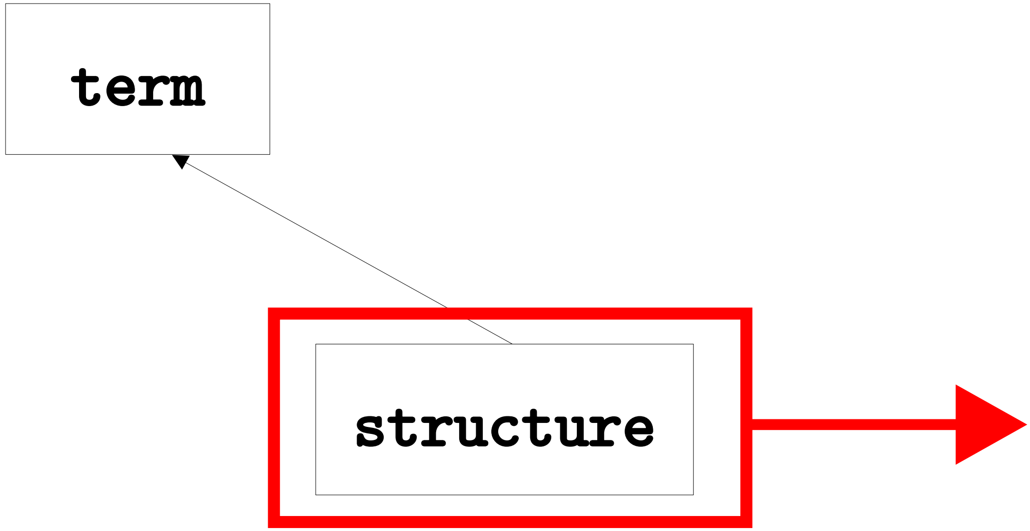


Logic-Based Programming



it represents complex data followed by a number of terms.

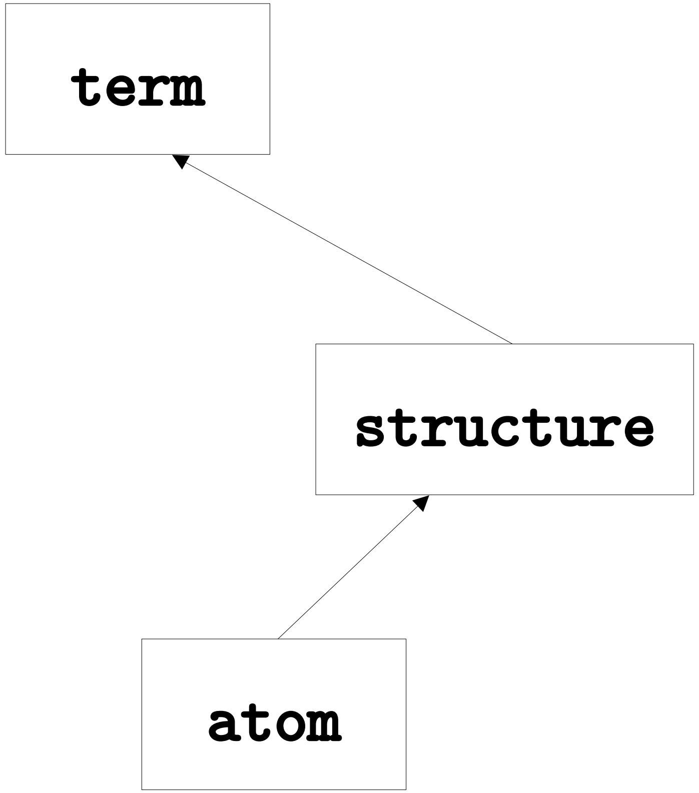
Logic-Based Programming



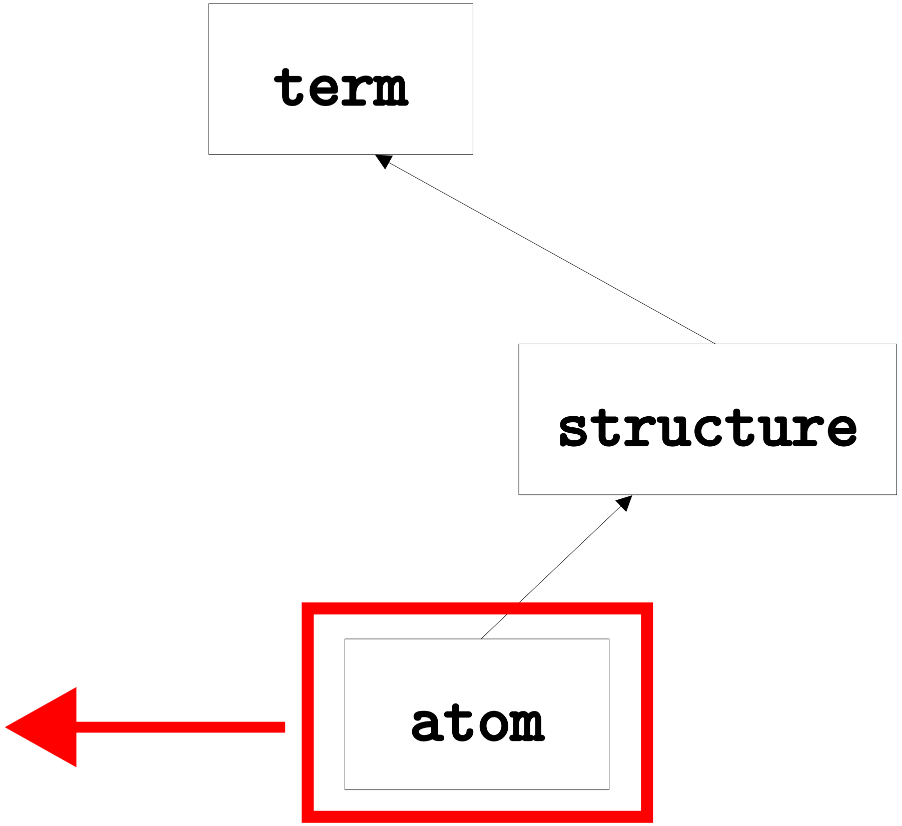
it represents complex data followed by a number of terms.

name(kate,18)

Logic-Based Programming



Logic-Based Programming

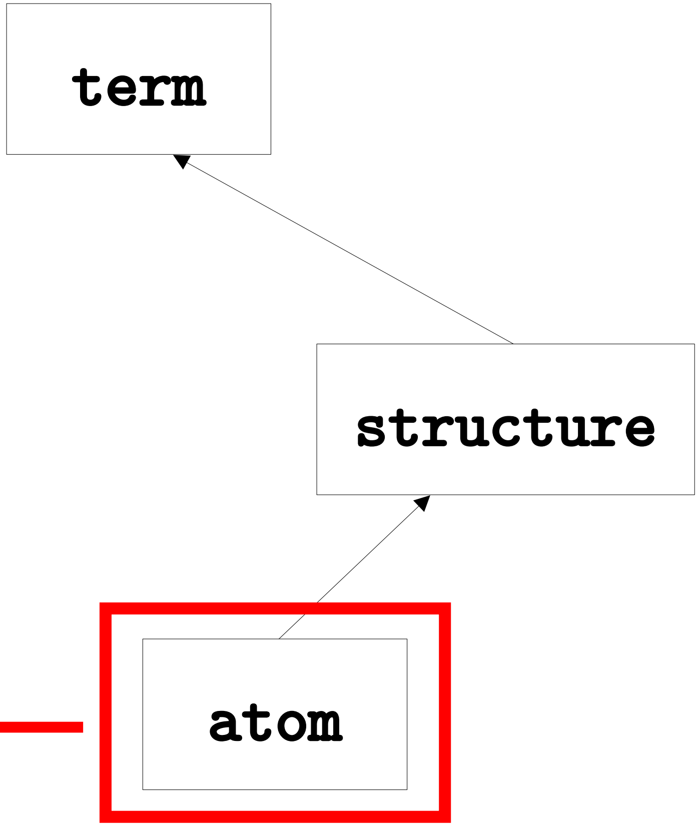


any symbol starting with a lowercase letter.

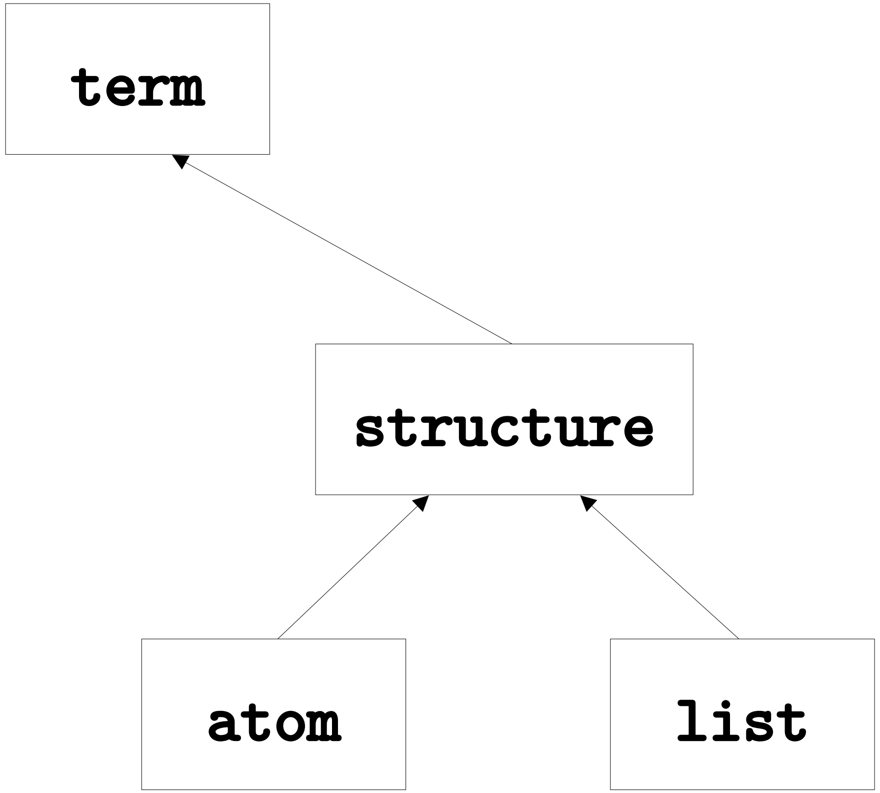
Logic-Based Programming

sunny

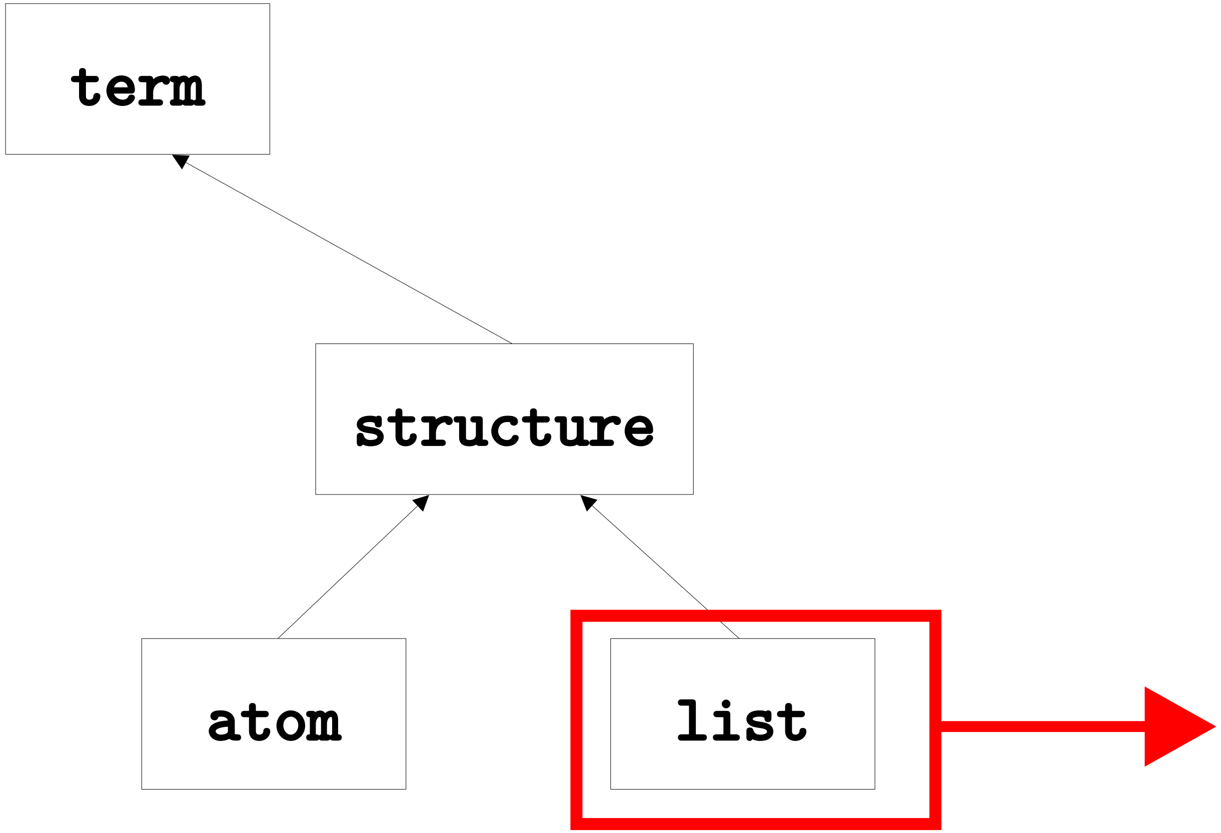
any symbol starting with a lowercase letter.



Logic-Based Programming

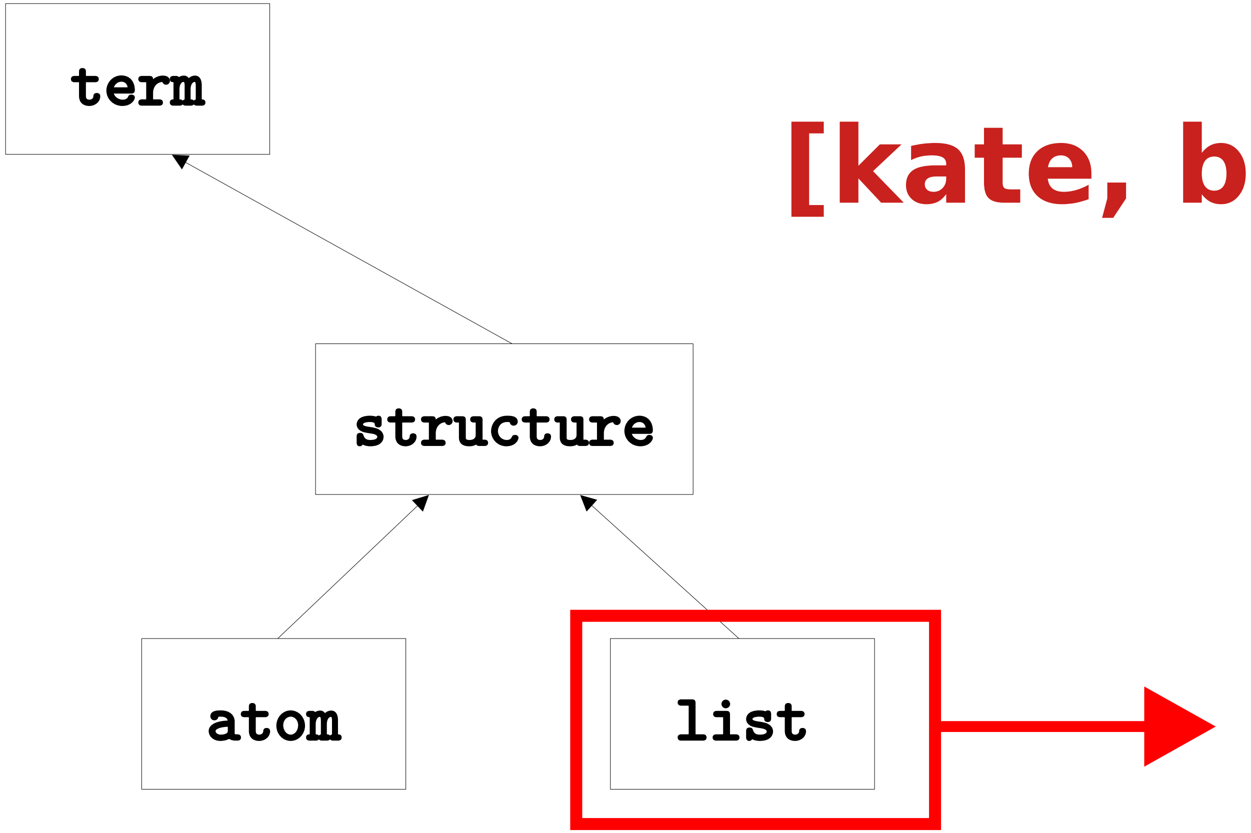


Logic-Based Programming



a list of atoms or values.

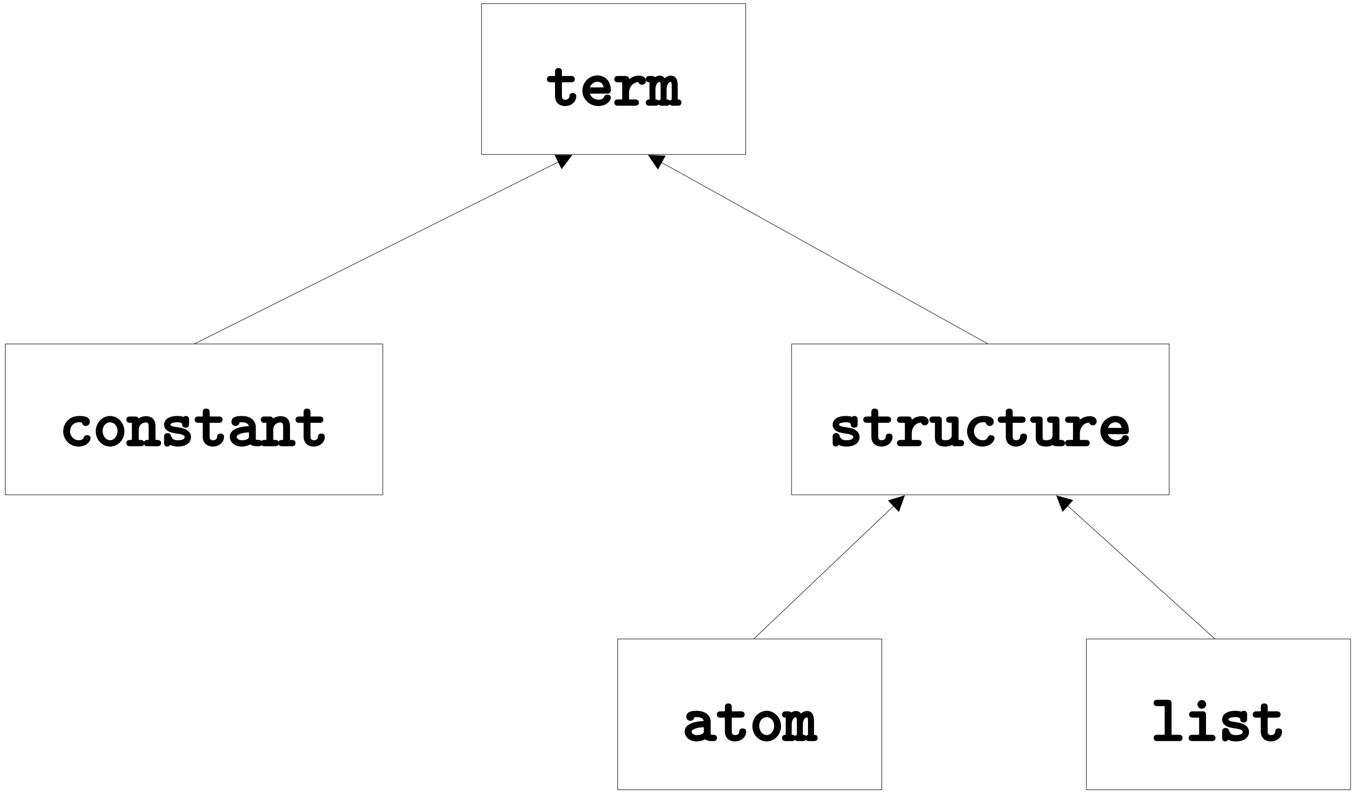
Logic-Based Programming



[kate, bob, alice]

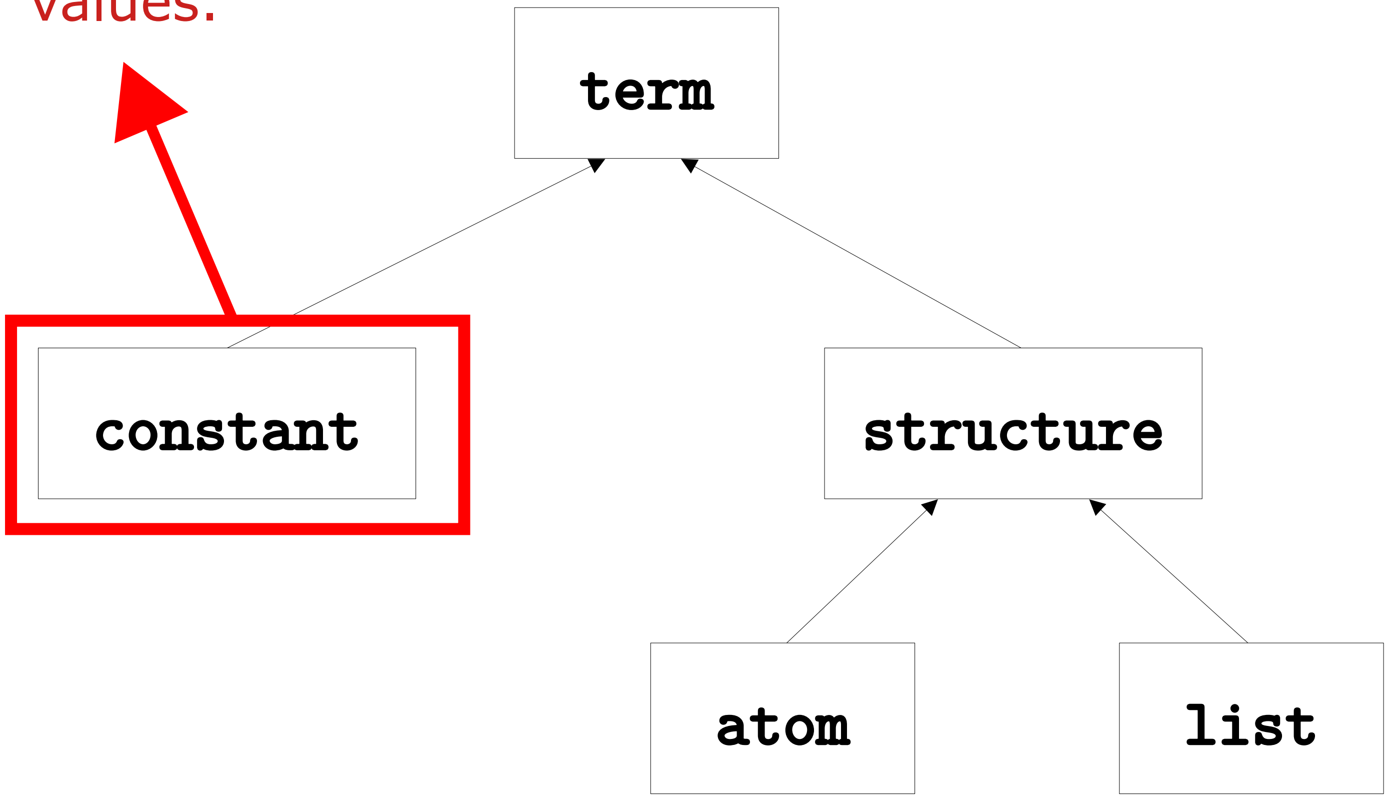
a list of atoms or values.

Logic-Based Programming

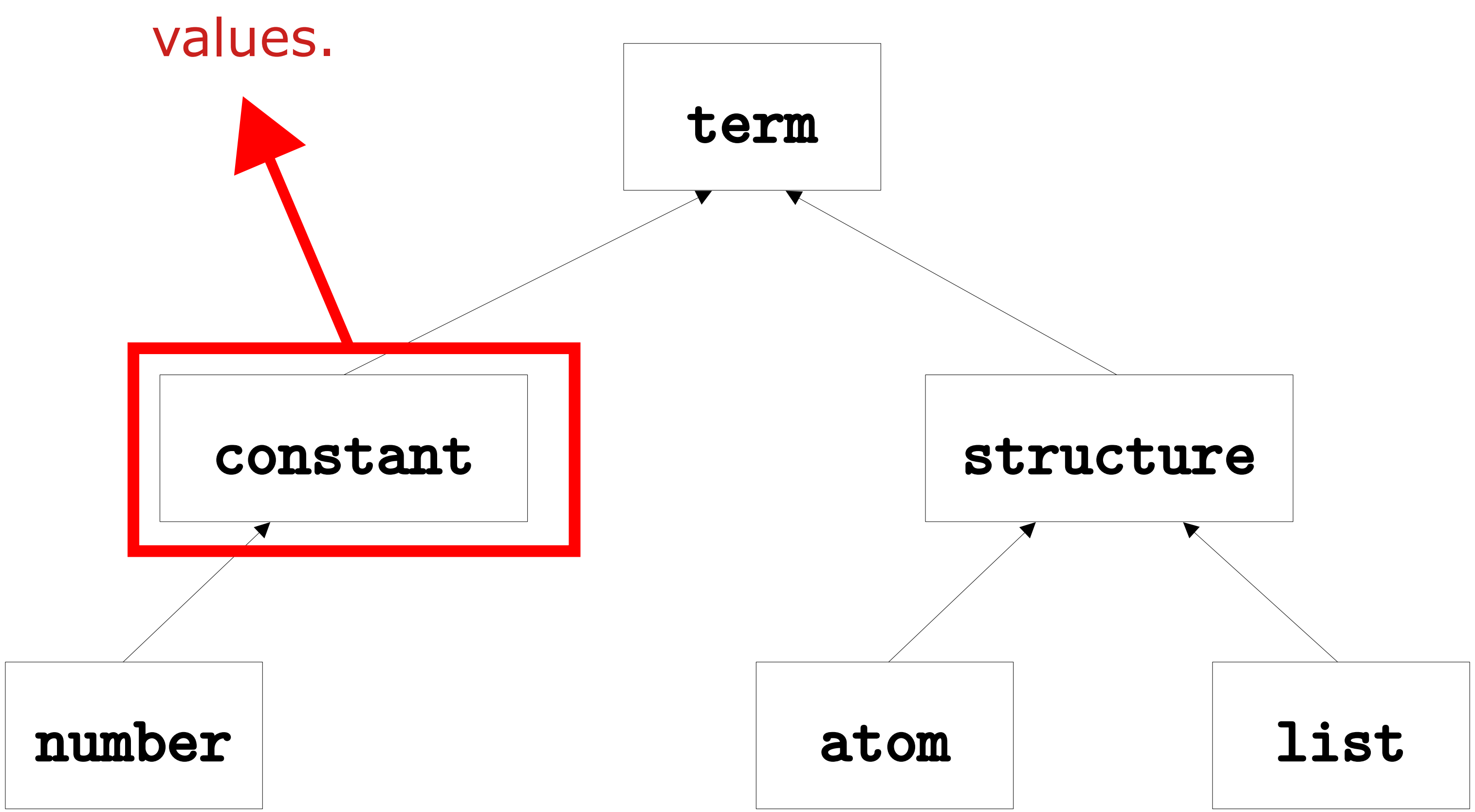


Logic-Based Programming

values.

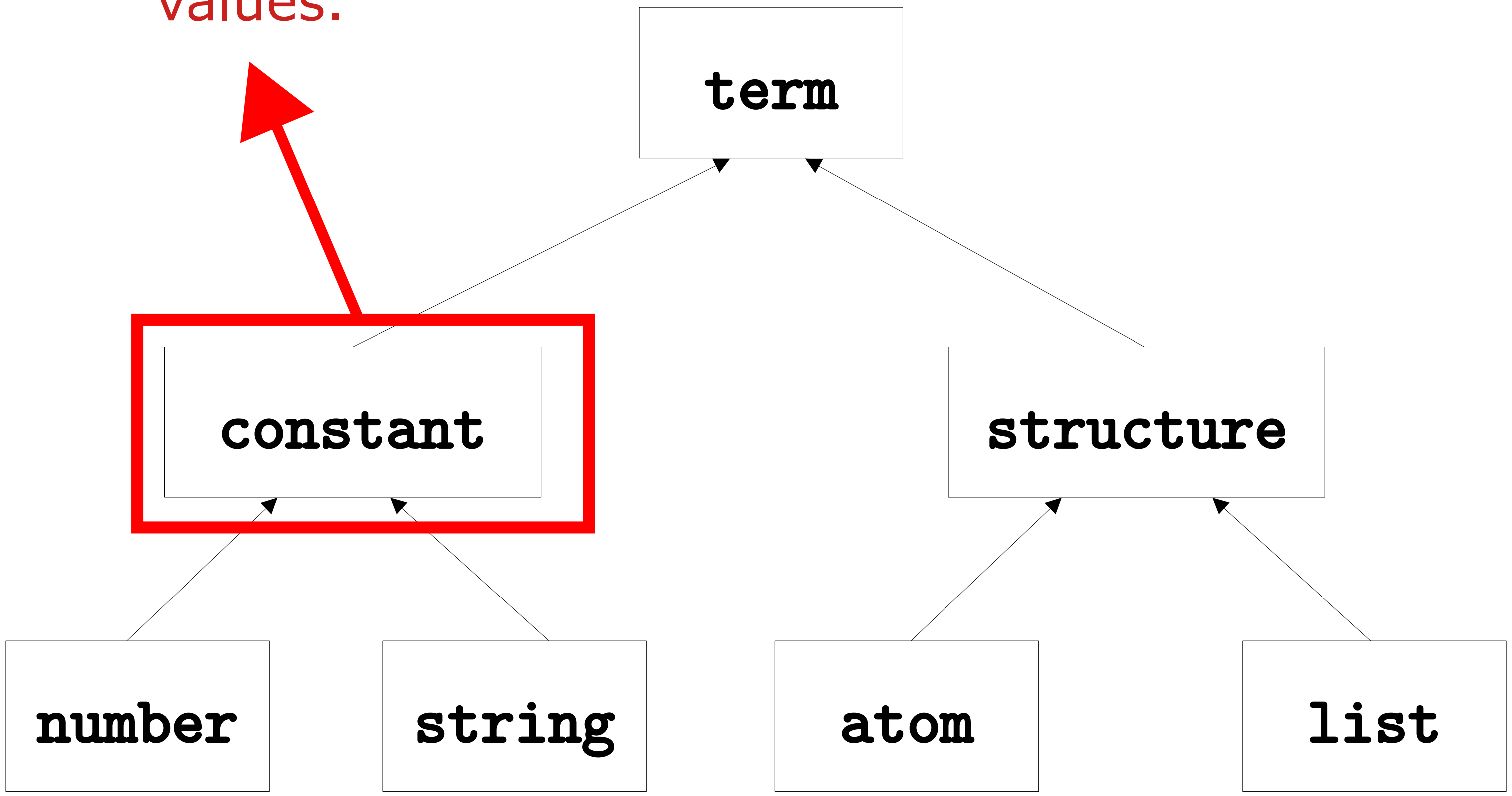


Logic-Based Programming

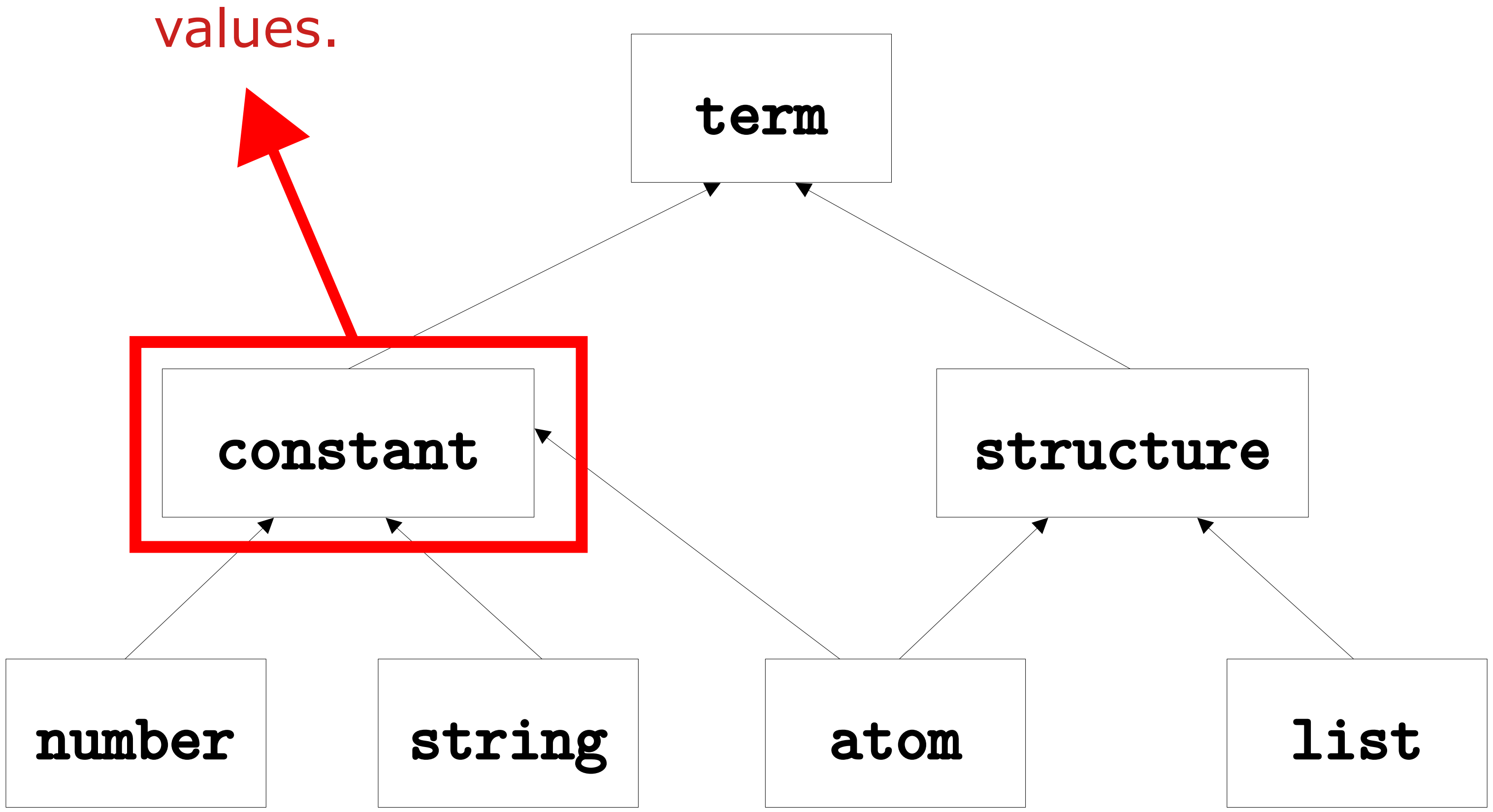


Logic-Based Programming

values.

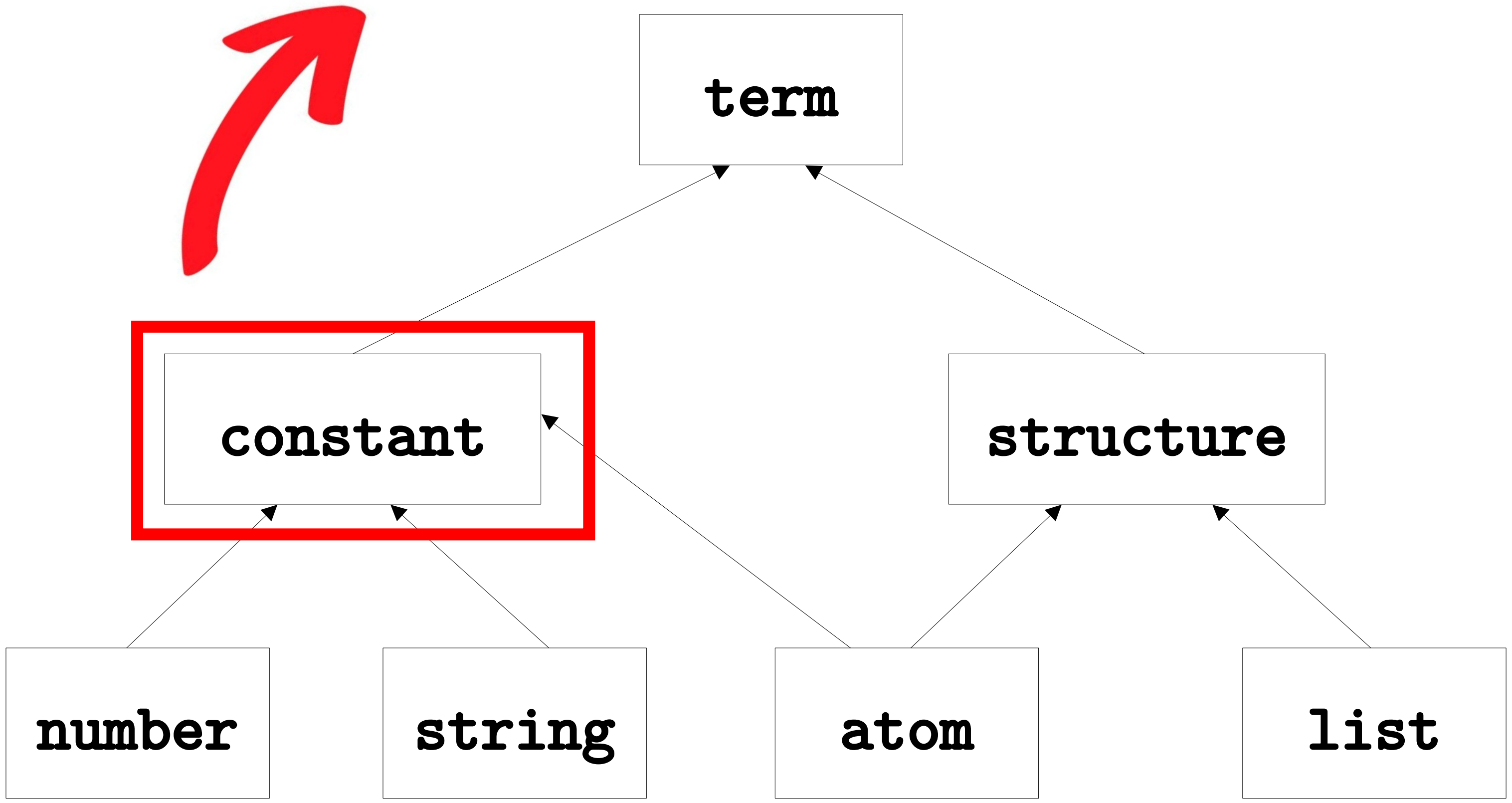


Logic-Based Programming



Logic-Based Programming

employer(kate, 3582, "kate@kate.com", full(40))



Jason Framework: Plans Format

triggering_event : context <- body.

Jason Framework: Plans Format

triggering_event : context <- body.

1. **triggering Event**
   - Um agente pode ter diversos objetivos. Os planos são ativados baseados nos eventos que podem ser ativados em determinado momento.

Jason Framework: Plans Format

triggering_event : context <- body.

1. **triggering Event**
   - Um agente pode ter diversos objetivos. Os planos são ativados baseados nos eventos que podem ser ativados em determinado momento.

2. **context**
   - São as condições para a ativação de um plano dentro vários eventos.

Jason Framework: Plans Format

triggering_event : context <- body.

1. **triggering Event**
   - Um agente pode ter diversos objetivos. Os planos são ativados baseados nos eventos que podem ser ativados em determinado momento.

2. **context**
   - São as condições para a ativação de um plano dentro vários eventos.

3. **body**.
   - É o corpo do plano. Uma sequência de ações a ser executada pelo agente.

Plan: Format

```
{+|-}{!|?}event [source(type)]:
    context ←
        action 1;
        action 2;
        action n.
```

Plan: Format

```
{+|-}{!|?}event [source(type)]:
    context ←
        action 1;
        action 2;
        action n.
```

The change type.

Plan: Format

```
{+|-}{!|?}event [source(type)]:
    context ←
        action 1;
        action 2;
        action n.
```

The type of the goal.

Plan: Format

```
{+|-}{!|?}event [source(type)]:
    context ←
        action 1;
        action 2;
        action n.
```

The triggering event.

Plan: Format

```
{+|-}{!|?}event [source(type)]:
    context ←
        action 1;
        action 2;
        action n.
```

the source of the plan.

Plan: Format

```
{+|-}{!|?}event [source(type)]:
    context ←
        action 1;
        action 2;
        action n.
```

the beginning of
the conditions.

Plan: Format

```
{+|-}{!|?}event [source(type)]:
    context ←
        action 1;
        action 2;
        action n.
```

the beginning of the conditions.

It is optional if there are no condintions.

Plan: Format

```
{+|-}{!|?}event [source(type)]:
    context ←
        action 1;
        action 2;
        action n.
```

The activation conditions.

Plan: Format

```
{+|-}{!|?}event [source(type)]:
    context ←
        action 1;
        action 2;
        action n.
```

The activation conditions.

It is not mandatory.

Plan: Format

```
{+|-}{!|?}event [source(type)]:
     context ←
          action 1;
          action 2;
          action n.
```

the beginning of the body.

Plan: Format

```
{+|-}{!|?}event [source(type)]:
    context ←
        action 1;
        action 2;
        action n.
```

the actions of the body.

Plan: Format

```
{+|-}{!|?}event [source(type)]:
     context ←
       action 1;
       action 2;
       action n.
```

the actions of the body.

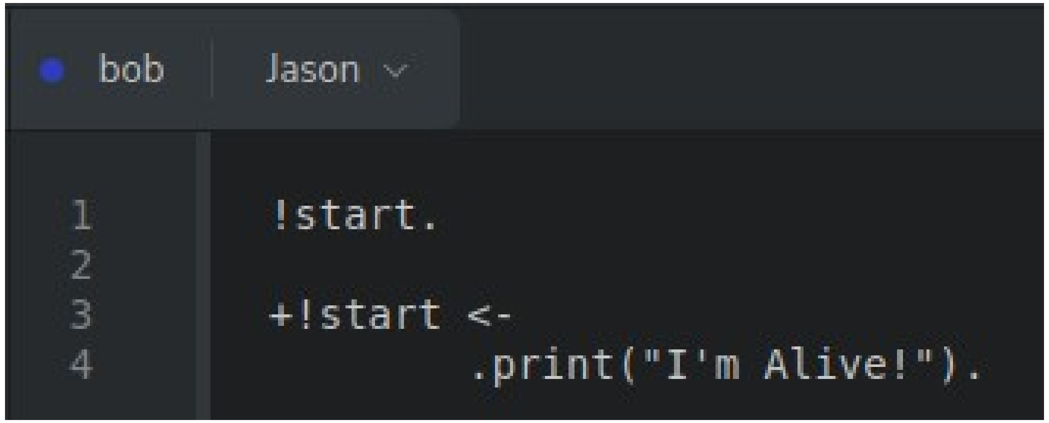
Separated by semi-colon.

Plan: Format

```
{+|-}{!|?}event [source(type)]:
     context ←
          action 1;
          action 2;
          action n.
```

the end of the plan.

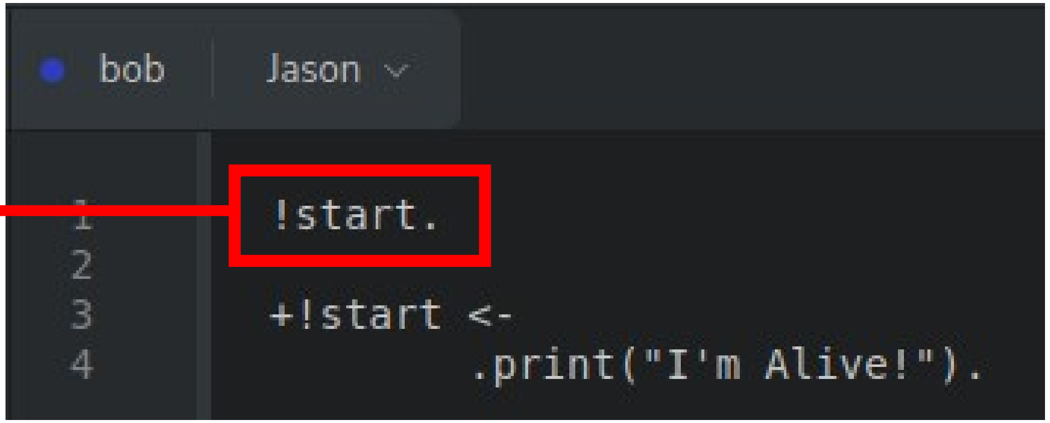
Plan: Hello World



```
!start.

+!start <-
        .print("I'm Alive!").
```

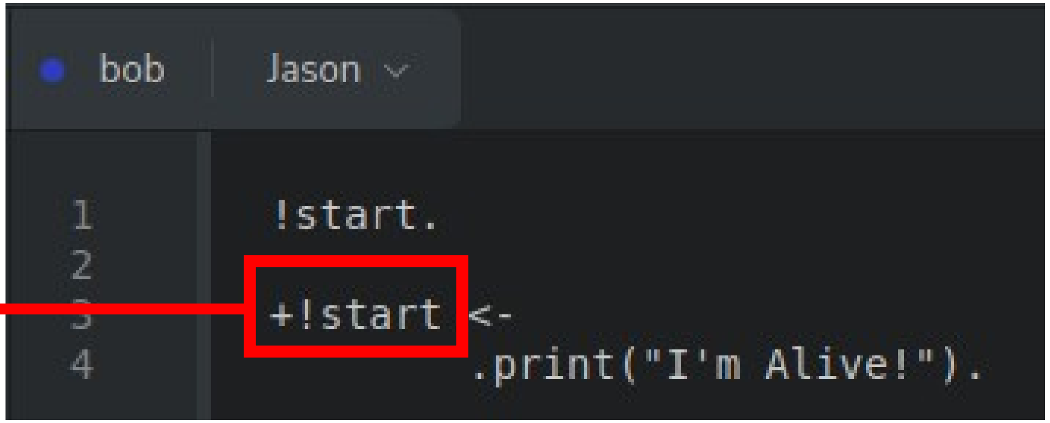
Plan: Hello World

An initial goal named start.



```
!start.

+!start <-
        .print("I'm Alive!").
```

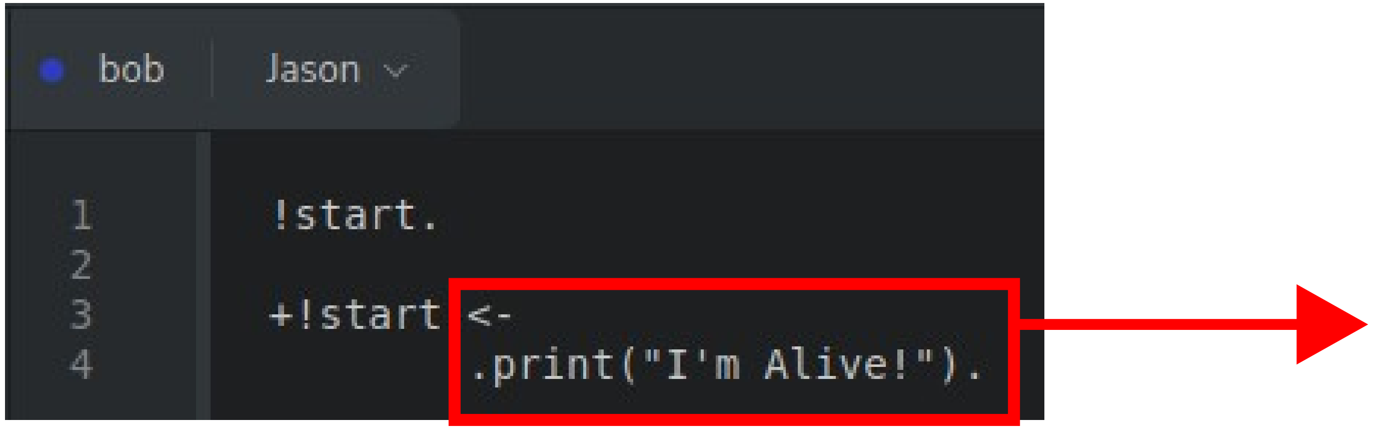
Plan: Hello World



```
!start.

+!start <-
        .print("I'm Alive!").
```

An achievement plan without triggering event named <u>start</u>.

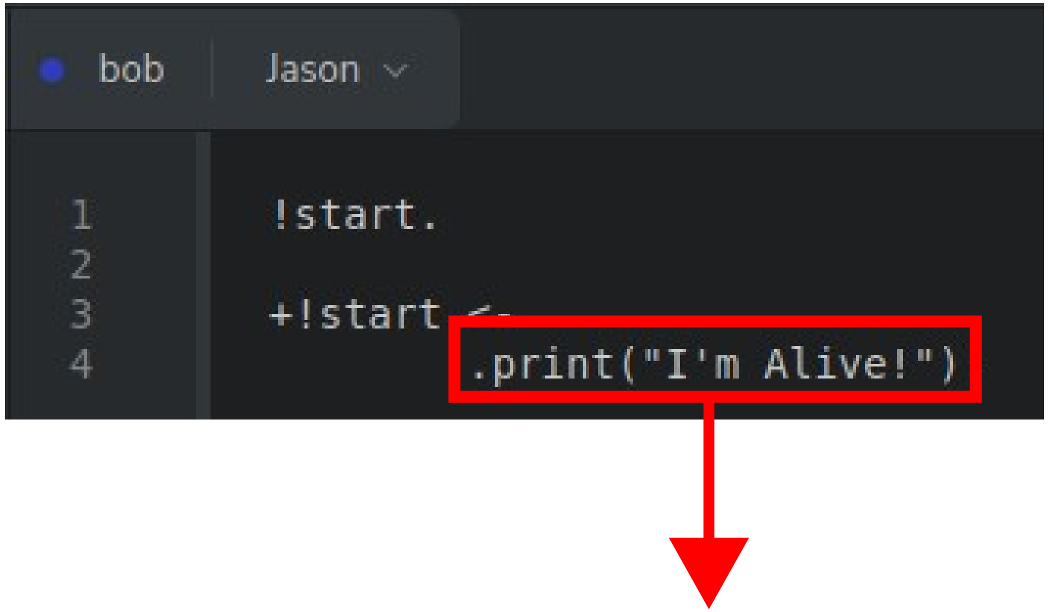
Plan: Hello World



```
!start.

+!start <-
        .print("I'm Alive!").
```

The body of the plan.

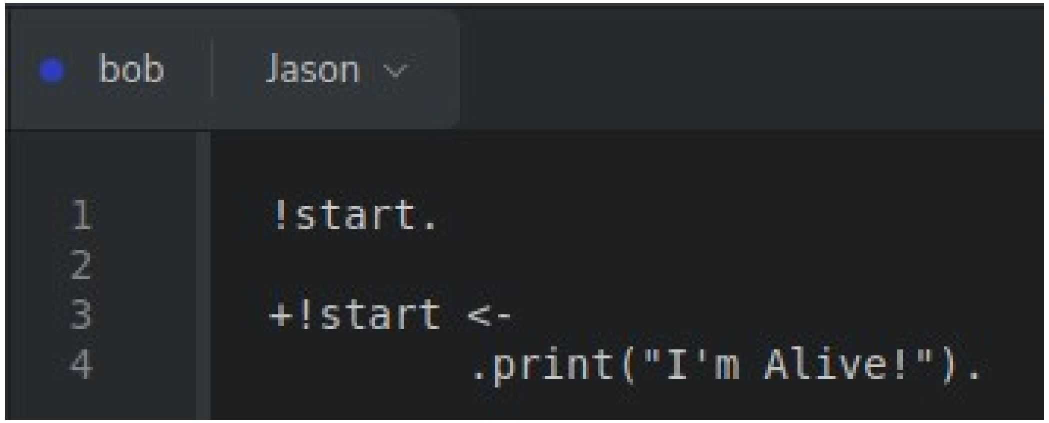
Plan: Hello World



```
!start.

+!start <-
        .print("I'm Alive!")
```

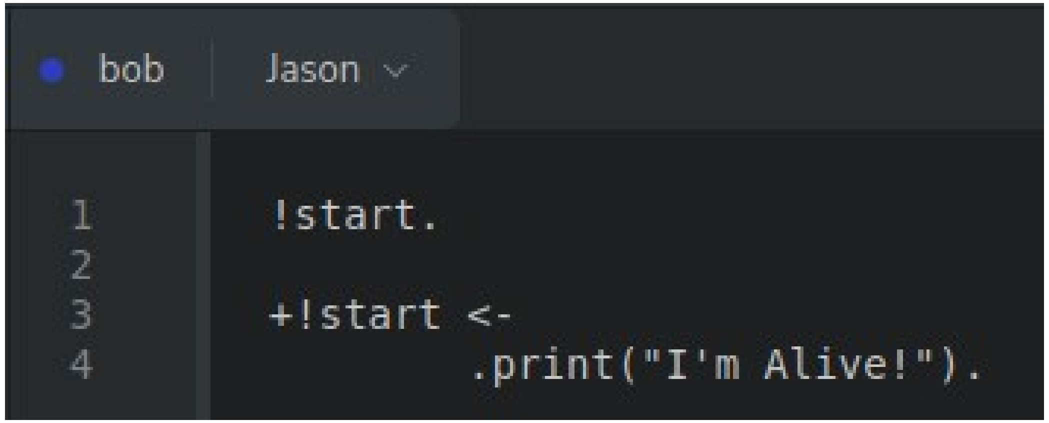
An action that prints in the agent tracer (console).

Plan: Hello World

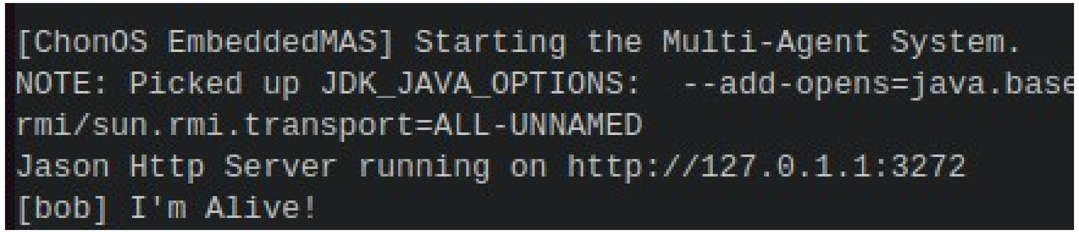


```
!start.

+!start <-
        .print("I'm Alive!").
```

Plan: Hello World



```
!start.

+!start <-
        .print("I'm Alive!").
```



```
[ChonOS EmbeddedMAS] Starting the Multi-Agent System.
NOTE: Picked up JDK_JAVA_OPTIONS:  --add-opens=java.base
rmi/sun.rmi.transport=ALL-UNNAMED
Jason Http Server running on http://127.0.1.1:3272
[bob] I'm Alive!
```

Plan: Types

1. **Achievement Goal**
   - São ativados quando um plano é transformado de um desejo para uma intenção na mente do agente.

Plan: Types

1. **Achievement Goal**
   - São ativados quando um plano é transformado de um desejo para uma intenção na mente do agente.

2. **Test Goal**
   - São objetivos que recuperam informações da base de crenças.

Plan: Types

1. **Achievement Goal**
   - São ativados quando um plano é transformado de um desejo para uma intenção na mente do agente.
2. **Test Goal**
   - São objetivos que recuperam informações da base de crenças.
3. **Belief Goal**
   - São planos ativados quando o agente adiciona ou remove uma crença da sua base de crenças.

Plans: Achievement Goal

```
{+|-}!event[source(type)]:
    context ←
        action 1;
        action 2;
        action n.
```

Plans: Achievement Goal

```
{+|-}!event[source(type)]:
    context ←
        action 1;
        action 2;
        action n.
```

defines if it is an addition (+) or a deletion (-) plan.

Plans: Achievement Goal

specifies the achievement plan

```
{+|-}!event[source(type)]:
     context ←
          action 1;
          action 2;
          action n.
```

Plans: Addition Achievement Goal

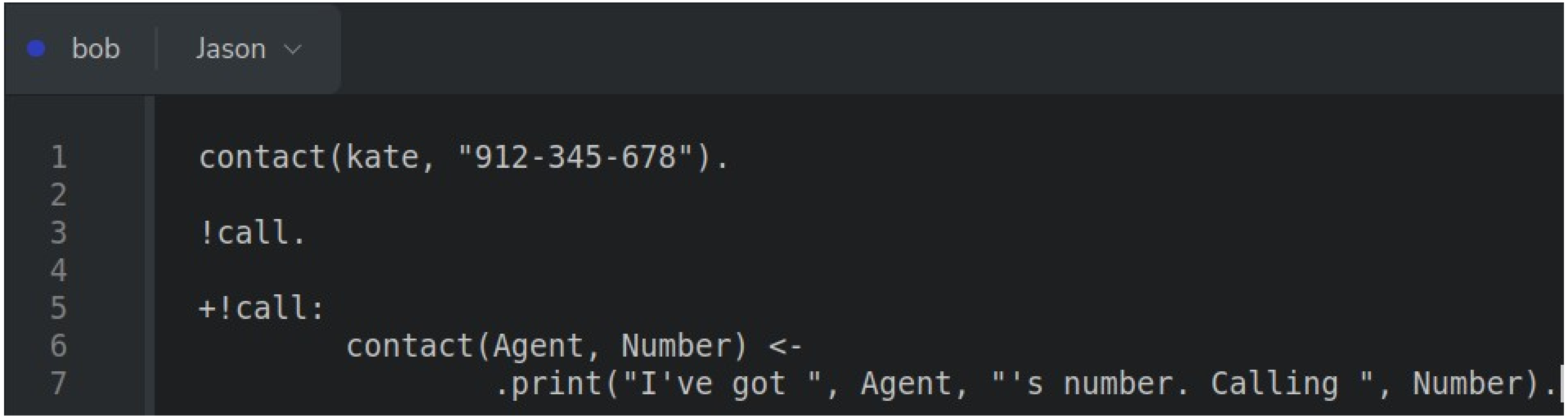
```
+!event[source(type)]:
    context ←
        action 1;
        action 2;
        action n.
```

Plans: Addition Achievement Goal

```
+!event[source(type)]:
    context ←
        action 1;
        action 2;
        action n.
```

defines an addition plan.

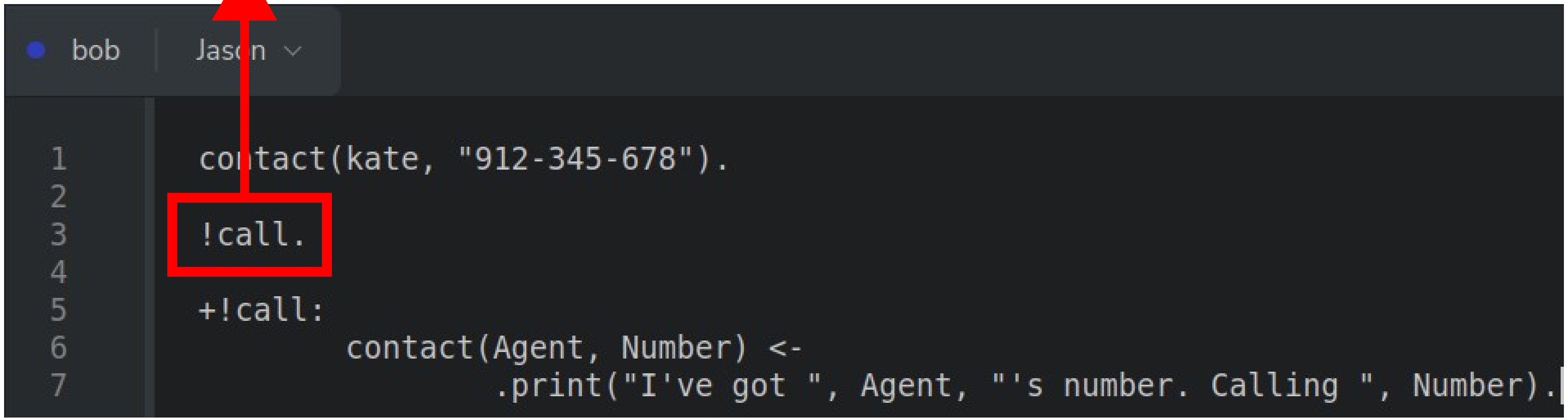
Addition Achievement Goal: Predicate



```
contact(kate, "912-345-678").

!call.

+!call:
        contact(Agent, Number) <-
                .print("I've got ", Agent, "'s number. Calling ", Number).
```

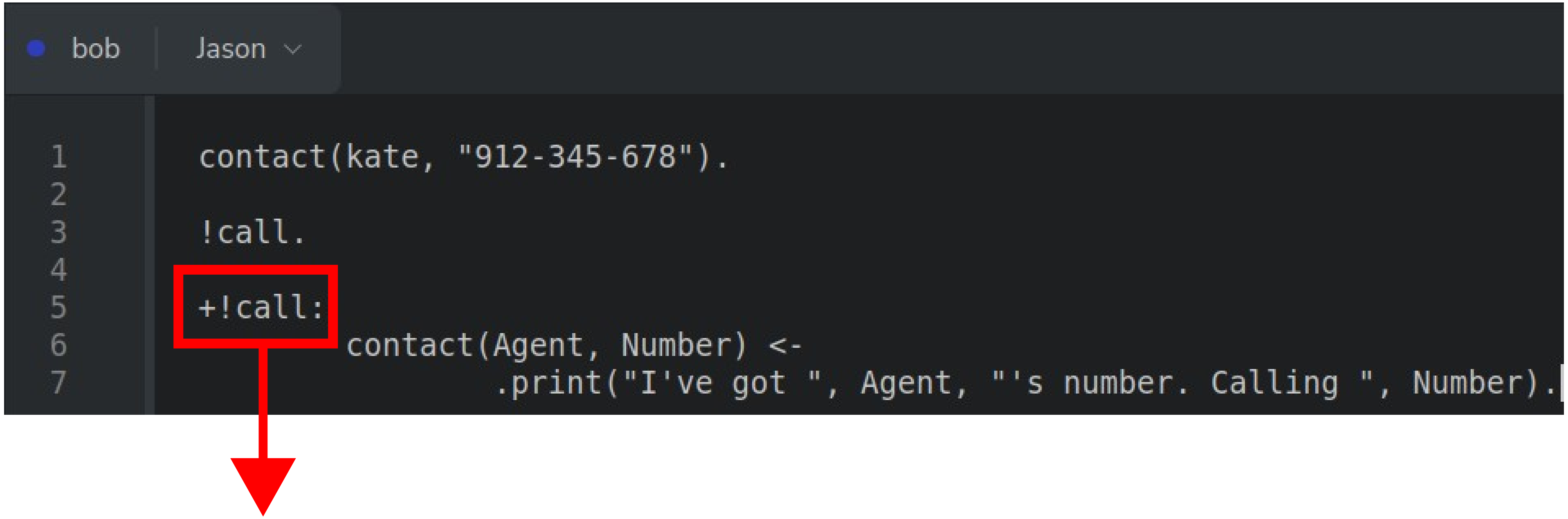
Addition Achievement Goal: Predicate

A predicate as a goal...



```
contact(kate, "912-345-678").

!call.

+!call:
        contact(Agent, Number) <-
                .print("I've got ", Agent, "'s number. Calling ", Number).
```

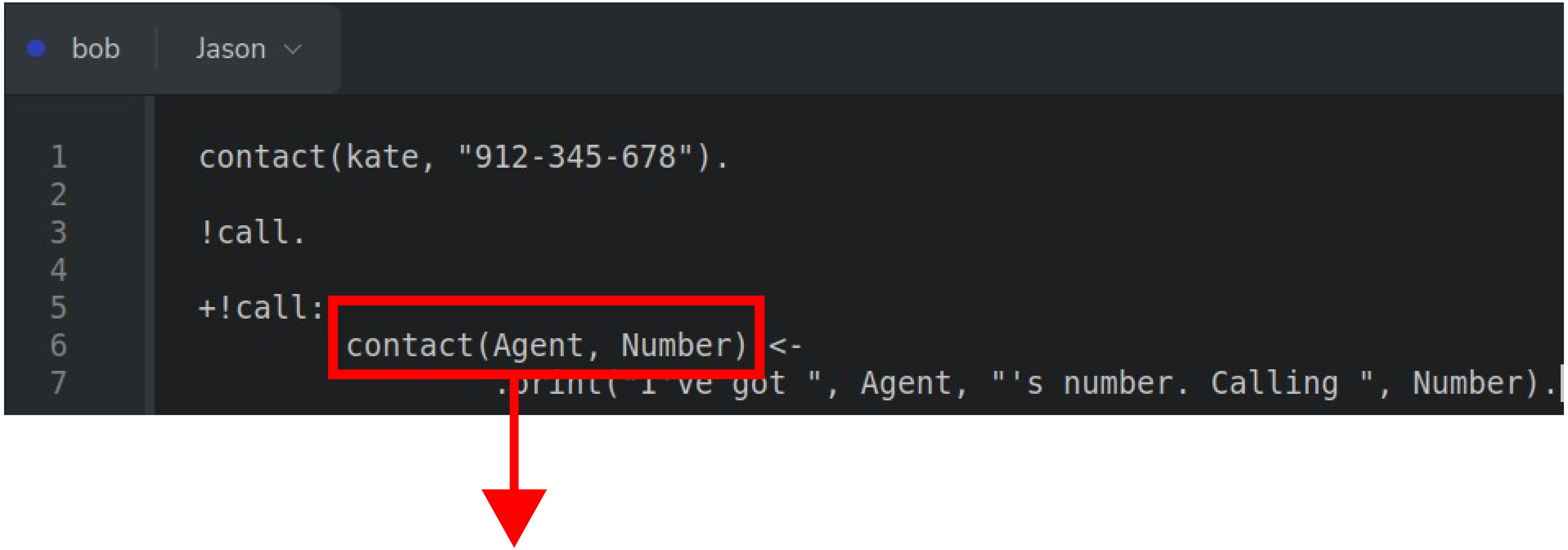
Addition Achievement Goal: Predicate



```
contact(kate, "912-345-678").

!call.

+!call:
        contact(Agent, Number) <-
                 .print("I've got ", Agent, "'s number. Calling ", Number).
```

… which activates a plan with the same name…

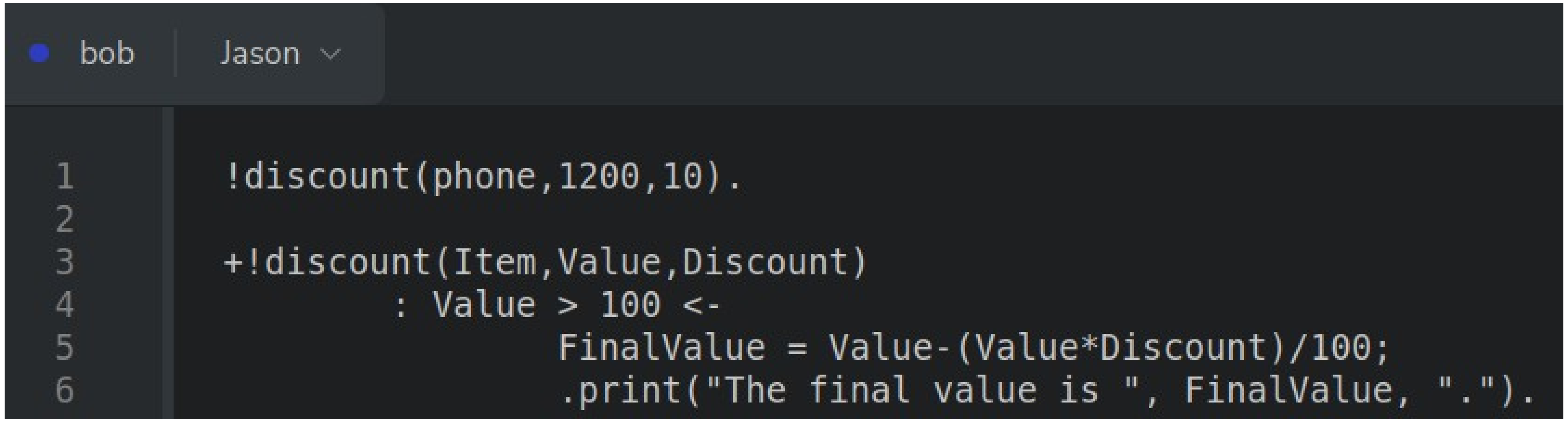
Addition Achievement Goal: Predicate



```
contact(kate, "912-345-678").

!call.

+!call:
        contact(Agent, Number) <-
                .print("I've got ", Agent, "'s number. Calling ", Number).
```

… if the context is satisfied.

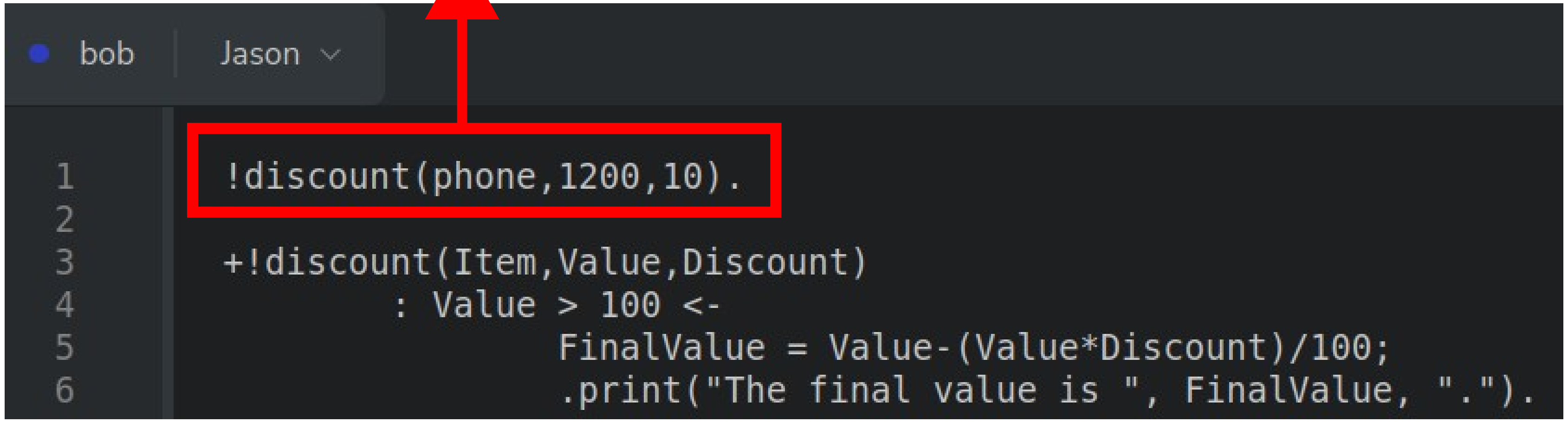
Addition Achievement Goal: One Plan



```
!discount(phone,1200,10).

+!discount(Item,Value,Discount)
        : Value > 100 <-
                FinalValue = Value-(Value*Discount)/100;
                .print("The final value is ", FinalValue, ".").
```

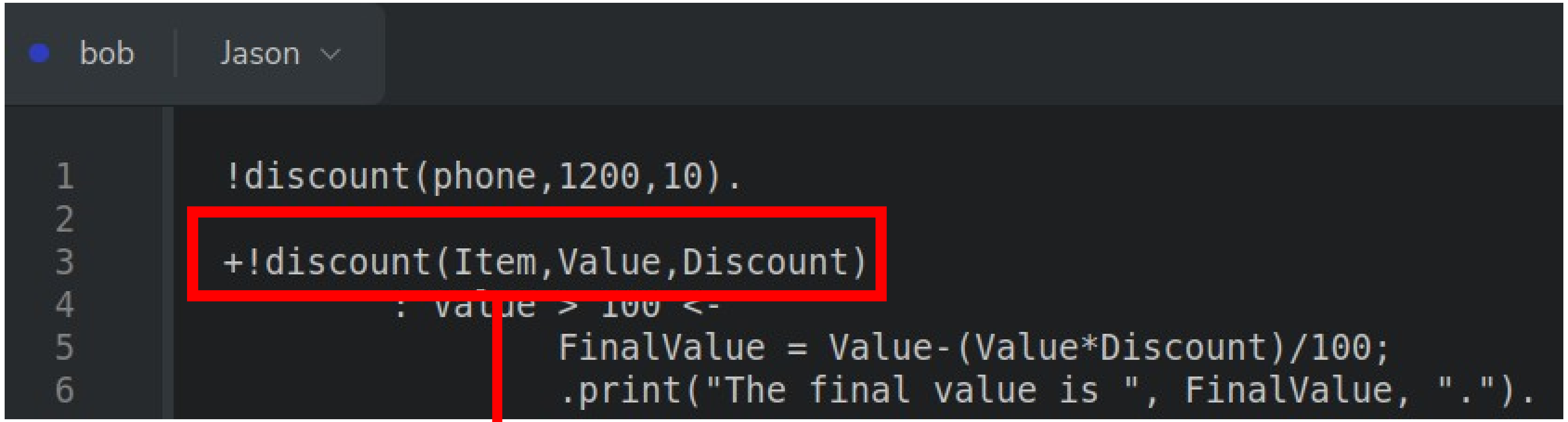
Addition Achievement Goal: One Plan

An initial goal with predicate and values.



```
!discount(phone,1200,10).

+!discount(Item,Value,Discount)
        : Value > 100 <-
                FinalValue = Value-(Value*Discount)/100;
                .print("The final value is ", FinalValue, ".").
```

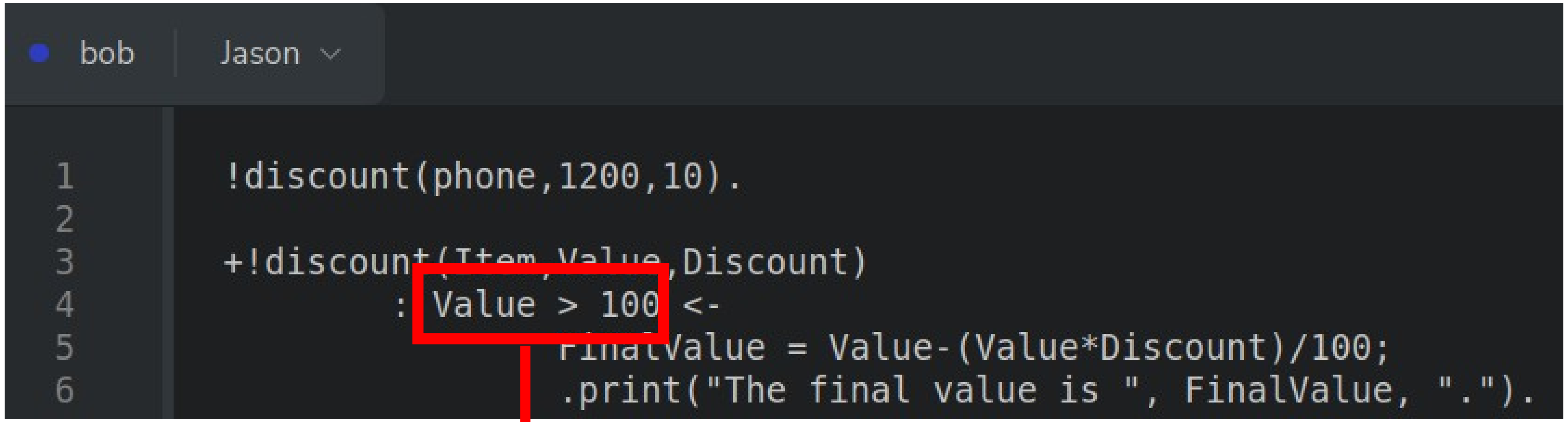
Addition Achievement Goal: One Plan



```
!discount(phone,1200,10).

+!discount(Item,Value,Discount)
        : Value > 100 <-
                FinalValue = Value-(Value*Discount)/100;
                .print("The final value is ", FinalValue, ".").
```

It activates this addition achievement goal...

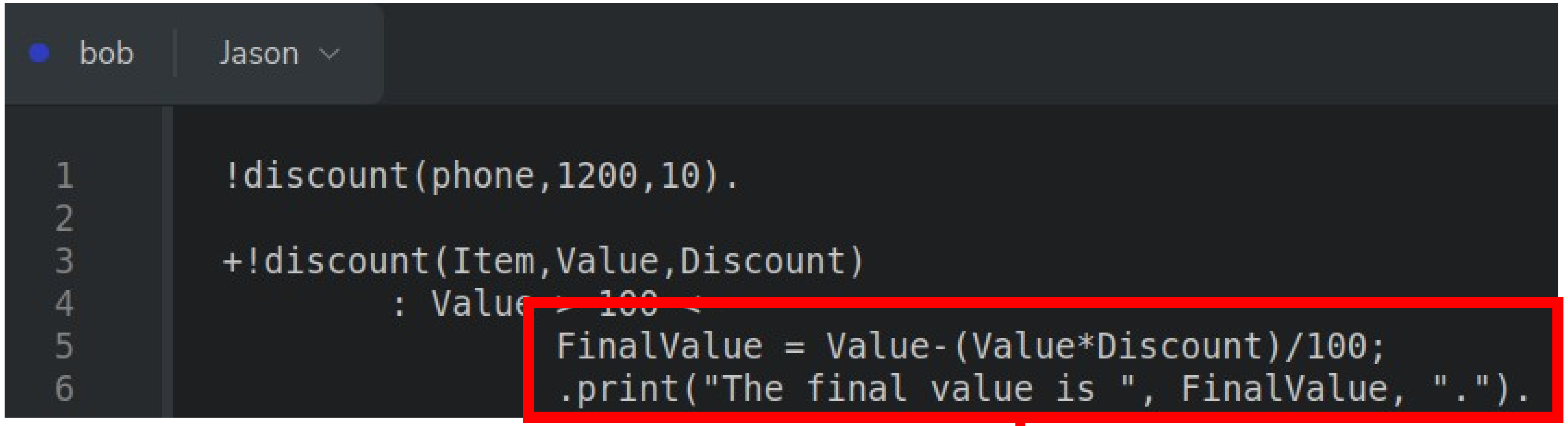
Addition Achievement Goal: One Plan



```
!discount(phone,1200,10).

+!discount(Item,Value,Discount)
        : Value > 100 <-
                FinalValue = Value-(Value*Discount)/100;
                .print("The final value is ", FinalValue, ".").
```

... if the context is satisfied.

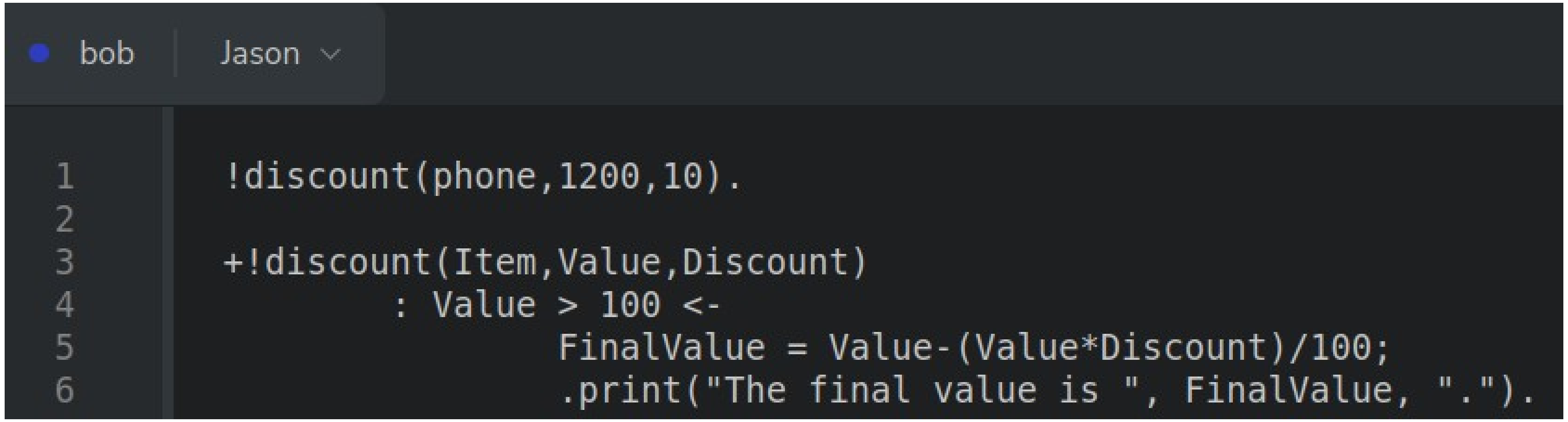
Addition Achievement Goal: One Plan



```
!discount(phone,1200,10).

+!discount(Item,Value,Discount)
        : Value > 100 <-
                FinalValue = Value-(Value*Discount)/100;
                .print("The final value is ", FinalValue, ".").
```

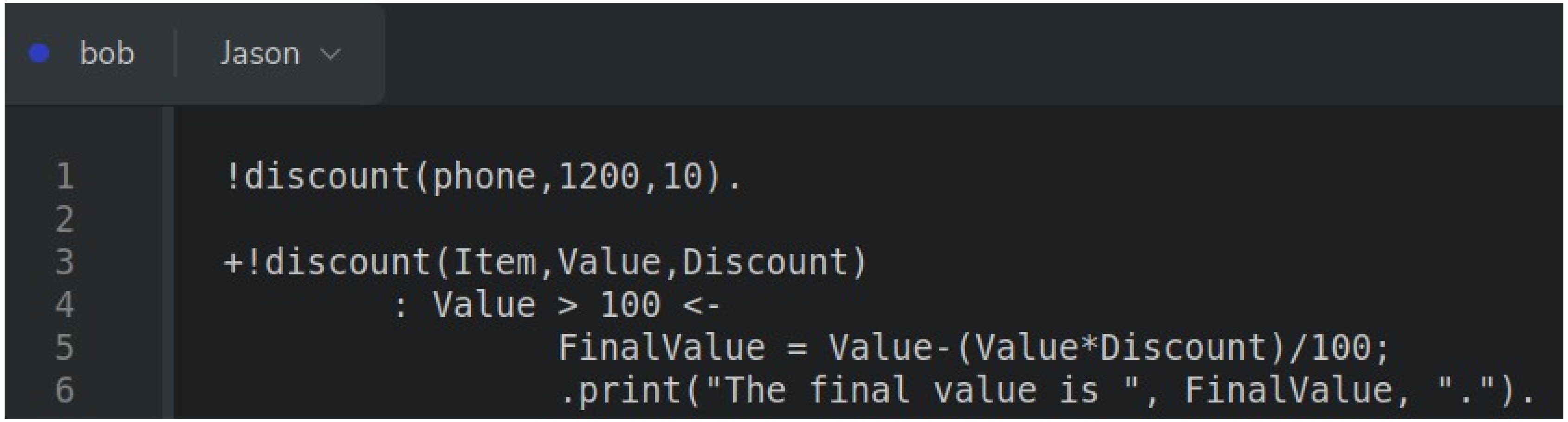
Then, the actions run.

Addition Achievement Goal: One Plan

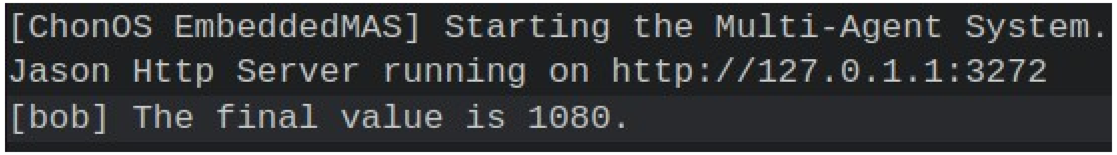


```
!discount(phone,1200,10).

+!discount(Item,Value,Discount)
        : Value > 100 <-
                FinalValue = Value-(Value*Discount)/100;
                .print("The final value is ", FinalValue, ".").
```

Addition Achievement Goal: One Plan

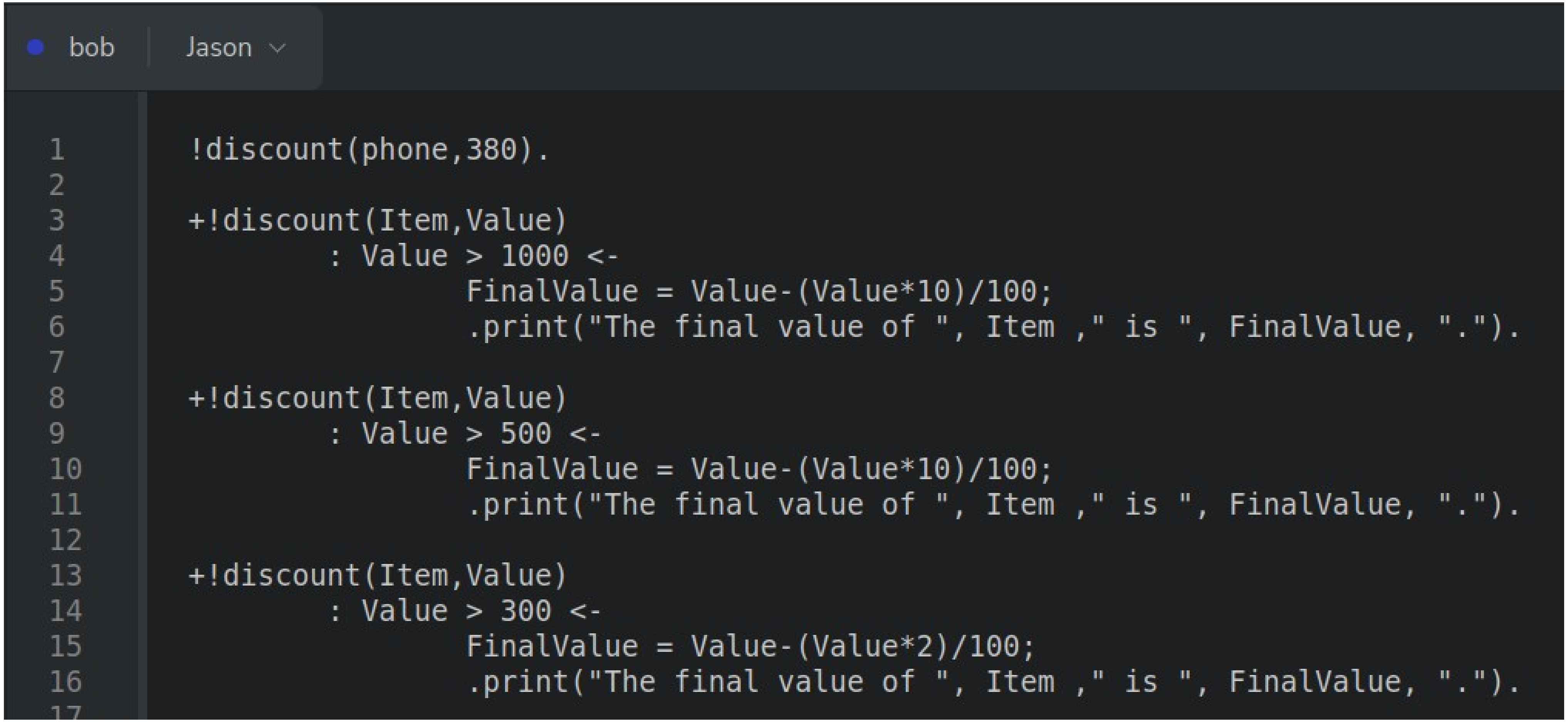


```
!discount(phone,1200,10).

+!discount(Item,Value,Discount)
        : Value > 100 <-
                FinalValue = Value-(Value*Discount)/100;
                .print("The final value is ", FinalValue, ".").
```



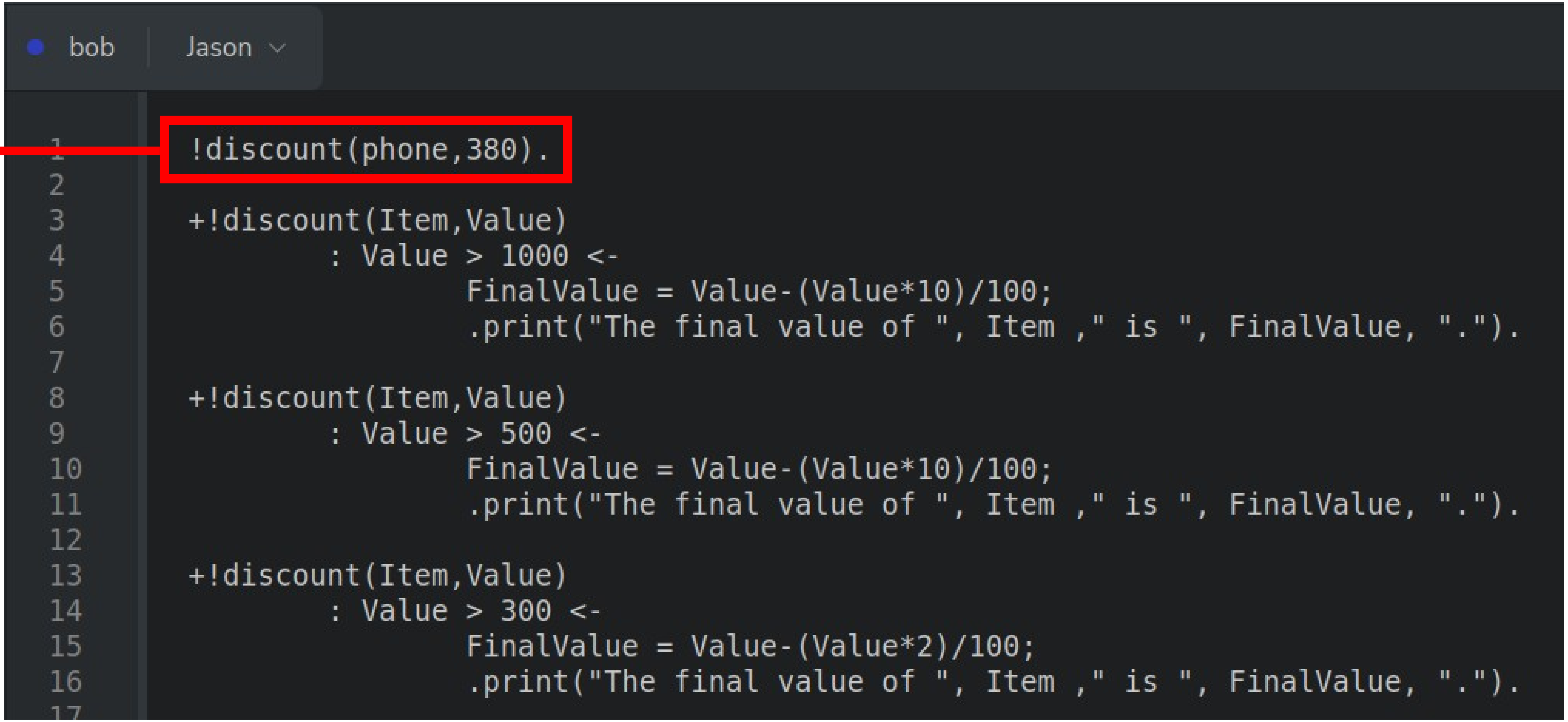
```
[ChonOS EmbeddedMAS] Starting the Multi-Agent System.
Jason Http Server running on http://127.0.1.1:3272
[bob] The final value is 1080.
```

Addition Achievement Goal: More Than One Plan



```
!discount(phone,380).

+!discount(Item,Value)
        : Value > 1000 <-
                FinalValue = Value-(Value*10)/100;
                .print("The final value of ", Item ," is ", FinalValue, ".").

+!discount(Item,Value)
        : Value > 500 <-
                FinalValue = Value-(Value*10)/100;
                .print("The final value of ", Item ," is ", FinalValue, ".").

+!discount(Item,Value)
        : Value > 300 <-
                FinalValue = Value-(Value*2)/100;
                .print("The final value of ", Item ," is ", FinalValue, ".").
```

Addition Achievement Goal: More Than One Plan

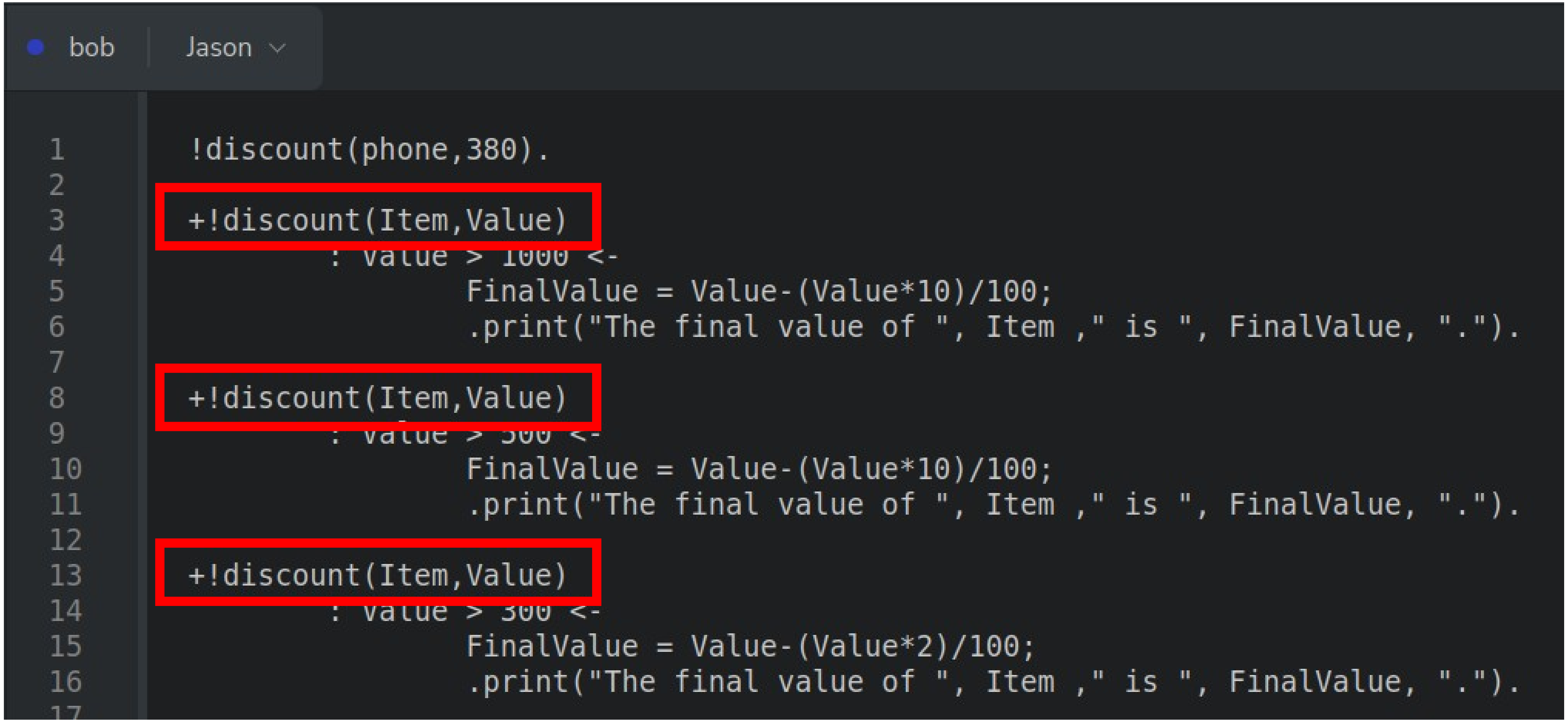


```
!discount(phone,380).

+!discount(Item,Value)
        : Value > 1000 <-
                FinalValue = Value-(Value*10)/100;
                .print("The final value of ", Item ," is ", FinalValue, ".").

+!discount(Item,Value)
        : Value > 500 <-
                FinalValue = Value-(Value*10)/100;
                .print("The final value of ", Item ," is ", FinalValue, ".").

+!discount(Item,Value)
        : Value > 300 <-
                FinalValue = Value-(Value*2)/100;
                .print("The final value of ", Item ," is ", FinalValue, ".").
```

The goal pursued.

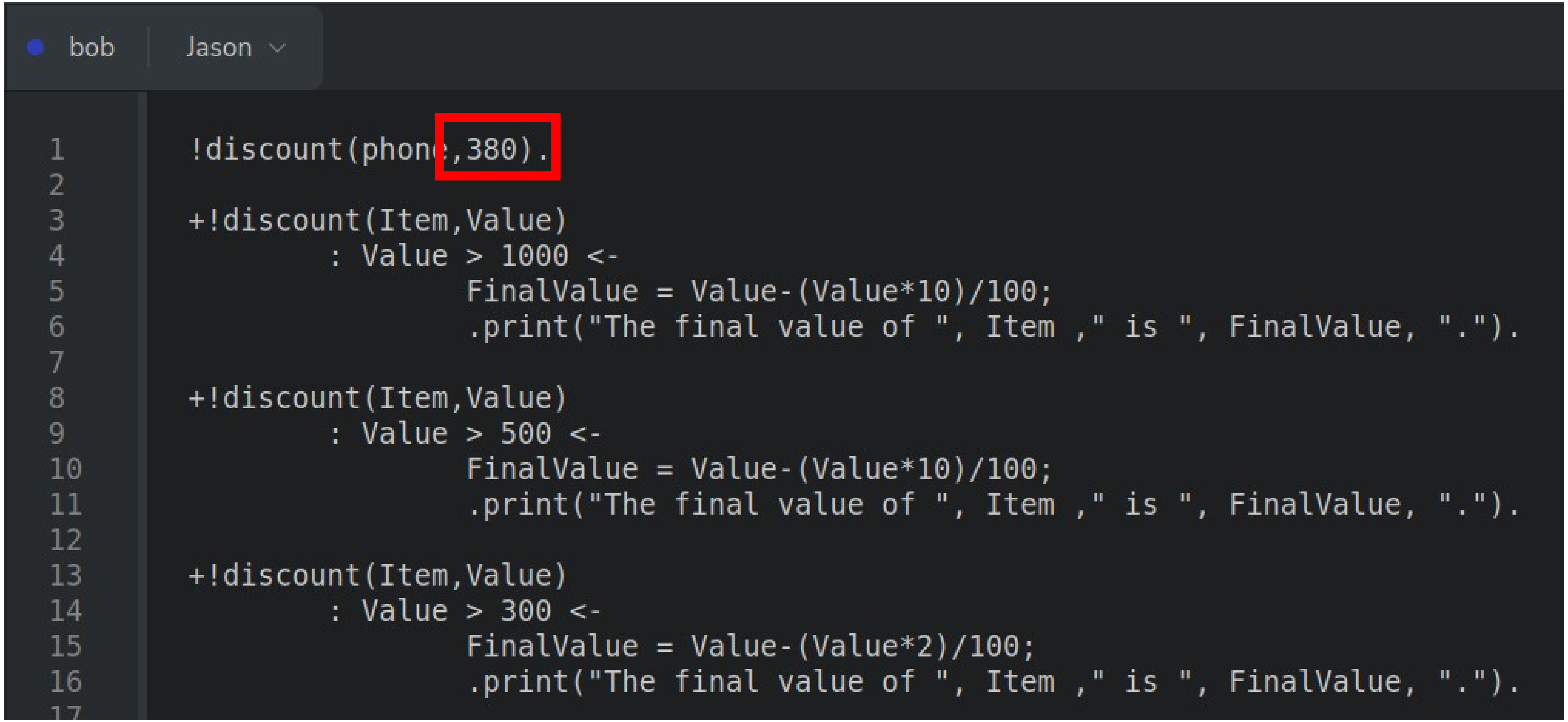
Addition Achievement Goal: More Than One Plan

The possible plans.



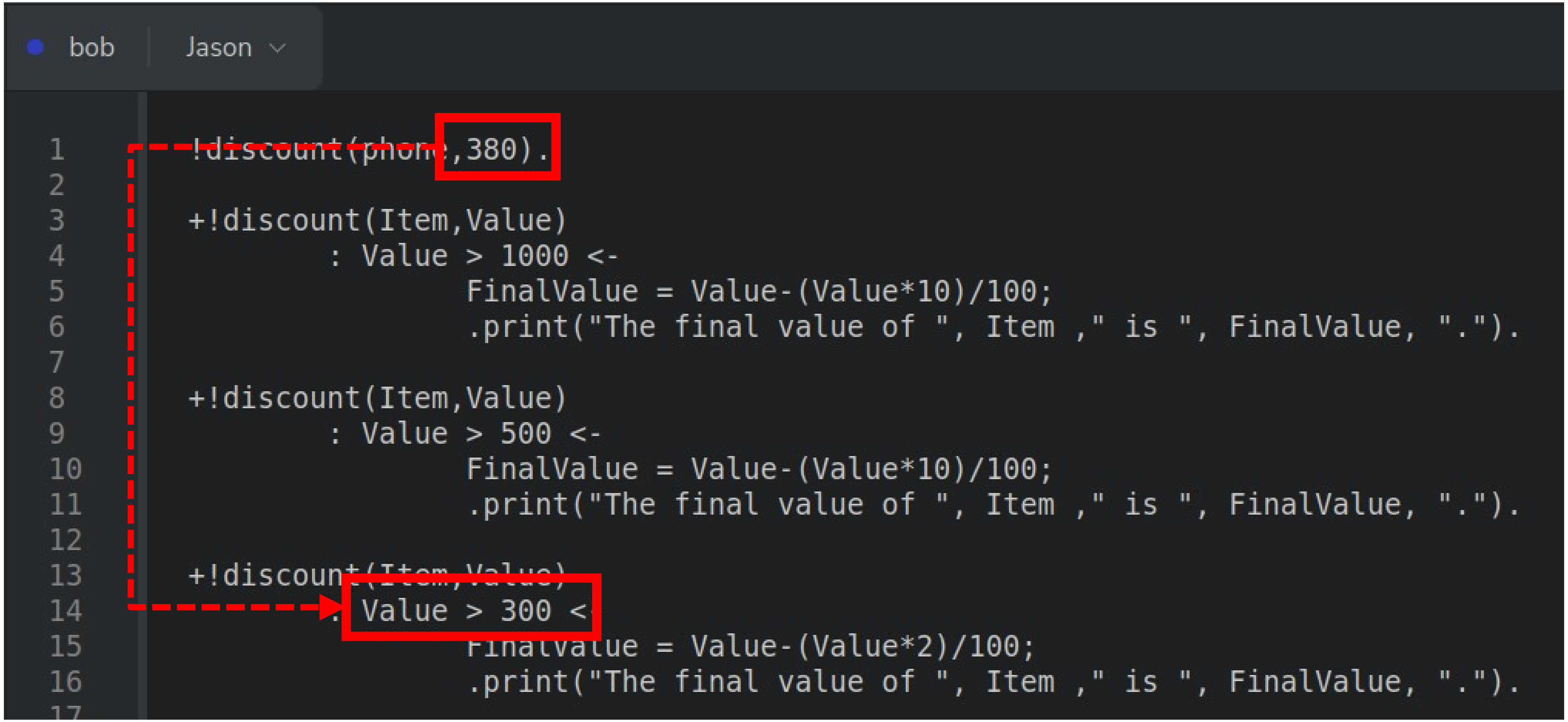
```
!discount(phone,380).

+!discount(Item,Value)
        : Value > 1000 <-
                FinalValue = Value-(Value*10)/100;
                .print("The final value of ", Item ," is ", FinalValue, ".").

+!discount(Item,Value)
        : Value > 500 <-
                FinalValue = Value-(Value*10)/100;
                .print("The final value of ", Item ," is ", FinalValue, ".").

+!discount(Item,Value)
        : Value > 300 <-
                FinalValue = Value-(Value*2)/100;
                .print("The final value of ", Item ," is ", FinalValue, ".").
```

Addition Achievement Goal: More Than One Plan



```
!discount(phone,380).

+!discount(Item,Value)
        : Value > 1000 <-
                FinalValue = Value-(Value*10)/100;
                .print("The final value of ", Item ," is ", FinalValue, ".").

+!discount(Item,Value)
        : Value > 500 <-
                FinalValue = Value-(Value*10)/100;
                .print("The final value of ", Item ," is ", FinalValue, ".").

+!discount(Item,Value)
        : Value > 300 <-
                FinalValue = Value-(Value*2)/100;
                .print("The final value of ", Item ," is ", FinalValue, ".").
```

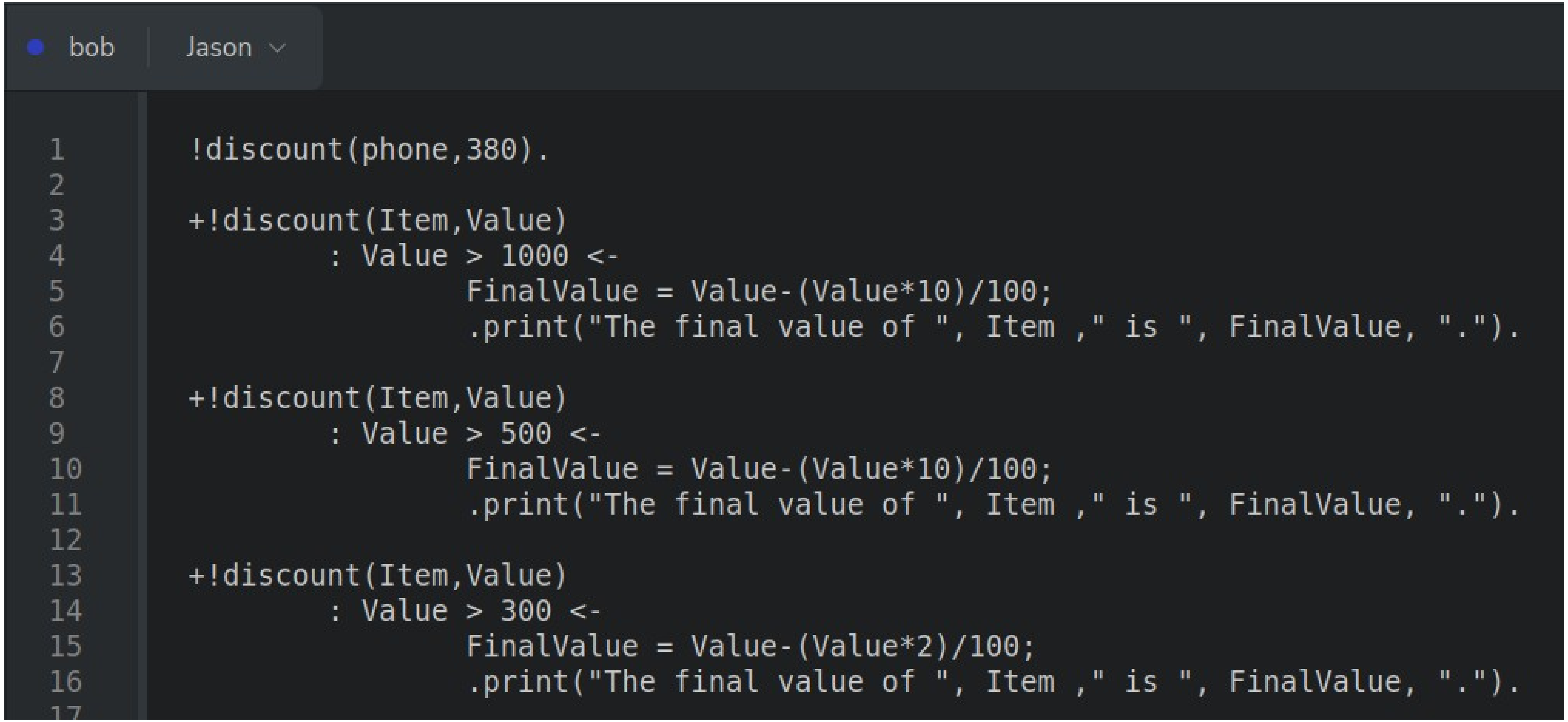
Addition Achievement Goal: More Than One Plan



```
!discount(phone,380).

+!discount(Item,Value)
        : Value > 1000 <-
                FinalValue = Value-(Value*10)/100;
                .print("The final value of ", Item ," is ", FinalValue, ".").

+!discount(Item,Value)
        : Value > 500 <-
                FinalValue = Value-(Value*10)/100;
                .print("The final value of ", Item ," is ", FinalValue, ".").

+!discount(Item,Value)
        : Value > 300 <-
                FinalValue = Value-(Value*2)/100;
                .print("The final value of ", Item ," is ", FinalValue, ".").
```

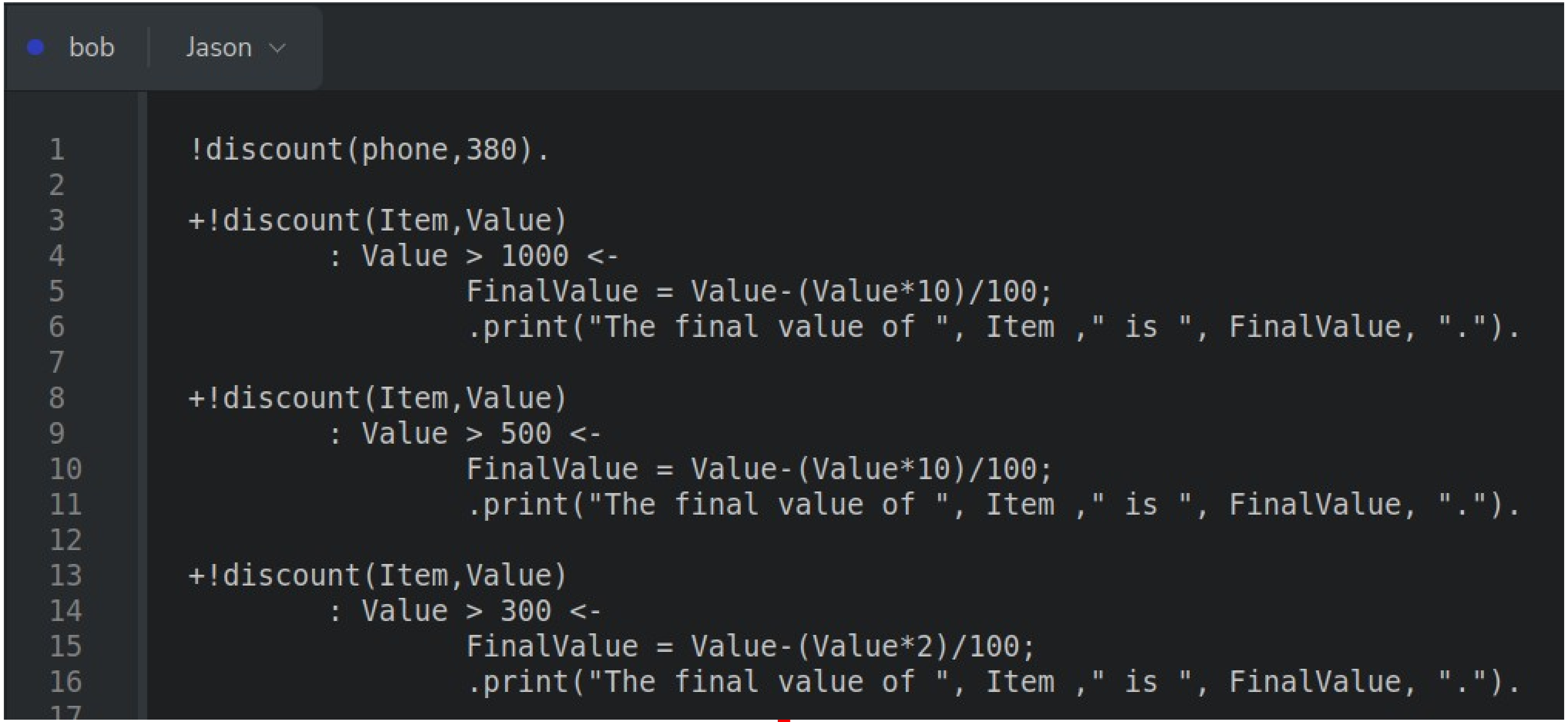
The activated plan.

Addition Achievement Goal: More Than One Plan

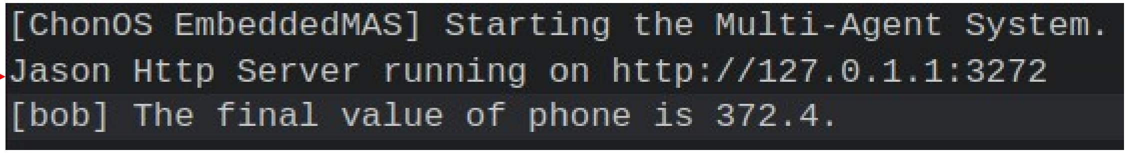


```
!discount(phone,380).

+!discount(Item,Value)
        : Value > 1000 <-
                FinalValue = Value-(Value*10)/100;
                .print("The final value of ", Item ," is ", FinalValue, ".").

+!discount(Item,Value)
        : Value > 500 <-
                FinalValue = Value-(Value*10)/100;
                .print("The final value of ", Item ," is ", FinalValue, ".").

+!discount(Item,Value)
        : Value > 300 <-
                FinalValue = Value-(Value*2)/100;
                .print("The final value of ", Item ," is ", FinalValue, ".").
```

Addition Achievement Goal: More Than One Plan



```
!discount(phone,380).

+!discount(Item,Value)
        : Value > 1000 <-
                FinalValue = Value-(Value*10)/100;
                .print("The final value of ", Item ," is ", FinalValue, ".").

+!discount(Item,Value)
        : Value > 500 <-
                FinalValue = Value-(Value*10)/100;
                .print("The final value of ", Item ," is ", FinalValue, ".").

+!discount(Item,Value)
        : Value > 300 <-
                FinalValue = Value-(Value*2)/100;
                .print("The final value of ", Item ," is ", FinalValue, ".").
```



```
[ChonOS EmbeddedMAS] Starting the Multi-Agent System.
Jason Http Server running on http://127.0.1.1:3272
[bob] The final value of phone is 372.4.
```

Addition Achievement Goal: More Than One Plan

When there are more plans with the **same event** name, the agent will try them **one by one**...

Addition Achievement Goal: More Than One Plan

+!event[source(type)]: context ← (...).

+!event[source(type)]: context ← (...).

+!event[source(type)]: context ← (...).

⋮

+!event[source(type)]: context ← (...).

Addition Achievement Goal: More Than One Plan

+!event[source(type)]: context ← (...).

+!event[source(type)]: context ← (...).

+!event[source(type)]: context ← (...).

⋮

+!event[source(type)]: context ← (...).

Addition Achievement Goal: More Than One Plan

```
+!event[source(type)]: context ← (...).
+!event[source(type)]: context ← (...).
+!event[source(type)]: context ← (...).
                    ⋮
+!event[source(type)]: context ← (...).
```

Addition Achievement Goal: More Than One Plan

```
+!event[source(type)]: context ← (...).
+!event[source(type)]: context ← (...).
+!event[source(type)]: context ← (...).
.
.
.
+!event[source(type)]: context ← (...).
```

Addition Achievement Goal: More Than One Plan

… until it finds one that **satisfies the context**.

Addition Achievement Goal: More Than One Plan

+!event[source(type)]: context ← (...).

+!event[source(type)]: context ← (...).

+!event[source(type)]: context ← (...).

⋮

+!event[source(type)]: context ← (...).

Addition Achievement Goal: More Than One Plan

When there are two or more plans that could be **activated** at the **same time**...

Addition Achievement Goal: More Than One Plan

+!event[source(type)]: context ← (...).

+!event[source(type)]: context ← (...).

+!event[source(type)]: context ← (...).

⋮

+!event[source(type)]: context ← (...).

Addition Achievement Goal: More Than One Plan

+!event[source(type)]: context ← (...).

+!event[source(type)]: context ← (...).

+!event[source(type)]: context ← (...).

⋮

+!event[source(type)]: context ← (...).

Addition Achievement Goal: More Than One Plan

… it chooses the first one that **fits the context**.

Addition Achievement Goal: More Than One Plan

+!event[source(type)]: context ← (...).

+!event[source(type)]: context ← (...).

+!event[source(type)]: context ← (...).

⋮

+!event[source(type)]: context ← (...).

Addition Achievement Goal: More Than One Plan

However, when no one is activated, the agent finds a goal for which **no relevant plan** is available.

Plans: Deletion Achievement Goal

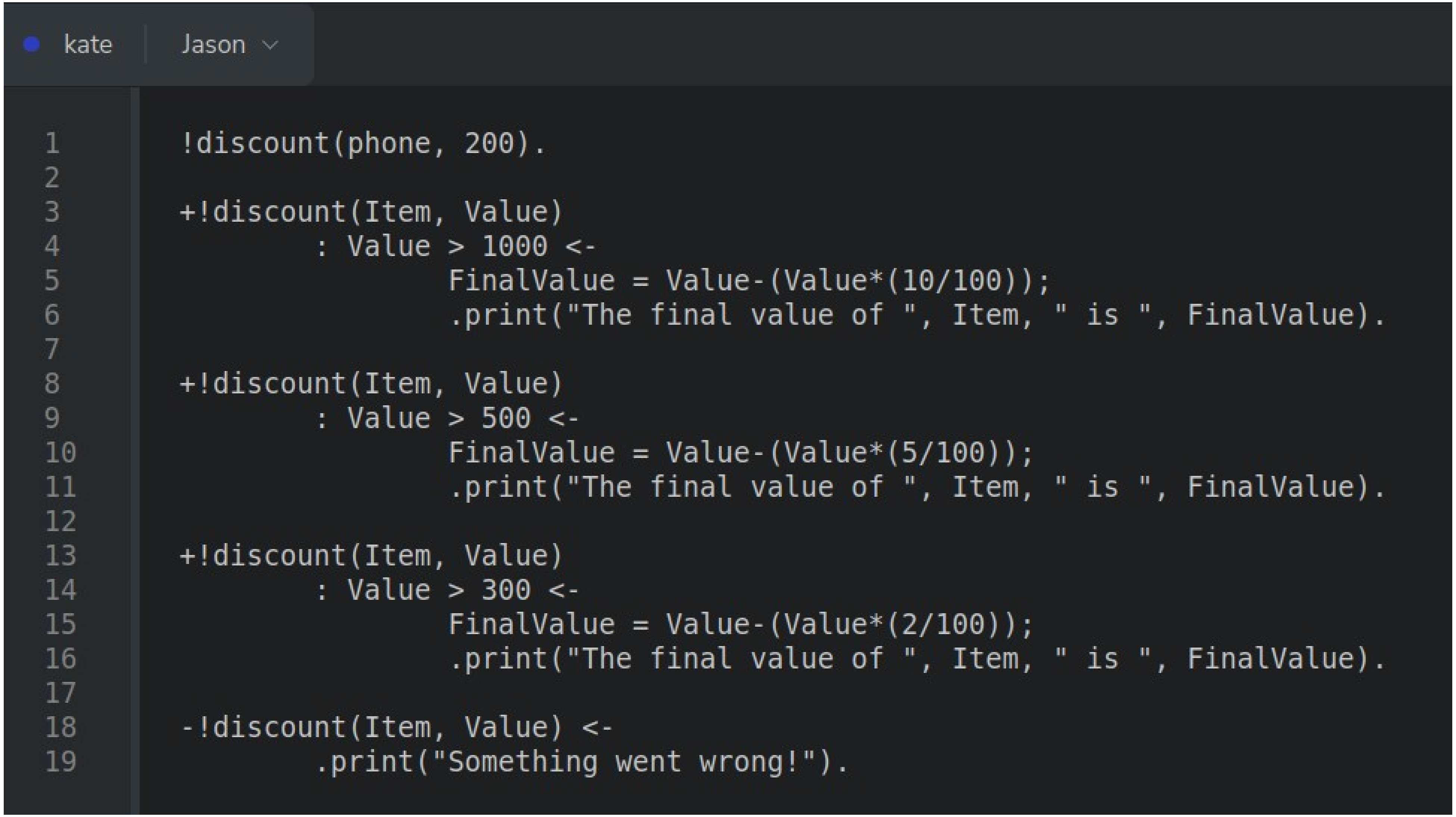
```
-!event[source(type)]:
    context ←
        action 1;
        action 2;
        action n.
```

Plans: Deletion Achievement Goal

```
-!event[source(type)]:
     context ←
          action 1;
          action 2;
          action n.
```

defines a deletion plan.

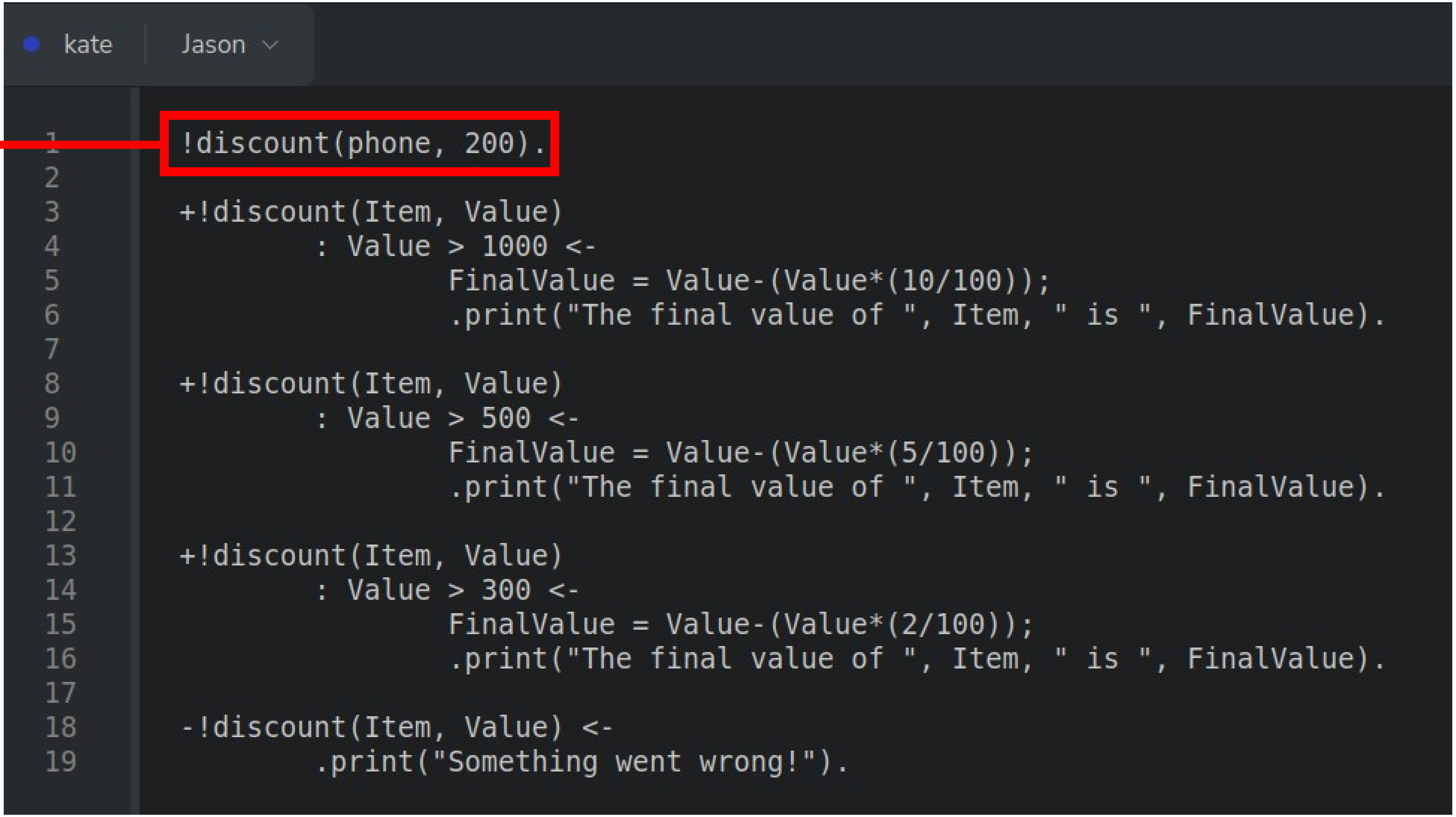
Plans: Deletion Achievement Goal



```
!discount(phone, 200).

+!discount(Item, Value)
        : Value > 1000 <-
                FinalValue = Value-(Value*(10/100));
                .print("The final value of ", Item, " is ", FinalValue).

+!discount(Item, Value)
        : Value > 500 <-
                FinalValue = Value-(Value*(5/100));
                .print("The final value of ", Item, " is ", FinalValue).

+!discount(Item, Value)
        : Value > 300 <-
                FinalValue = Value-(Value*(2/100));
                .print("The final value of ", Item, " is ", FinalValue).

-!discount(Item, Value) <-
        .print("Something went wrong!").
```

Plans: Deletion Achievement Goal

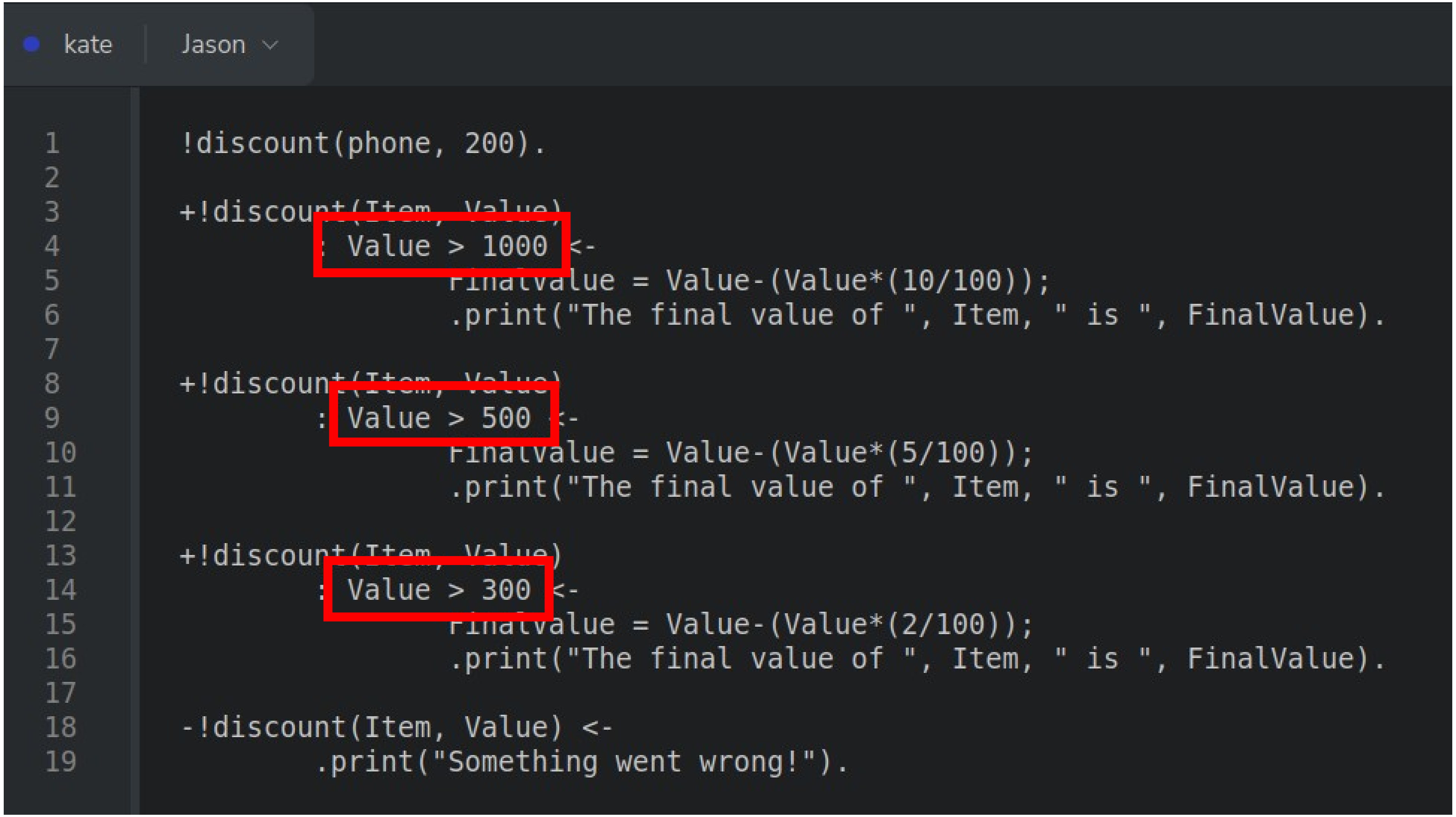
By using these values...



```
!discount(phone, 200).

+!discount(Item, Value)
        : Value > 1000 <-
                FinalValue = Value-(Value*(10/100));
                .print("The final value of ", Item, " is ", FinalValue).

+!discount(Item, Value)
        : Value > 500 <-
                FinalValue = Value-(Value*(5/100));
                .print("The final value of ", Item, " is ", FinalValue).

+!discount(Item, Value)
        : Value > 300 <-
                FinalValue = Value-(Value*(2/100));
                .print("The final value of ", Item, " is ", FinalValue).

-!discount(Item, Value) <-
        .print("Something went wrong!").
```

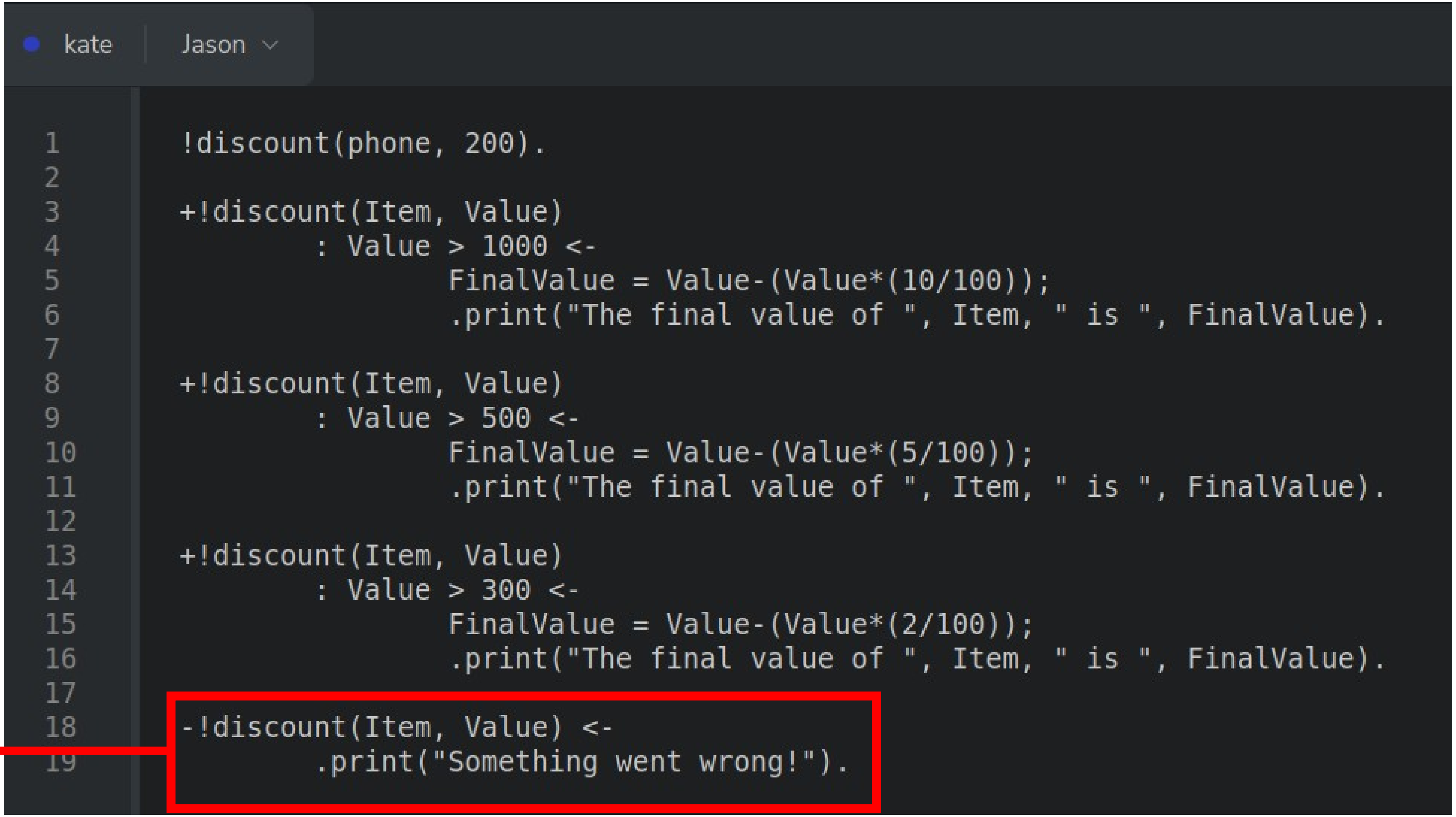
Plans: Deletion Achievement Goal

… none of the available plans activate.



```
!discount(phone, 200).

+!discount(Item, Value)
        : Value > 1000 <-
                FinalValue = Value-(Value*(10/100));
                .print("The final value of ", Item, " is ", FinalValue).

+!discount(Item, Value)
        : Value > 500 <-
                FinalValue = Value-(Value*(5/100));
                .print("The final value of ", Item, " is ", FinalValue).

+!discount(Item, Value)
        : Value > 300 <-
                FinalValue = Value-(Value*(2/100));
                .print("The final value of ", Item, " is ", FinalValue).

-!discount(Item, Value) <-
        .print("Something went wrong!").
```

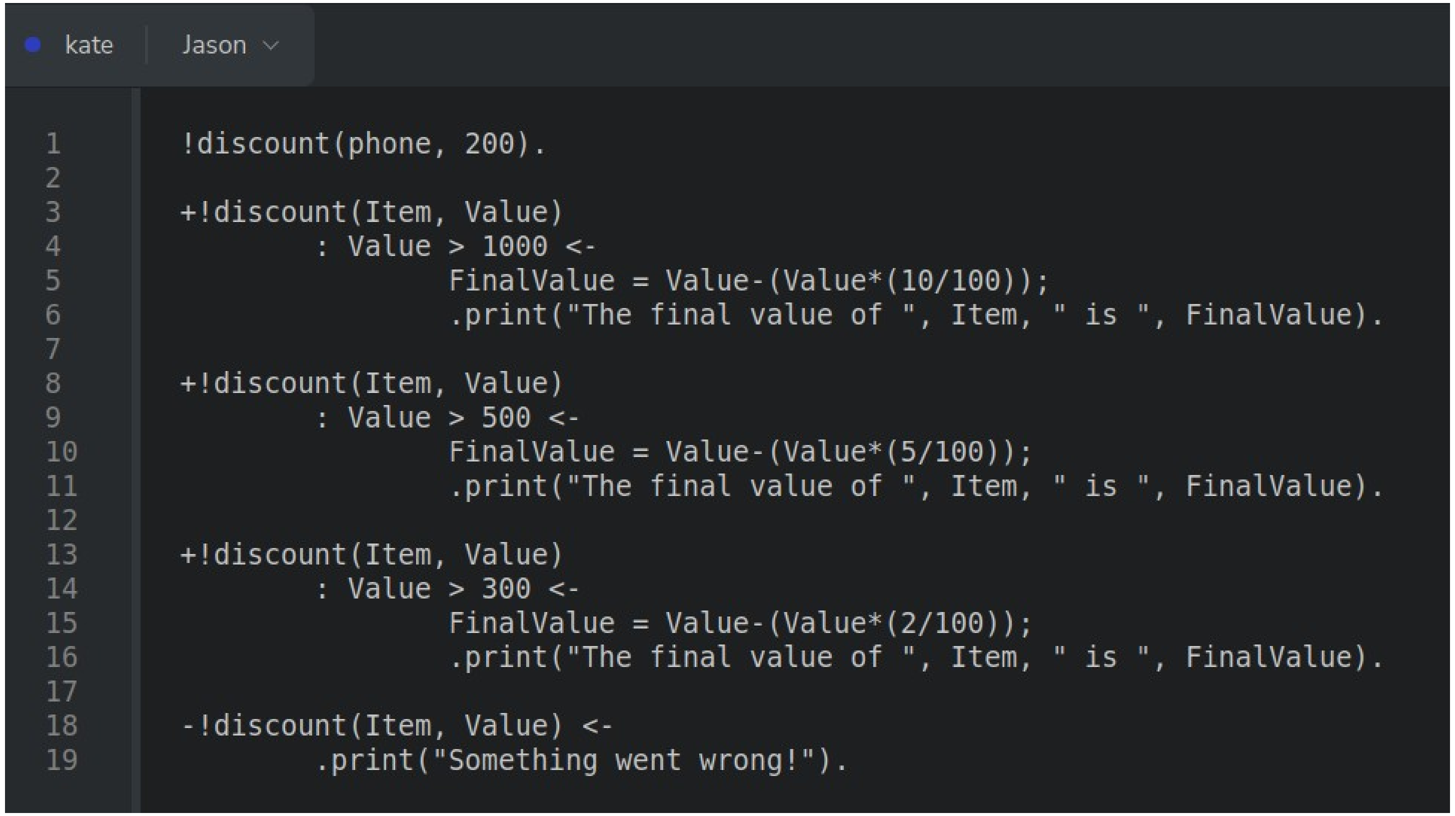
Plans: Deletion Achievement Goal



```
!discount(phone, 200).

+!discount(Item, Value)
        : Value > 1000 <-
                FinalValue = Value-(Value*(10/100));
                .print("The final value of ", Item, " is ", FinalValue).

+!discount(Item, Value)
        : Value > 500 <-
                FinalValue = Value-(Value*(5/100));
                .print("The final value of ", Item, " is ", FinalValue).

+!discount(Item, Value)
        : Value > 300 <-
                FinalValue = Value-(Value*(2/100));
                .print("The final value of ", Item, " is ", FinalValue).

-!discount(Item, Value) <-
        .print("Something went wrong!").
```

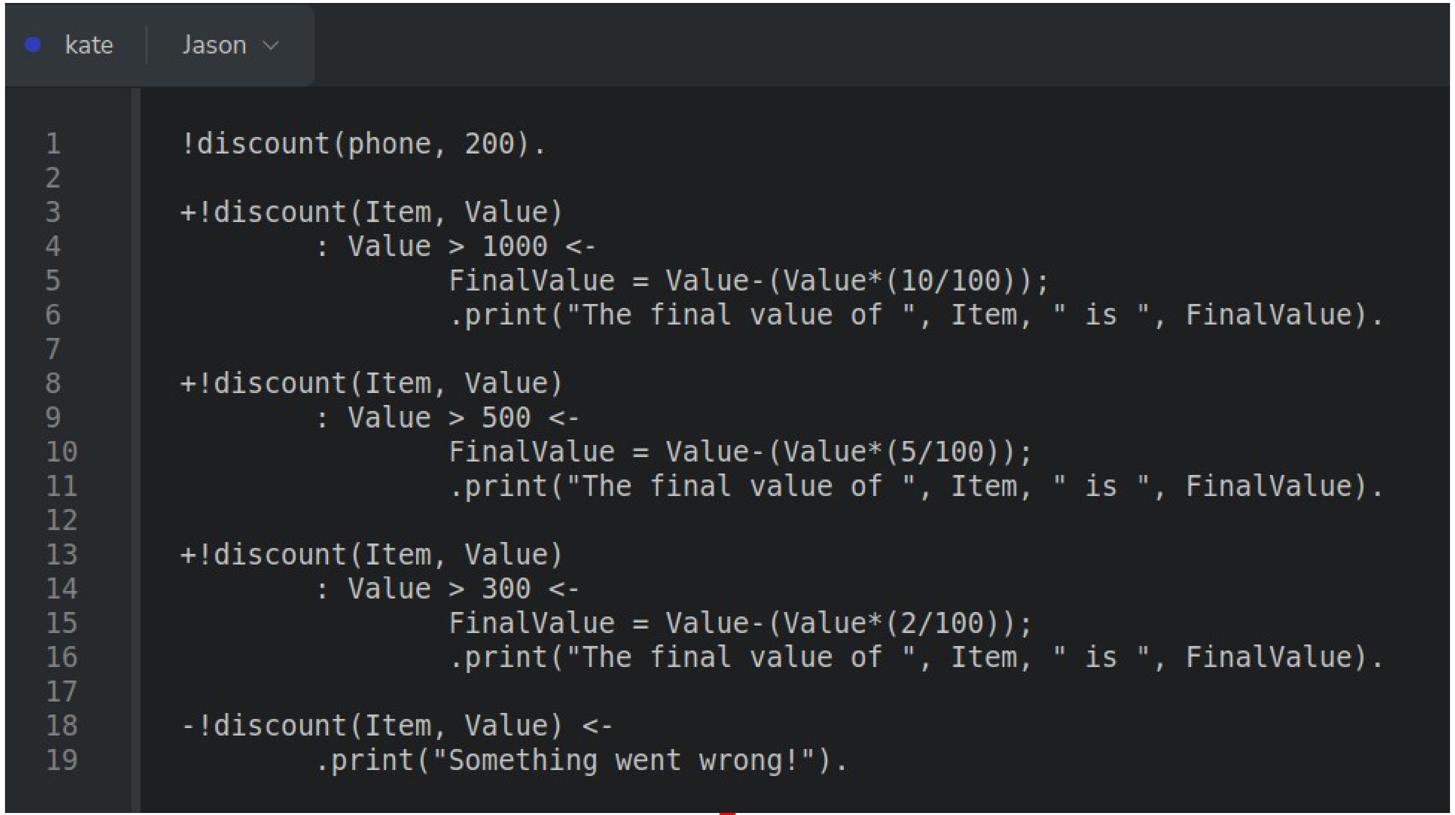
Then, a contingency plan activates.

Plans: Deletion Achievement Goal

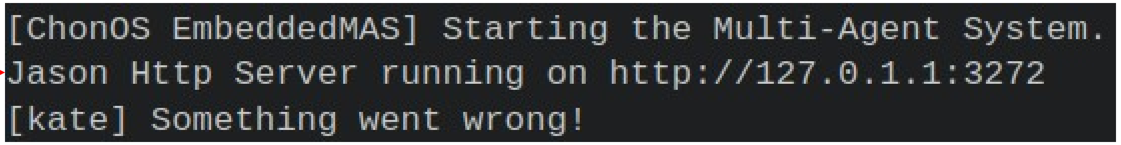


```
!discount(phone, 200).

+!discount(Item, Value)
        : Value > 1000 <-
                FinalValue = Value-(Value*(10/100));
                .print("The final value of ", Item, " is ", FinalValue).

+!discount(Item, Value)
        : Value > 500 <-
                FinalValue = Value-(Value*(5/100));
                .print("The final value of ", Item, " is ", FinalValue).

+!discount(Item, Value)
        : Value > 300 <-
                FinalValue = Value-(Value*(2/100));
                .print("The final value of ", Item, " is ", FinalValue).

-!discount(Item, Value) <-
        .print("Something went wrong!").
```

Plans: Deletion Achievement Goal



```
!discount(phone, 200).

+!discount(Item, Value)
        : Value > 1000 <-
                FinalValue = Value-(Value*(10/100));
                .print("The final value of ", Item, " is ", FinalValue).

+!discount(Item, Value)
        : Value > 500 <-
                FinalValue = Value-(Value*(5/100));
                .print("The final value of ", Item, " is ", FinalValue).

+!discount(Item, Value)
        : Value > 300 <-
                FinalValue = Value-(Value*(2/100));
                .print("The final value of ", Item, " is ", FinalValue).

-!discount(Item, Value) <-
        .print("Something went wrong!").
```

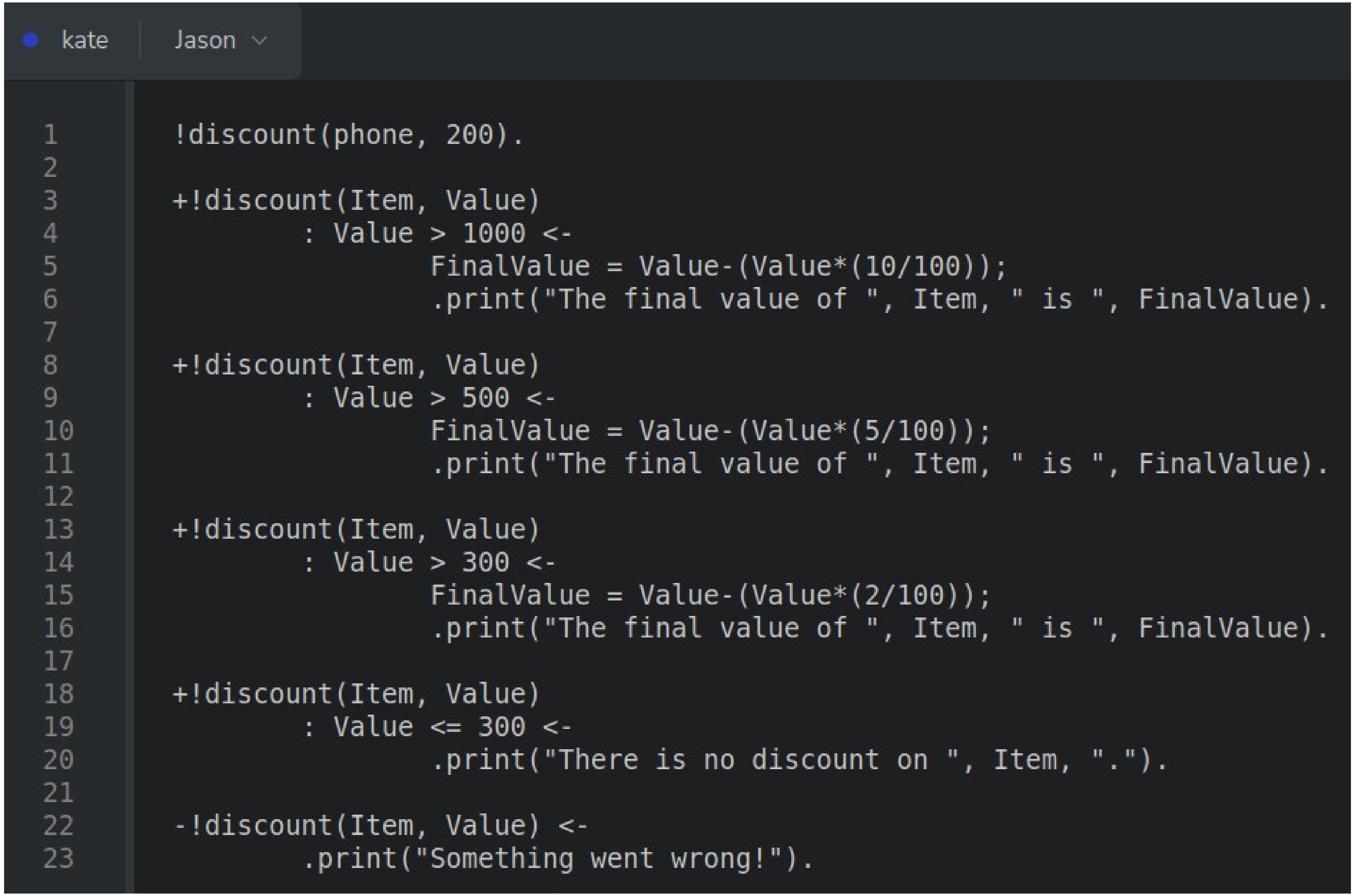


```
[ChonOS EmbeddedMAS] Starting the Multi-Agent System.
Jason Http Server running on http://127.0.1.1:3272
[kate] Something went wrong!
```

Plans: Deletion Achievement Goal

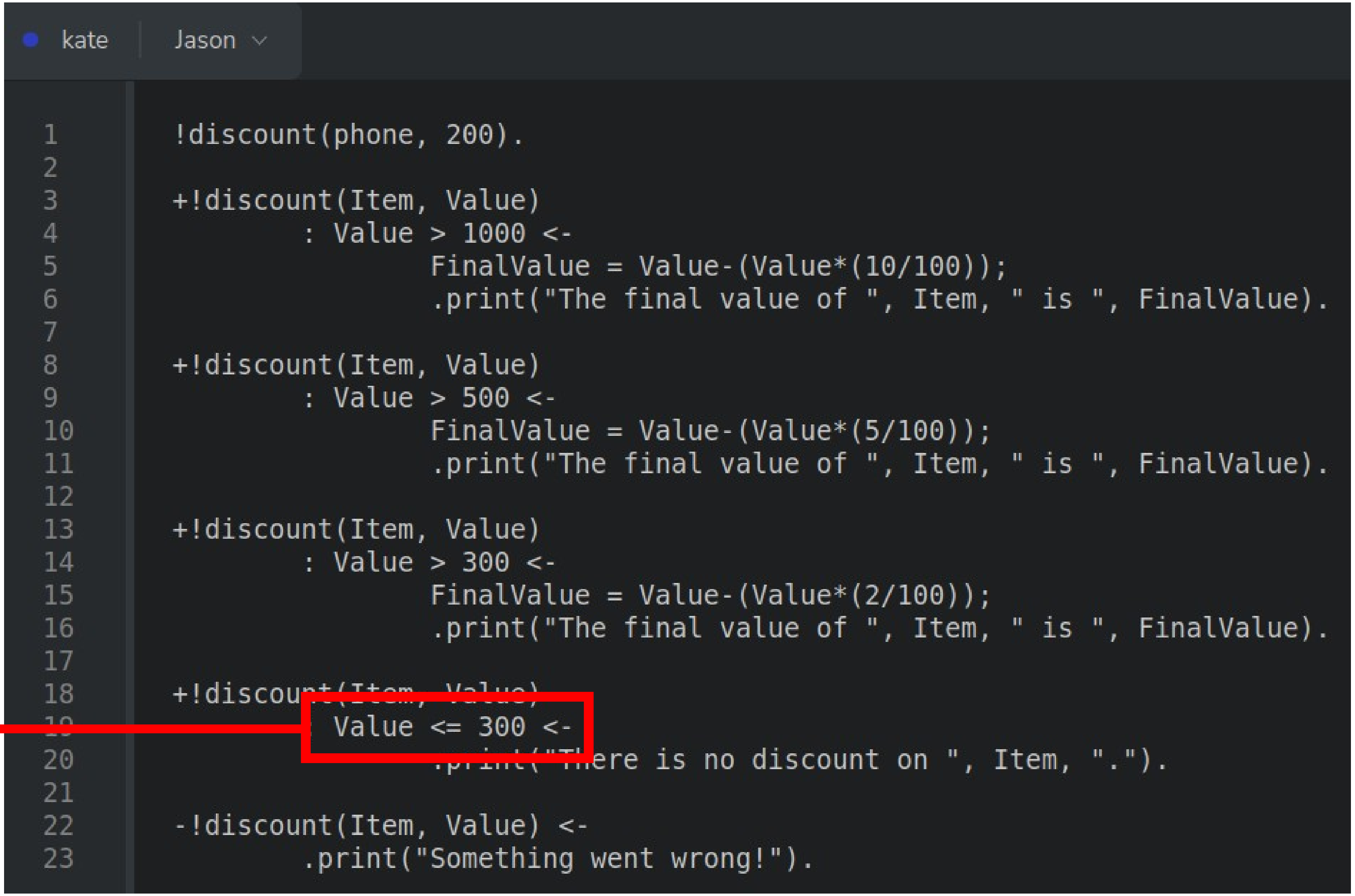
In this case, another **addition achievement** plan can cover the **missing context**.

Plans: Deletion Achievement Goal



```
!discount(phone, 200).

+!discount(Item, Value)
        : Value > 1000 <-
                FinalValue = Value-(Value*(10/100));
                .print("The final value of ", Item, " is ", FinalValue).

+!discount(Item, Value)
        : Value > 500 <-
                FinalValue = Value-(Value*(5/100));
                .print("The final value of ", Item, " is ", FinalValue).

+!discount(Item, Value)
        : Value > 300 <-
                FinalValue = Value-(Value*(2/100));
                .print("The final value of ", Item, " is ", FinalValue).

+!discount(Item, Value)
        : Value <= 300 <-
                .print("There is no discount on ", Item, ".").

-!discount(Item, Value) <-
        .print("Something went wrong!").
```

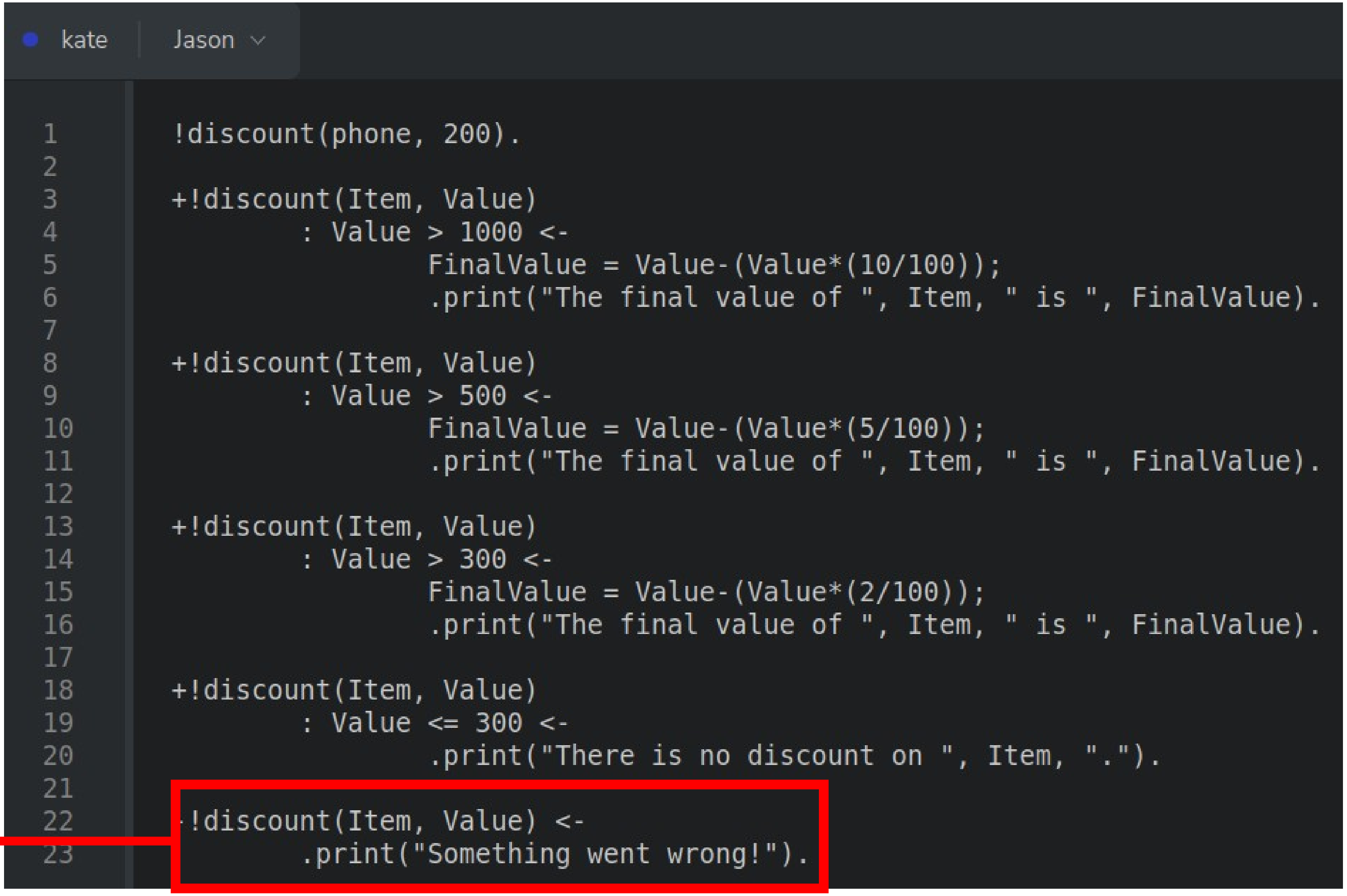
Plans: Deletion Achievement Goal



```
!discount(phone, 200).

+!discount(Item, Value)
        : Value > 1000 <-
                FinalValue = Value-(Value*(10/100));
                .print("The final value of ", Item, " is ", FinalValue).

+!discount(Item, Value)
        : Value > 500 <-
                FinalValue = Value-(Value*(5/100));
                .print("The final value of ", Item, " is ", FinalValue).

+!discount(Item, Value)
        : Value > 300 <-
                FinalValue = Value-(Value*(2/100));
                .print("The final value of ", Item, " is ", FinalValue).

+!discount(Item, Value)
        : Value <= 300 <-
                .print("There is no discount on ", Item, ".").

-!discount(Item, Value) <-
        .print("Something went wrong!").
```

So, this plan activates.

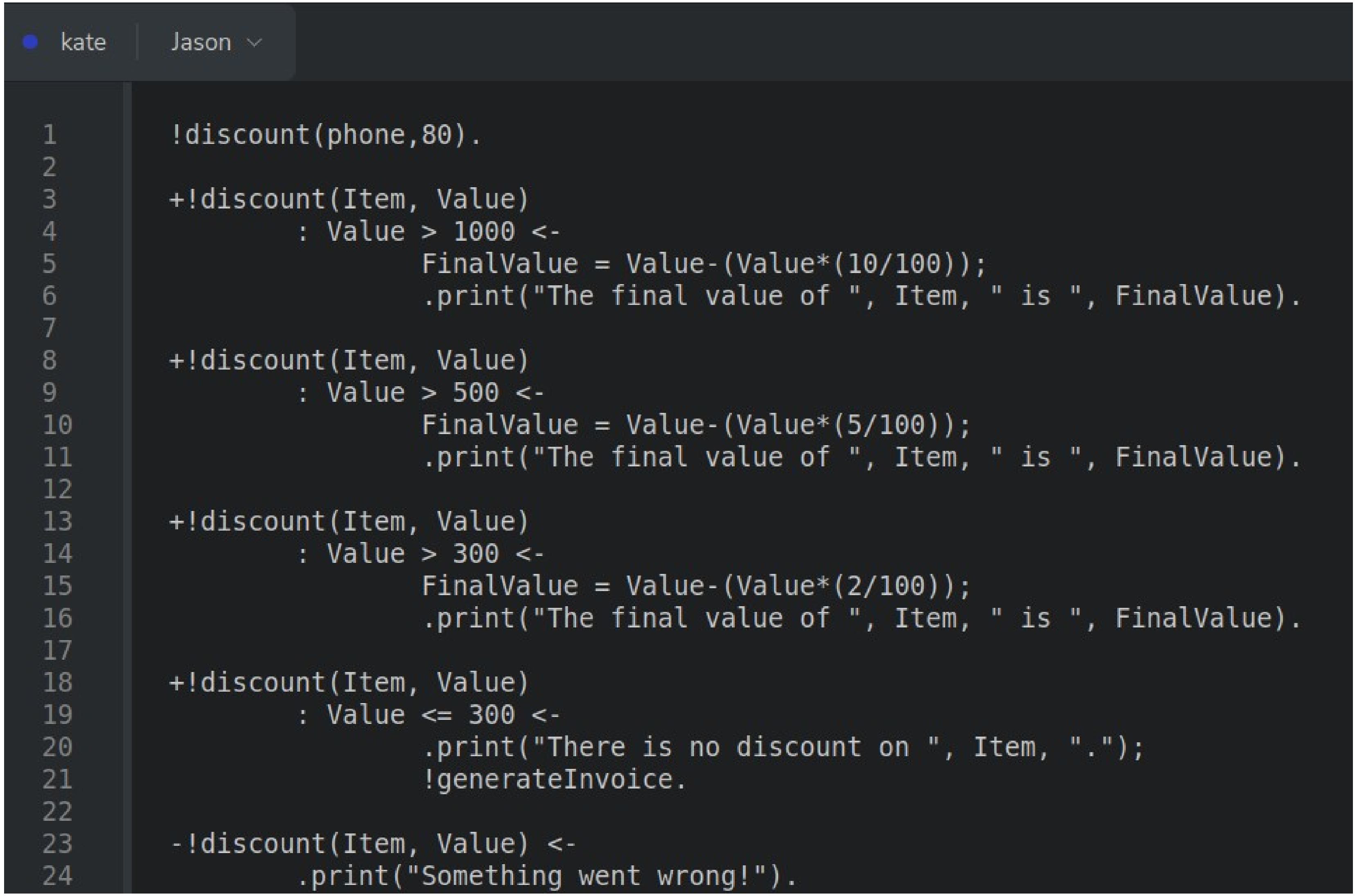
Plans: Deletion Achievement Goal



```
!discount(phone, 200).

+!discount(Item, Value)
        : Value > 1000 <-
                FinalValue = Value-(Value*(10/100));
                .print("The final value of ", Item, " is ", FinalValue).

+!discount(Item, Value)
        : Value > 500 <-
                FinalValue = Value-(Value*(5/100));
                .print("The final value of ", Item, " is ", FinalValue).

+!discount(Item, Value)
        : Value > 300 <-
                FinalValue = Value-(Value*(2/100));
                .print("The final value of ", Item, " is ", FinalValue).

+!discount(Item, Value)
        : Value <= 300 <-
                .print("There is no discount on ", Item, ".").

-!discount(Item, Value) <-
        .print("Something went wrong!").
```

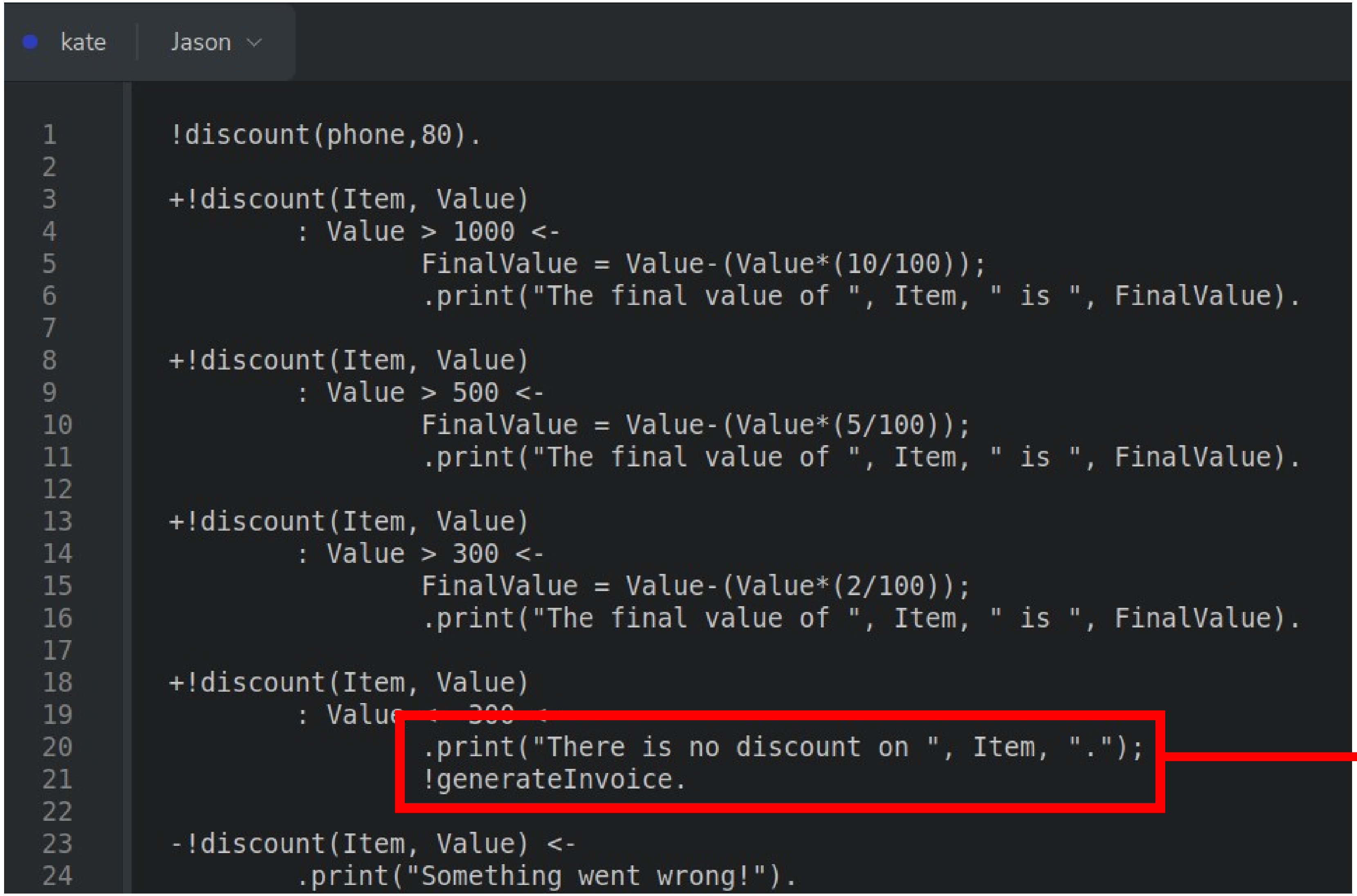
It only activates if the activated plan fails.

Plans: Deletion Achievement Goal



```
!discount(phone,80).

+!discount(Item, Value)
        : Value > 1000 <-
                FinalValue = Value-(Value*(10/100));
                .print("The final value of ", Item, " is ", FinalValue).

+!discount(Item, Value)
        : Value > 500 <-
                FinalValue = Value-(Value*(5/100));
                .print("The final value of ", Item, " is ", FinalValue).

+!discount(Item, Value)
        : Value > 300 <-
                FinalValue = Value-(Value*(2/100));
                .print("The final value of ", Item, " is ", FinalValue).

+!discount(Item, Value)
        : Value <= 300 <-
                .print("There is no discount on ", Item, ".");
                !generateInvoice.

-!discount(Item, Value) <-
        .print("Something went wrong!").
```

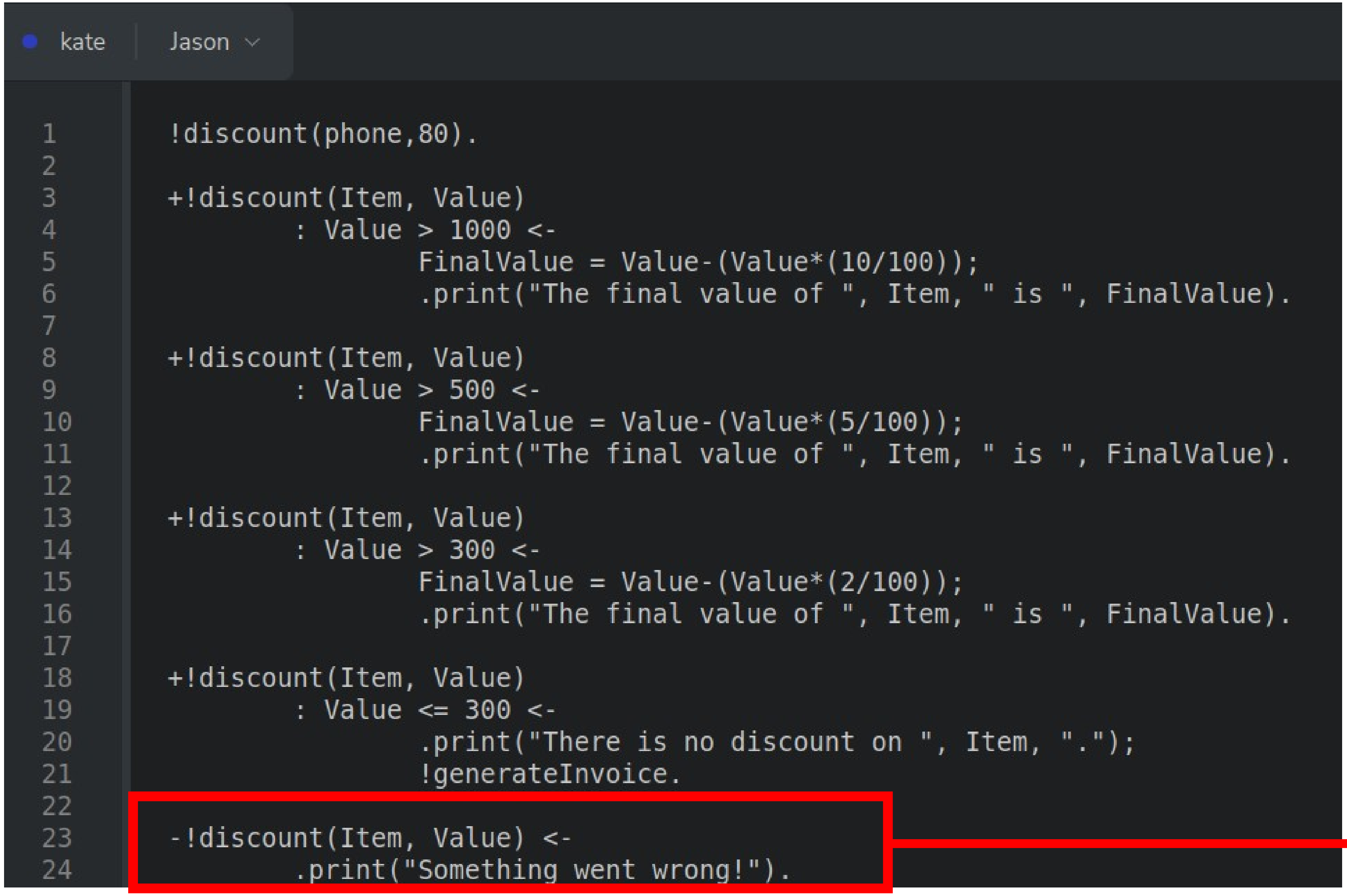
Plans: Deletion Achievement Goal



```
!discount(phone,80).

+!discount(Item, Value)
        : Value > 1000 <-
                FinalValue = Value-(Value*(10/100));
                .print("The final value of ", Item, " is ", FinalValue).

+!discount(Item, Value)
        : Value > 500 <-
                FinalValue = Value-(Value*(5/100));
                .print("The final value of ", Item, " is ", FinalValue).

+!discount(Item, Value)
        : Value > 300 <-
                FinalValue = Value-(Value*(2/100));
                .print("The final value of ", Item, " is ", FinalValue).

+!discount(Item, Value)
        : Value <= 300 <-
                .print("There is no discount on ", Item, ".");
                !generateInvoice.

-!discount(Item, Value) <-
        .print("Something went wrong!").
```

The agent tries to commit with generateInvoice, but it does not have a relevant plan.

Plans: Deletion Achievement Goal



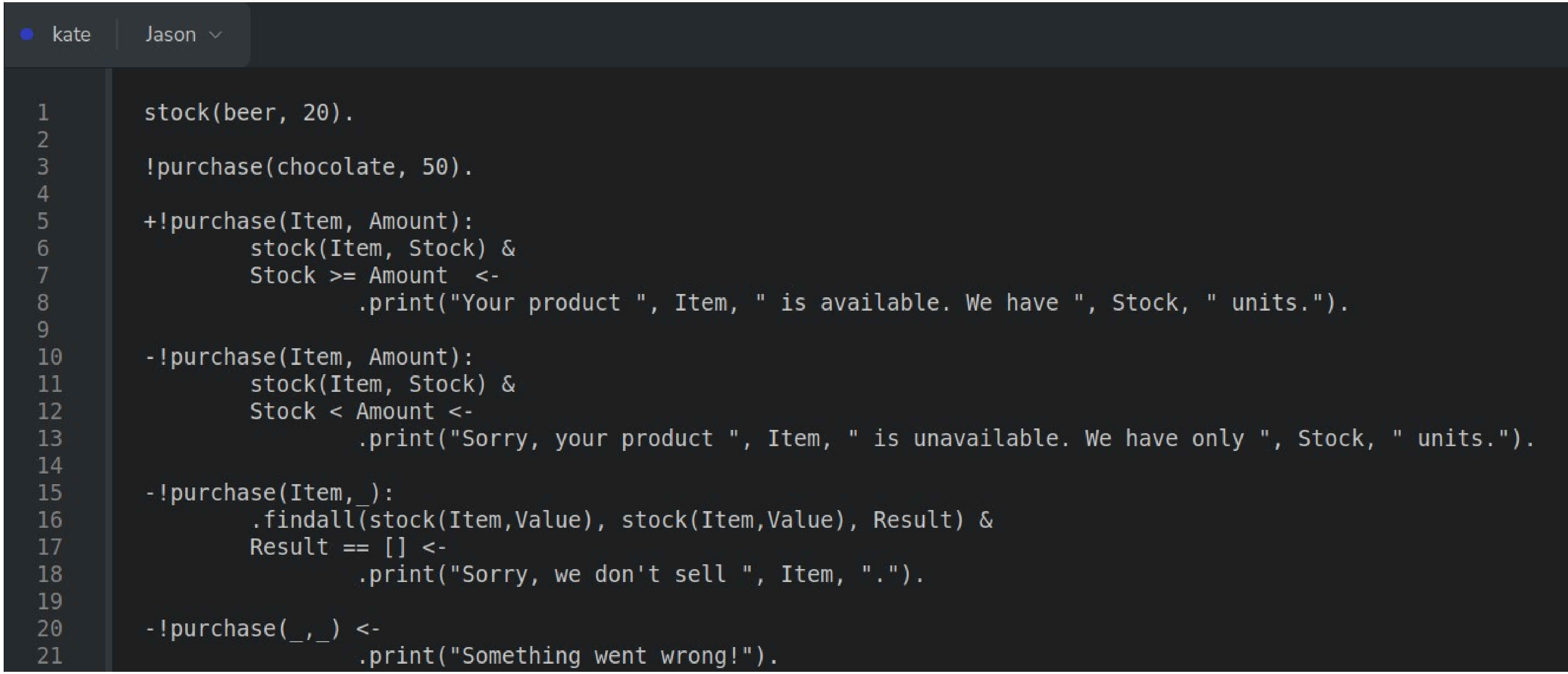
```
!discount(phone,80).

+!discount(Item, Value)
        : Value > 1000 <-
                FinalValue = Value-(Value*(10/100));
                .print("The final value of ", Item, " is ", FinalValue).

+!discount(Item, Value)
        : Value > 500 <-
                FinalValue = Value-(Value*(5/100));
                .print("The final value of ", Item, " is ", FinalValue).

+!discount(Item, Value)
        : Value > 300 <-
                FinalValue = Value-(Value*(2/100));
                .print("The final value of ", Item, " is ", FinalValue).

+!discount(Item, Value)
        : Value <= 300 <-
                .print("There is no discount on ", Item, ".");
                !generateInvoice.

-!discount(Item, Value) <-
        .print("Something went wrong!").
```

As the previous plan fails, the contingency plan is activated.

Plans: Deletion Achievement Goal

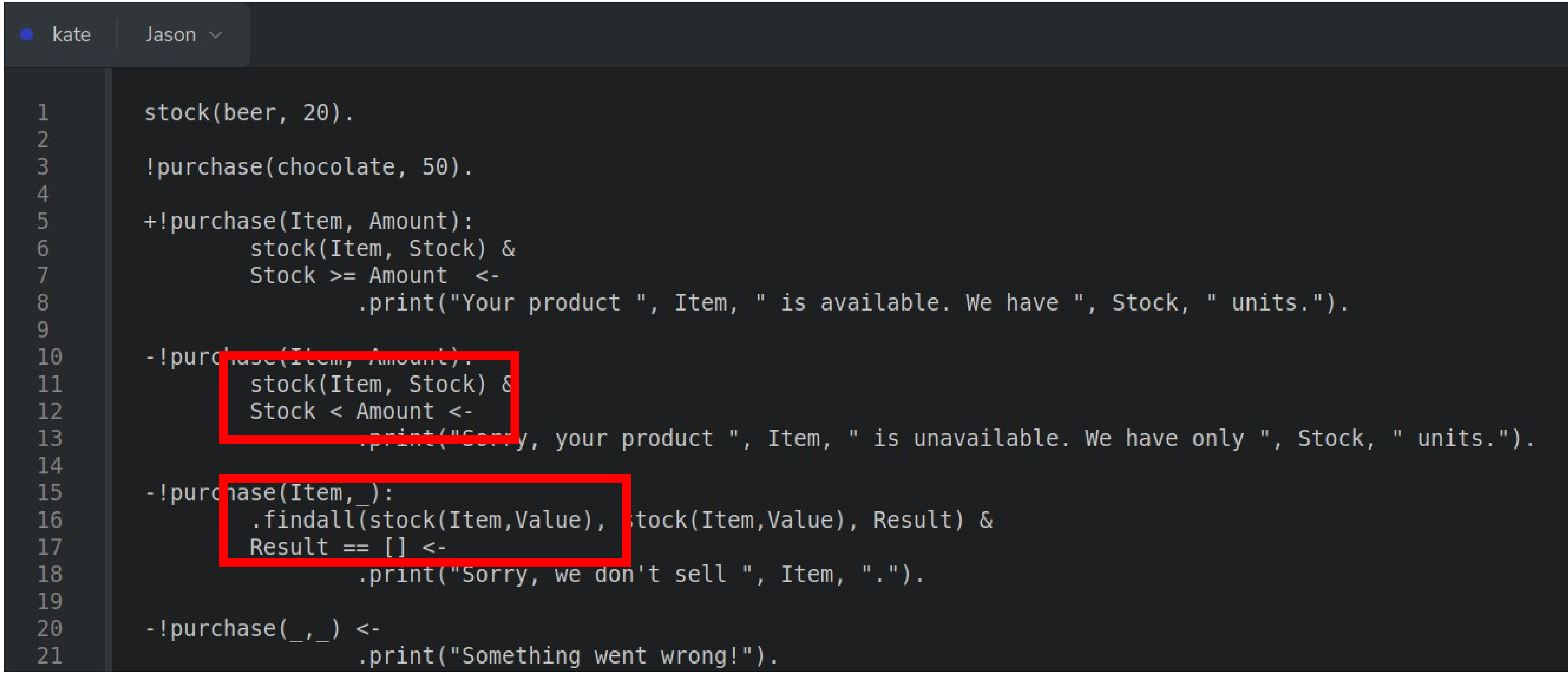
Deletion plans can also have a **context**.

Plans: Deletion Achievement Goal



```
stock(beer, 20).

!purchase(chocolate, 50).

+!purchase(Item, Amount):
        stock(Item, Stock) &
        Stock >= Amount  <-
                .print("Your product ", Item, " is available. We have ", Stock, " units.").

-!purchase(Item, Amount):
        stock(Item, Stock) &
        Stock < Amount <-
                .print("Sorry, your product ", Item, " is unavailable. We have only ", Stock, " units.").

-!purchase(Item,_):
        .findall(stock(Item,Value), stock(Item,Value), Result) &
        Result == [] <-
                .print("Sorry, we don't sell ", Item, ".").

-!purchase(_,_) <-
                .print("Something went wrong!").
```

Plans: Deletion Achievement Goal

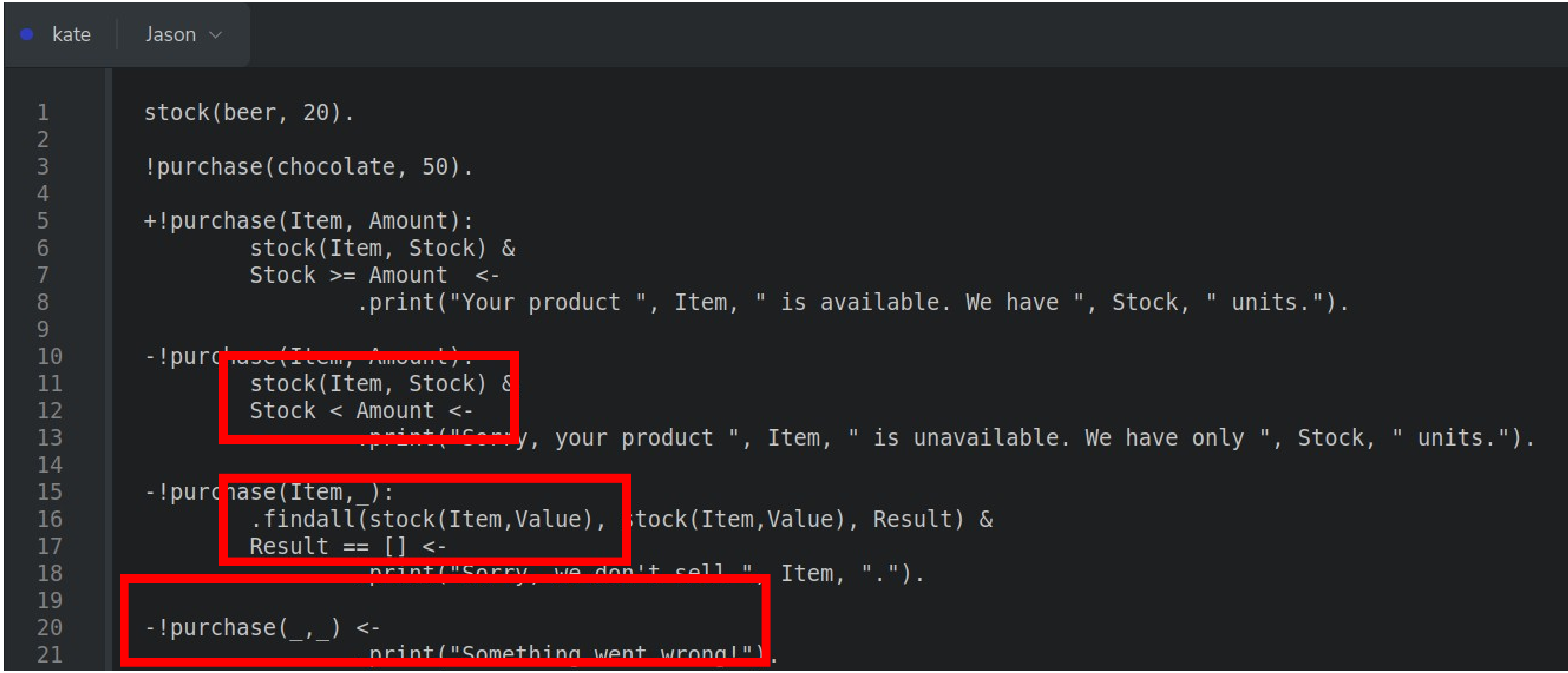


```
stock(beer, 20).

!purchase(chocolate, 50).

+!purchase(Item, Amount):
        stock(Item, Stock) &
        Stock >= Amount  <-
                .print("Your product ", Item, " is available. We have ", Stock, " units.").

-!purchase(Item, Amount):
        stock(Item, Stock) &
        Stock < Amount <-
                .print("Sorry, your product ", Item, " is unavailable. We have only ", Stock, " units.").

-!purchase(Item,_):
        .findall(stock(Item,Value), stock(Item,Value), Result) &
        Result == [] <-
                .print("Sorry, we don't sell ", Item, ".").

-!purchase(_,_) <-
                .print("Something went wrong!").
```

Deletion plans with context...

Plans: Deletion Achievement Goal

... and one without context.



```
stock(beer, 20).

!purchase(chocolate, 50).

+!purchase(Item, Amount):
        stock(Item, Stock) &
        Stock >= Amount  <-
                .print("Your product ", Item, " is available. We have ", Stock, " units.").

-!purchase(Item, Amount):
        stock(Item, Stock) &
        Stock < Amount <-
                .print("Sorry, your product ", Item, " is unavailable. We have only ", Stock, " units.").

-!purchase(Item,_):
        .findall(stock(Item,Value), stock(Item,Value), Result) &
        Result == [] <-
                .print("Sorry, we don't sell ", Item, ".").

-!purchase(_,_) <-
                .print("Something went wrong!").
```

Plans: Deletion Achievement Goal

+!event[source(type)]: context ← (…).

Plans: Deletion Achievement Goal

```
+!event[source(type)]: context ← (…).
+!event[source(type)]: context ← (…).
```

Plans: Deletion Achievement Goal

+!event[source(type)]: context ← (…).

+!event[source(type)]: context ← (…).

⋮

+!event[source(type)]: context ← (…).

Plans: Deletion Achievement Goal

```
+!event[source(type)]: context ← (…).
+!event[source(type)]: context ← (…).
                    .
                    .
                    .
+!event[source(type)]: context ← (…).
 -!event[source(type)]: context ← (…).
```

Plans: Deletion Achievement Goal

```
+!event[source(type)]: context ← (...).
+!event[source(type)]: context ← (...).
                 .
                 .
                 .
+!event[source(type)]: context ← (...).
 -!event[source(type)]: context ← (...).
 -!event[source(type)]: context ← (...).
```

Plans: Deletion Achievement Goal

```
+!event[source(type)]: context ← (…).
+!event[source(type)]: context ← (…).
                    ⋮
+!event[source(type)]: context ← (…).
 -!event[source(type)]: context ← (…).
 -!event[source(type)]: context ← (…).
                    ⋮
 -!event[source(type)]: context ← (…).
```

Plans: Test Goal

```
{+|-}?event[source(type)]:
    context ←
        action 1;
        action 2;
        action n.
```

Plans: Test Goal

```
{+|-}?event[source(type)]:
    context ←
        action 1;
        action 2;
        action n.
```

defines if it is an addition (+) or a deletion (-) plan.

Plans: Test Goal

specifies the test goal plan.

```
{+|-}?event[source(type)]:
     context ←
          action 1;
          action 2;
          action n.
```

Plans: Addition Test Goal

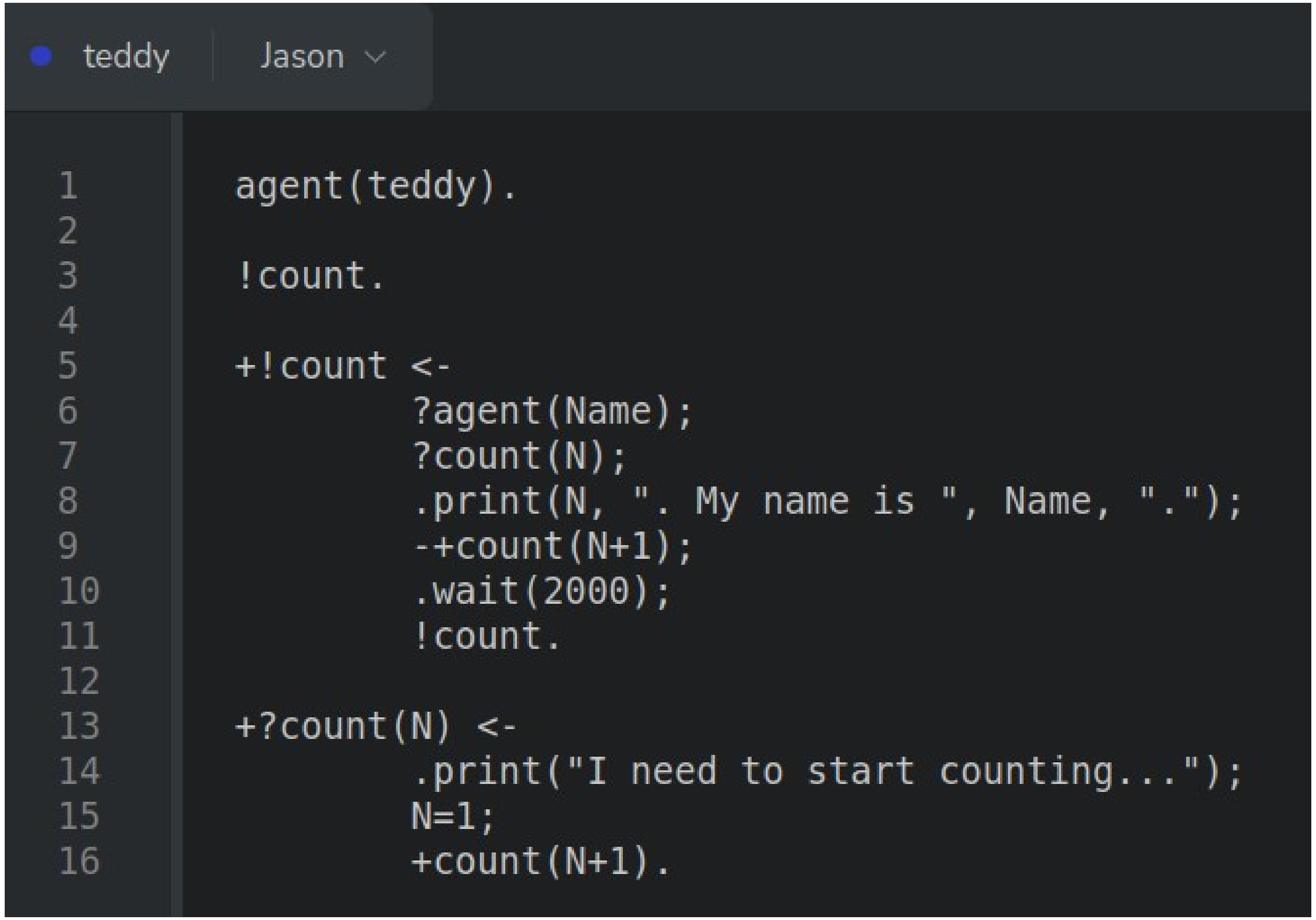
```
+?event[source(type)]:
    context ←
        action 1;
        action 2;
        action n.
```

Plans: Addition Test Goal

```
+?event[source(type)]:
    context ←
        action 1;
        action 2;
        action n.
```

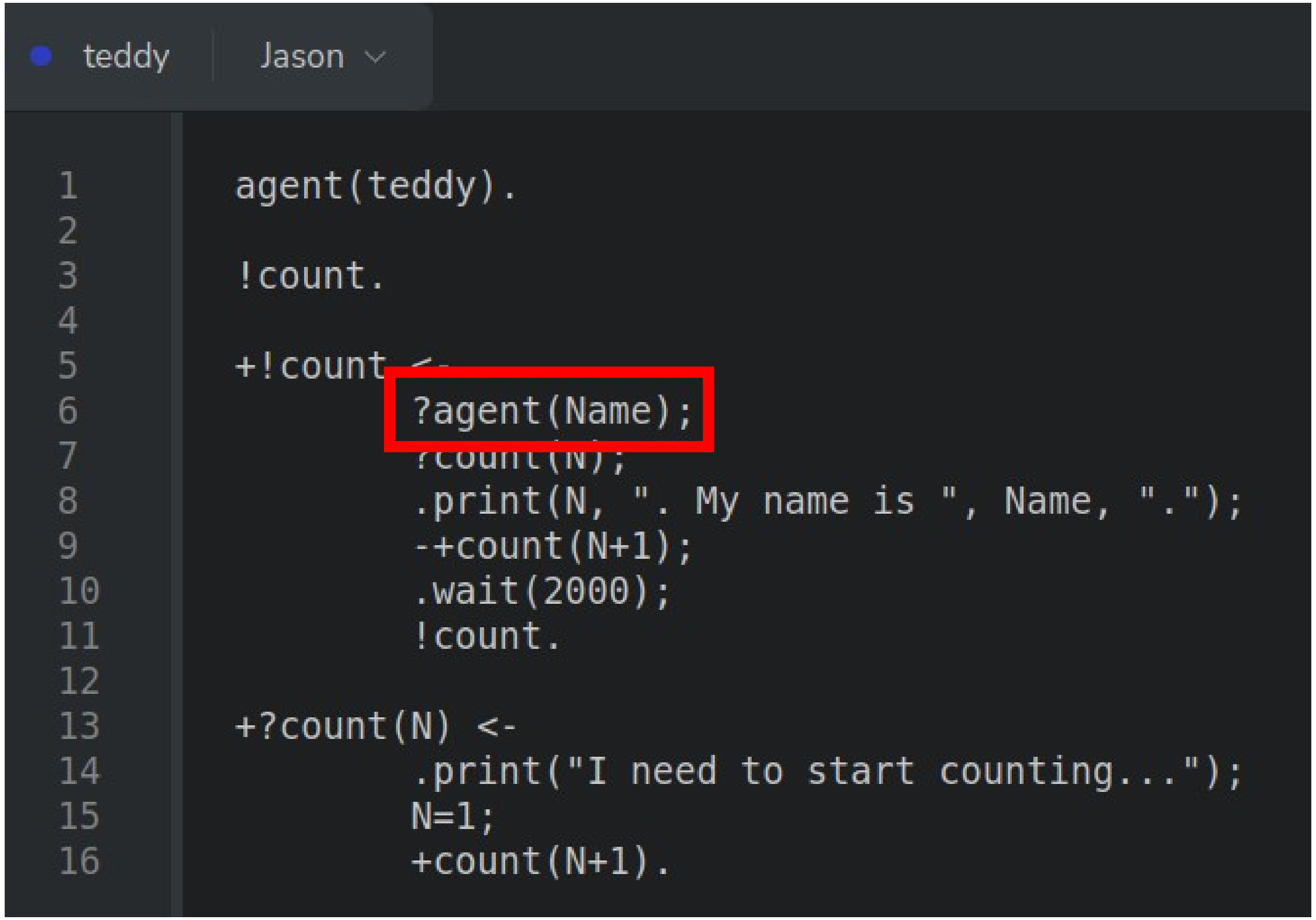
defines an addition plan.

Plans: Addition Test Goal



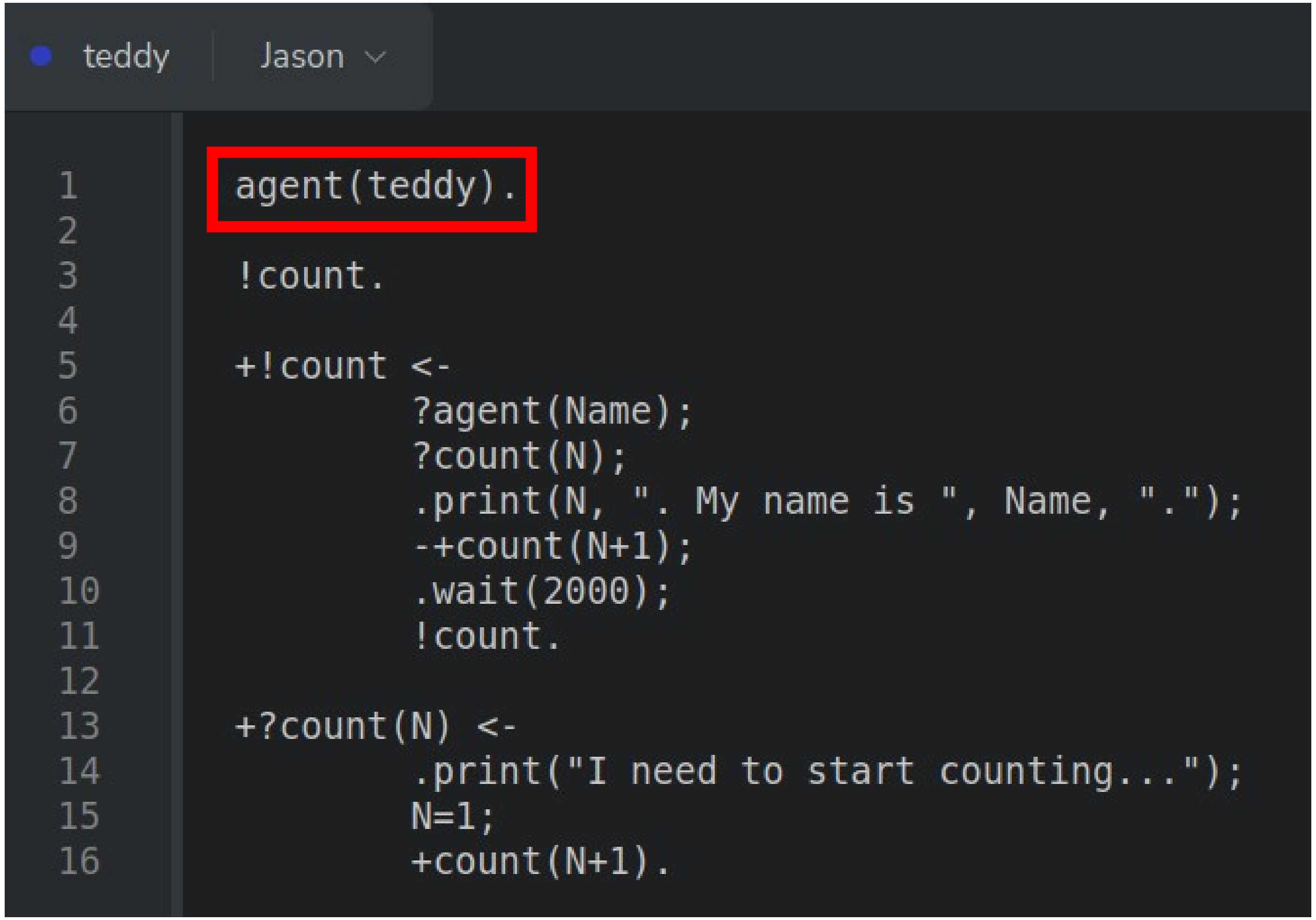
```
agent(teddy).

!count.

+!count <-
        ?agent(Name);
        ?count(N);
        .print(N, ". My name is ", Name, ".");
        -+count(N+1);
        .wait(2000);
        !count.

+?count(N) <-
        .print("I need to start counting...");
        N=1;
        +count(N+1).
```

Plans: Addition Test Goal



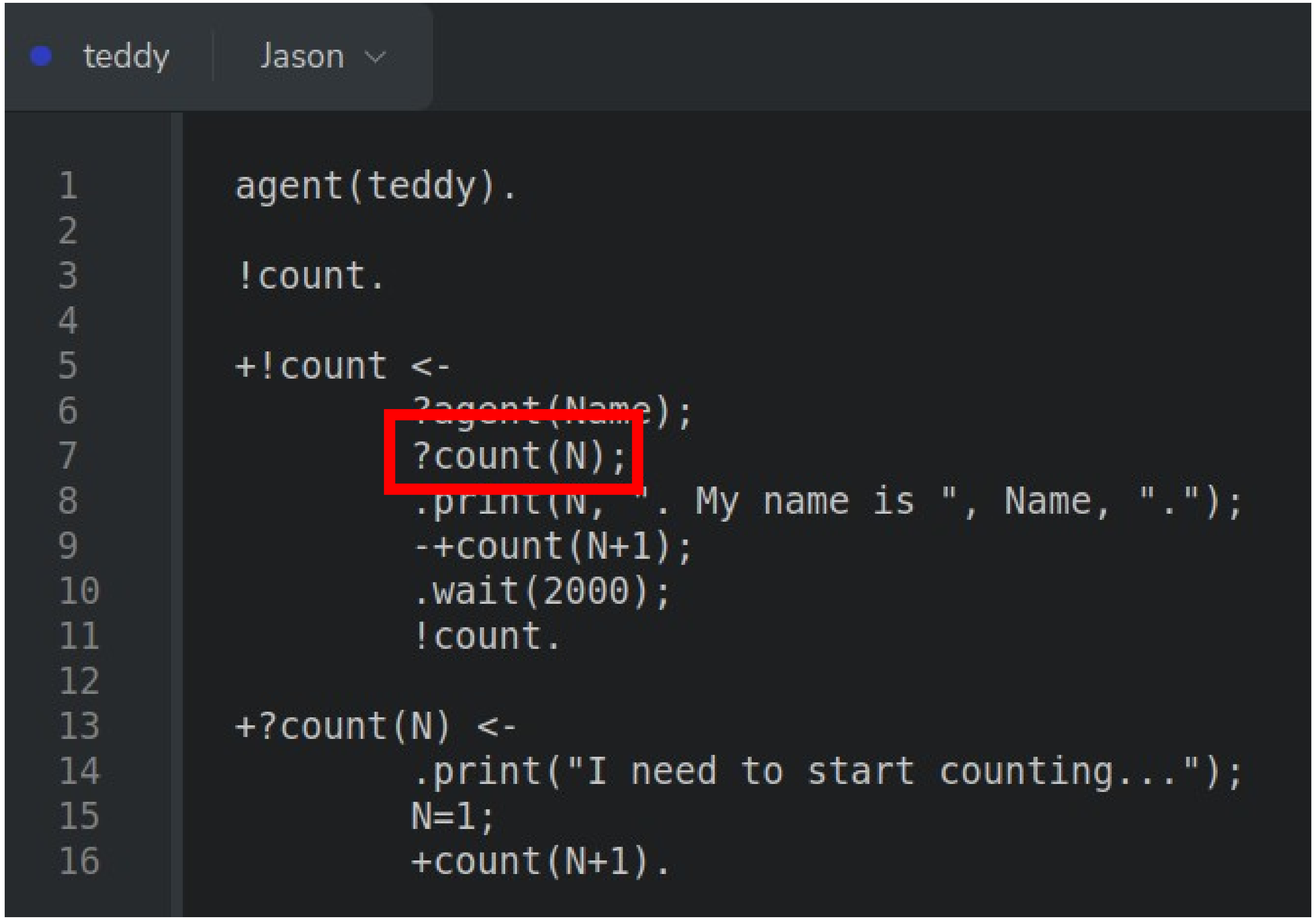
```
agent(teddy).

!count.

+!count <-
        ?agent(Name);
        ?count(N);
        .print(N, ". My name is ", Name, ".");
        -+count(N+1);
        .wait(2000);
        !count.

+?count(N) <-
        .print("I need to start counting...");
        N=1;
        +count(N+1).
```

Plans: Addition Test Goal



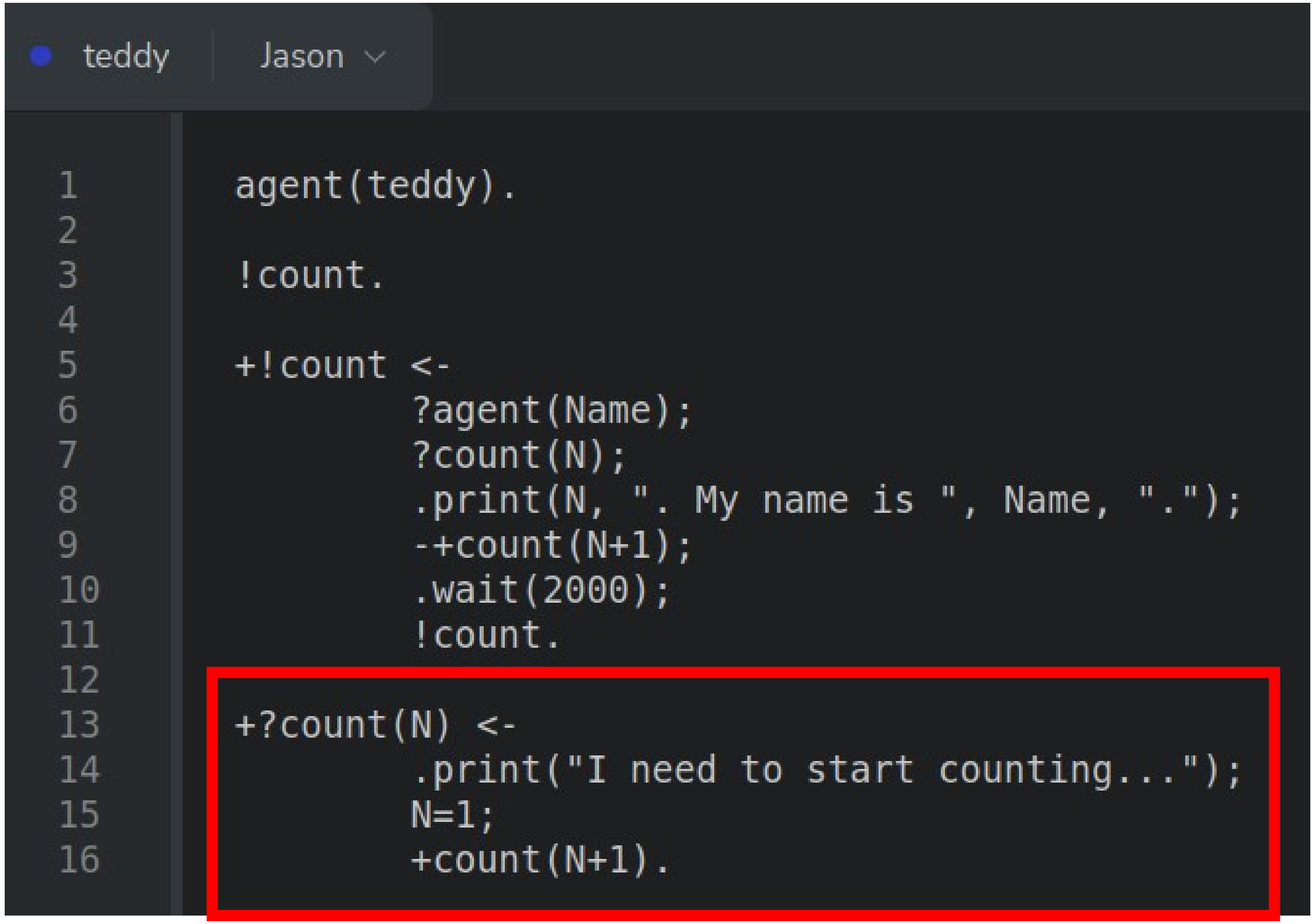
```
agent(teddy).

!count.

+!count <-
        ?agent(Name);
        ?count(N);
        .print(N, ". My name is ", Name, ".");
        -+count(N+1);
        .wait(2000);
        !count.

+?count(N) <-
        .print("I need to start counting...");
        N=1;
        +count(N+1).
```

Plans: Addition Test Goal



```
agent(teddy).

!count.

+!count <-
        ?agent(Name);
        ?count(N);
        .print(N, ". My name is ", Name, ".");
        -+count(N+1);
        .wait(2000);
        !count.

+?count(N) <-
        .print("I need to start counting...");
        N=1;
        +count(N+1).
```

Plans: Addition Test Goal



```
agent(teddy).

!count.

+!count <-
        ?agent(Name);
        ?count(N);
        .print(N, ". My name is ", Name, ".");
        -+count(N+1);
        .wait(2000);
        !count.

+?count(N) <-
        .print("I need to start counting...");
        N=1;
        +count(N+1).
```

Plans: Deletion Test Goal

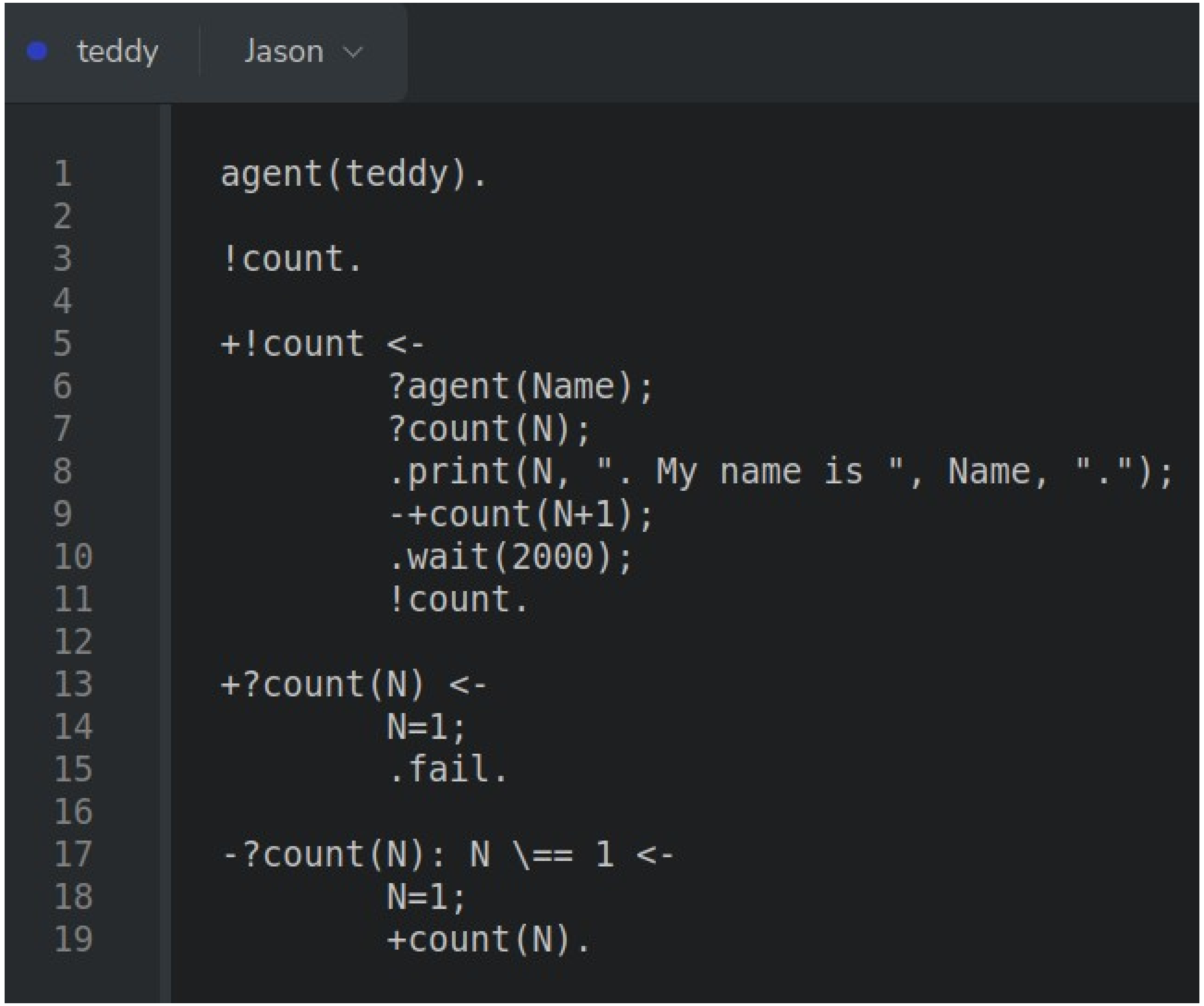
```
-?event[source(type)]:
    context ←
        action 1;
        action 2;
        action n.
```

Plans: Deletion Test Goal

```
-?event[source(type)]:
     context ←
          action 1;
          action 2;
          action n.
```

defines a deletion plan.

Plans: Deletion Test Goal



```
agent(teddy).

!count.

+!count <-
        ?agent(Name);
        ?count(N);
        .print(N, ". My name is ", Name, ".");
        -+count(N+1);
        .wait(2000);
        !count.

+?count(N) <-
        N=1;
        .fail.

-?count(N): N \== 1 <-
        N=1;
        +count(N).
```

Plans: Belief

```
{+|-}event[source(type)]:
    context ←
        action 1;
        action 2;
        action n.
```

Plans: Belief

```
{+|-}event[source(type)]:
    context ←
        action 1;
        action 2;
        action n.
```

defines if it is an addition (+) or a deletion (-) plan.

Plans: Additi

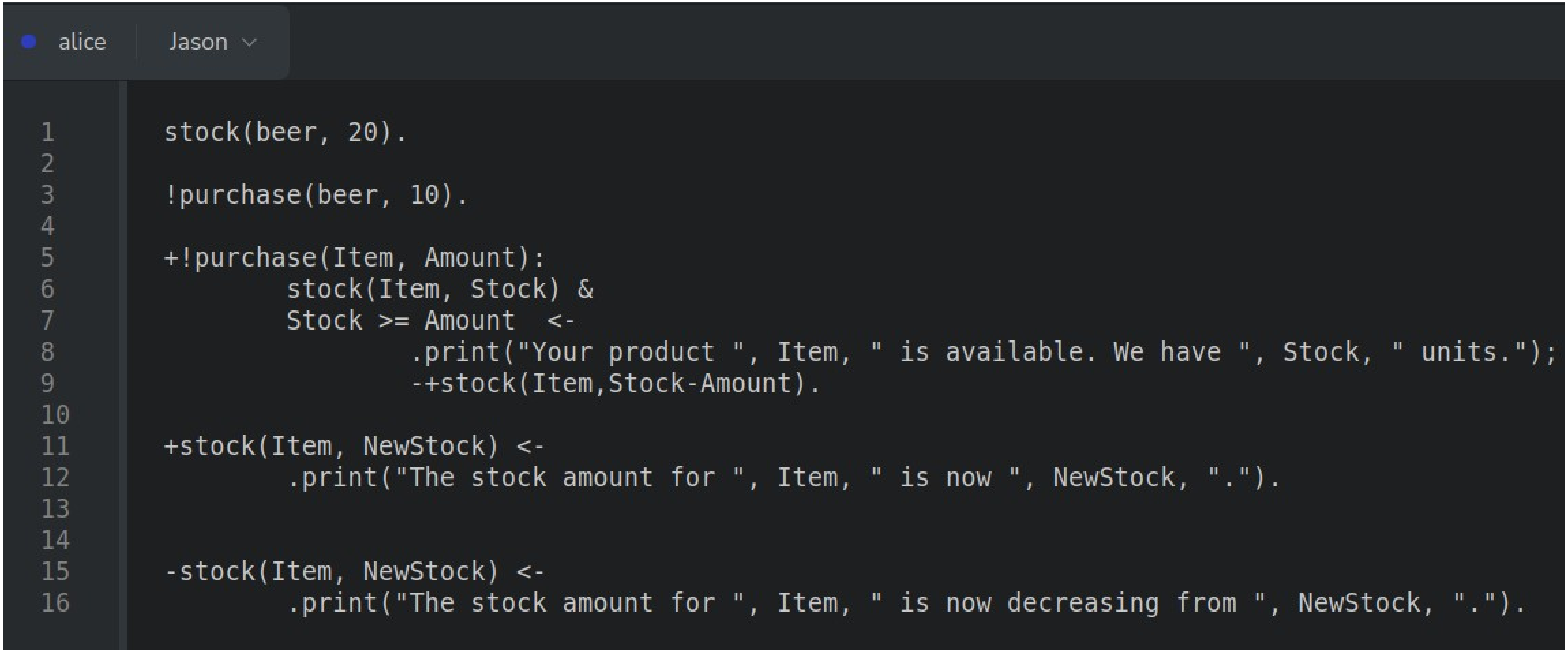
```
+event[source(type)]:
    context ←
        action 1;
        action 2;
        action n.
```

Plans: Addition Belief Goal

```
+event[source(type)]:
    context ←
        action 1;
        action 2;
        action n.
```

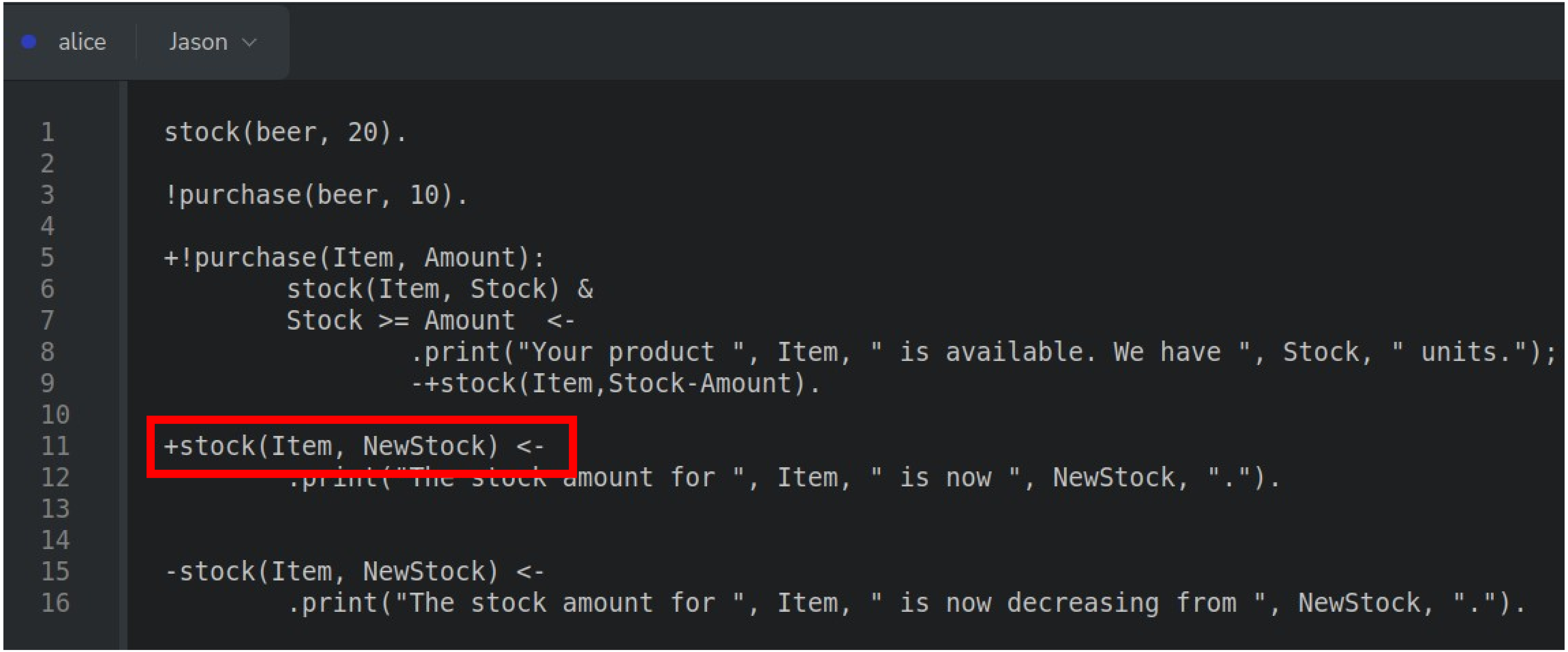
defines an addition plan.

Plans: Addition Belief Goal



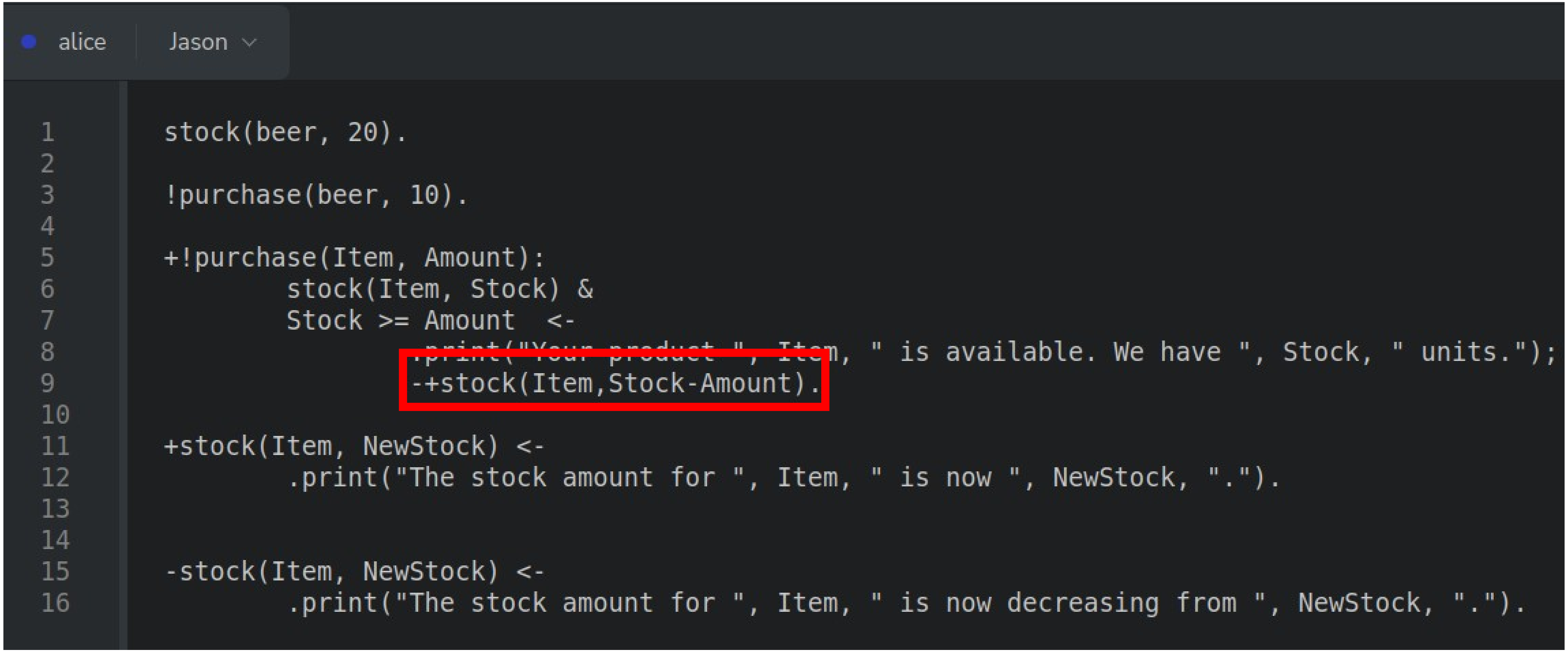
```
stock(beer, 20).

!purchase(beer, 10).

+!purchase(Item, Amount):
        stock(Item, Stock) &
        Stock >= Amount  <-
                .print("Your product ", Item, " is available. We have ", Stock, " units.");
                -+stock(Item,Stock-Amount).

+stock(Item, NewStock) <-
        .print("The stock amount for ", Item, " is now ", NewStock, ".").


-stock(Item, NewStock) <-
        .print("The stock amount for ", Item, " is now decreasing from ", NewStock, ".").
```

Plans: Addition Belief Goal



```
stock(beer, 20).

!purchase(beer, 10).

+!purchase(Item, Amount):
        stock(Item, Stock) &
        Stock >= Amount  <-
                .print("Your product ", Item, " is available. We have ", Stock, " units.");
                -+stock(Item,Stock-Amount).

+stock(Item, NewStock) <-
        .print("The stock amount for ", Item, " is now ", NewStock, ".").


-stock(Item, NewStock) <-
        .print("The stock amount for ", Item, " is now decreasing from ", NewStock, ".").
```

Plans: Addition Belief Goal



```
stock(beer, 20).

!purchase(beer, 10).

+!purchase(Item, Amount):
        stock(Item, Stock) &
        Stock >= Amount  <-
                .print("Your product ", Item, " is available. We have ", Stock, " units.");
                -+stock(Item,Stock-Amount).

+stock(Item, NewStock) <-
        .print("The stock amount for ", Item, " is now ", NewStock, ".").


-stock(Item, NewStock) <-
        .print("The stock amount for ", Item, " is now decreasing from ", NewStock, ".").
```

Plans: Deleti

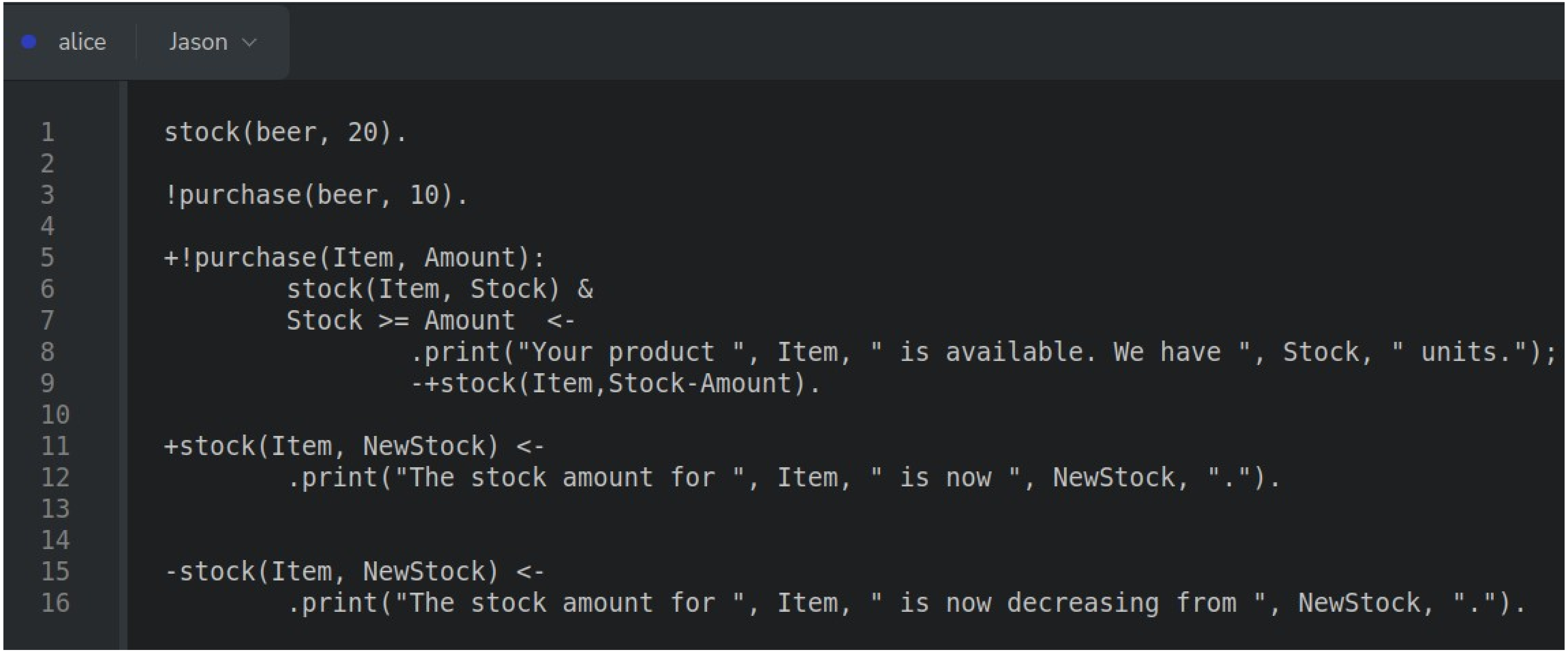
```
-event[source(type)]:
    context ←
        action 1;
        action 2;
        action n.
```

Plans: Deletion Belief Goal

```
-event[source(type)]:
    context ←
        action 1;
        action 2;
        action n.
```

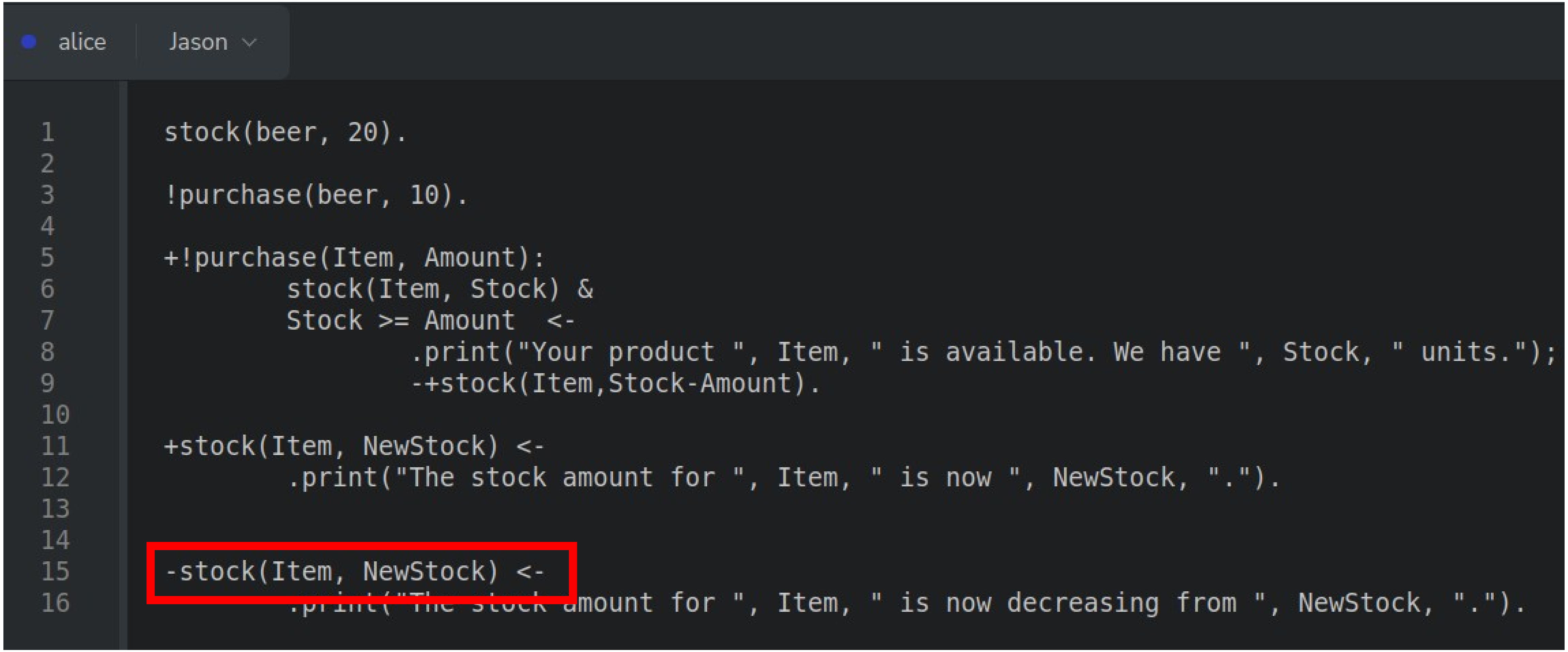
defines a deletion plan.

Plans: Deletion Belief Goal



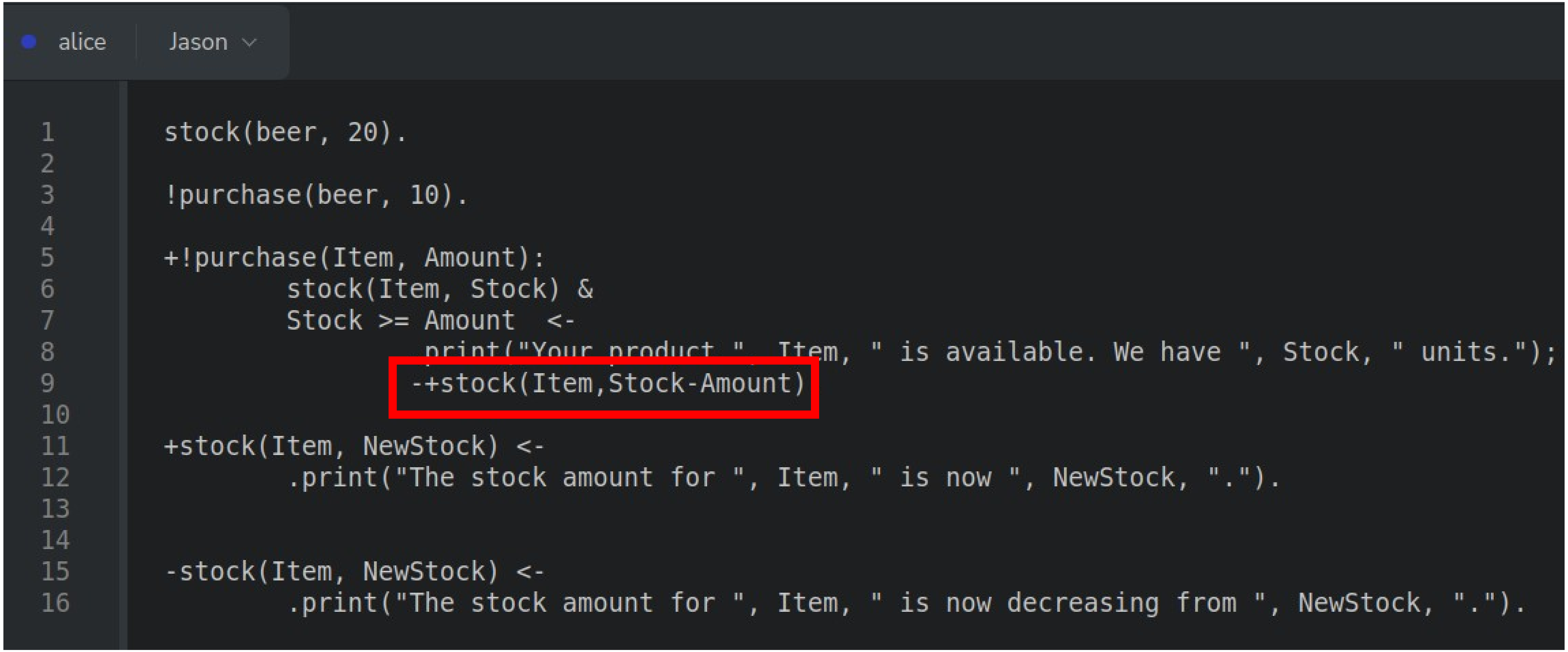
```
stock(beer, 20).

!purchase(beer, 10).

+!purchase(Item, Amount):
        stock(Item, Stock) &
        Stock >= Amount  <-
                .print("Your product ", Item, " is available. We have ", Stock, " units.");
                -+stock(Item,Stock-Amount).

+stock(Item, NewStock) <-
        .print("The stock amount for ", Item, " is now ", NewStock, ".").


-stock(Item, NewStock) <-
        .print("The stock amount for ", Item, " is now decreasing from ", NewStock, ".").
```

Plans: Deletion Belief Goal



```
stock(beer, 20).

!purchase(beer, 10).

+!purchase(Item, Amount):
        stock(Item, Stock) &
        Stock >= Amount  <-
                .print("Your product ", Item, " is available. We have ", Stock, " units.");
                -+stock(Item,Stock-Amount).

+stock(Item, NewStock) <-
        .print("The stock amount for ", Item, " is now ", NewStock, ".").


-stock(Item, NewStock) <-
        .print("The stock amount for ", Item, " is now decreasing from ", NewStock, ".").
```

Plans: Deletion Belief Goal



```
stock(beer, 20).

!purchase(beer, 10).

+!purchase(Item, Amount):
        stock(Item, Stock) &
        Stock >= Amount  <-
                .print("Your product ", Item, " is available. We have ", Stock, " units.");
                -+stock(Item,Stock-Amount)

+stock(Item, NewStock) <-
        .print("The stock amount for ", Item, " is now ", NewStock, ".").


-stock(Item, NewStock) <-
        .print("The stock amount for ", Item, " is now decreasing from ", NewStock, ".").
```

Agradecimentos

OBRIGADO!

pantoja@cefet-rj.br
nilson.lazarin@cefet-rj.br