

Doing Knowledge Representation and Reasoning with PowerLoom.

You get PowerLoom here: <http://www.isi.edu/isd/LOOM/PowerLoom/>

Create a new, empty, module to work in and specify representation language syntax:

```
(defmodule "PL-USER/FAMILY")  
(in-module "FAMILY")  
(reset-features)  
(in-dialect KIF)
```

Save your module to disk (shows up in your powerloom "kbs" directory), and loading it back later:

```
(save-module "FAMILY" "FAMILY.PLM")  
  
(load "FAMILY.PLM")  
(in-module "FAMILY") ;;; If not already in this module
```

Defining a basic type/class predicate (a unary relation) called a "concept" in Powerloom:

```
(defconcept Person(?p))  
(defconcept Male(?p))  
(defconcept Female(?p))
```

Basic TELLing and ASKing in Powerloom:

```
(assert (Male John))  
(assert (Female Mary))  
  
(ask (Male John))  
(ask (Female Mary))
```

Adding FOL axioms:

```
(assert (forall (?x) (=> (Male ?x) (Person ?x)))) ;; Being a male implies you are a person  
;; do the same for females
```

Asking for possible substitutions:

```
(retrieve (Person ?p)) ;;; Returns one possible substitutions for ?p if it exists  
(retrieve all (Person ?p)) ;;; Returns all possible substitutions for ?p
```

The Open-World semantics of Powerloom:

```
(ask (Male Mary)) ;;; Should be unknown since it wouldn't conflict with the KB  
(assert (forall (?p) (<=> (Male ?p) (not (Female ?p))))) ;; Being a male implies you are not a female  
;; do the same for females  
(ask (Male Mary))
```

Defining a regular relation predicate in Powerloom:

```
(defrelation BrotherOf ((?p1 Male) (?p2 Person)))
(assert (BrotherOf John Mary))
(assert (Person Olaf))
(assert (BrotherOf Olaf Mary))

(retrieve all (BrotherOf ?x Mary))
(ask (Male Olaf))

;; Create a new predicate called ParentOf
```

Defining a regular function in Powerloom:

```
(deffunction GetFather ((?p1 Person)) :-> (?p2 Person))

(assert (= (GetFather Mary) Zod))
```

A new axiom that uses a function and equivalence:

```
(assert (<=> (= (GetFather ?c) ?f) (and (Male ?f) (ParentOf ?f ?c))))

;; Answer: Is Zod Male? Is he Mary's parent?
```

Now you are on your own.... add more family relations like *SisterOf*, *GrandmotherOf*, *AreSiblings*, *UncleOf*, *SonOf*, *AuntOf*, *DaughterOf*, *ChildOf*, ... and try answering questions like „Is X a sibling of Y?“, „who are X's grandmothers?“, „who are X's uncles?“, „Does X's mother's mother have a male child?“...

Helpful commands:

```
(all-facts-of Zod) ;;; prints out all known facts about Zod
(help assert) ;;; prints out help text about "assert"
(demo) ;;; leads you through various PowerLoom demonstrations
```