

Jess

- ▶ Production Rule Language
 - ▶ Fact
 - ▶ Rule
- ▶ Procedural Language
- ▶ Inference Engine
 - ▶ Match
 - ▶ Choose
 - ▶ Execute

Basic Jess Info

- ▶ To run type `java jess.Main` The Jess prompt will then appear.
- ▶ You must have `/usr/local/Jess` in your `CLASSPATH`
- ▶ You can find some examples under `/usr/local/Jess`

Facts

```
Jess>
```

```
Jess> (assert (today is sunday))
```

```
<Fact-0>
```

```
Jess> (assert (weather is warm))
```

```
<Fact-1>
```

```
Jess> (facts)
```

```
f-0 (today is sunday)
```

```
f-0 (weather is warm)
```

```
Jess> (clear)
```

```
Jess> (facts>
```

```
Jess>(quit)
```

Rules

Rules: Pattern

```
(defrule <rule-name>
  <optional comment>
  <optional declaration>
  <premise_1>
  .....
  <premise_m>
  =>
  <action_1>
  .....
  <action_n>
)
```

Rules: Example

```
(defrule chores
  "Things to do on Sunday"
  (salience 10)
  (today is sunday)
  (weather is warm)
  =>
  (assert (wash car))
  (assert (chop wood))
)
```

Rules: Example

```
(defrule fun
  "Better things to do on Sunday"
  (salience 100)
  (today is sunday)
  (weather is warm)
  =>
  (assert (drink beer))
  (assert (play guitar))
)
```

Variables

```
(defrule pick-a-chore
  "allocating chores to days"
  (today is ?day)
  (chore is ?job)
  =>
  (assert (do ?job on ?day))
)
```


Variables (cont)

```
(defrule drop-a-chore
  "allocating chores to days"
  (today is ?day)
  ?chore <- (do ?job on ?day)
  =>
  (retract ?chore)
)
```

Running Examples

```
Jess> (assert (human Socrates))  
Jess> (defrule mortal (human ?X) => (mortal ?X))  
Jess> (watch facts)  
Jess (run)  
==> f-1 (MAIN: mortal Socrates)
```

Running Jess Alternatives

- ▶ Load From File

```
Jess> (batch socrates.clp)
      (watch facts)
      (run)
```

- ▶ From Command Line

```
java jess.Main socrates.clp
```