
Prolog Programming Assignment #2: A Favorite Pokemon KB plus Simple List Processing Exercises

Learning Abstract

The target of this assignment was to familiarize the students with Prolog and list processing. Most of this assignment involves interacting with Pokemon KB which is provided. There are a few list-processing exercises and tasks in the end. It helped me understand the basics of list processing and Prolog.

Task 1 - Pokemon

Part 1: Initial Pokemon KB

```
% -----  
% -----  
% --- File: pokemon.pro  
% --- Line: Just a few facts about pokemon  
% -----  
  
% -----  
% --- cen(P) :: Pokemon P was "creatio ex nihilo"  
  
cen(pikachu).  
cen(bulbasaur).  
cen(caterpie).  
cen(charmander).  
cen(vulpix).  
cen(poliwag).  
cen(squirtle).  
cen(staryu).  
  
% -----  
% --- evolves(P,Q) :: Pokemon P directly evolves to pokemon Q  
  
evolves(pikachu,raichu).  
evolves(bulbasaur,ivysaur).  
evolves(ivysaur,venusaur).  
evolves(caterpie,metapod).  
evolves(metapod,butterfree).  
evolves(charmander,charmeleon).  
evolves(charmeleon,charizard).  
evolves(vulpix,ninetails).  
evolves(poliwag,poliwhirl).  
evolves(poliwhirl,poliwrath).  
evolves(squirtle,wartortle).
```

evolves(wartortle,blastoise).
evolves(staryu,starmie).

% -----
% --- pokemon(name(N),T,hp(H),attach(A,D)) :: There is a pokemon with
% --- name N, type T, hit point value H, and attach named A that does
% --- damage D.

pokemon(name(pikachu), electric, hp(60), attack(gnaw, 10)).
pokemon(name(raichu), electric, hp(90), attack(thunder-shock, 90)).

pokemon(name(bulbasaur), grass, hp(40), attack(leech-seed, 20)).
pokemon(name(ivysaur), grass, hp(60), attack(vine-whip, 30)).
pokemon(name(venusaur), grass, hp(140), attack(poison-powder, 70)).

pokemon(name(caterpie), grass, hp(50), attack(gnaw, 20)).
pokemon(name(metapod), grass, hp(70), attack(stun-spore, 20)).
pokemon(name(butterfree), grass, hp(130), attack(whirlwind, 80)).

pokemon(name(charmander), fire, hp(50), attack(scratch, 10)).
pokemon(name(charmeleon), fire, hp(80), attack(slash, 50)).
pokemon(name(charizard), fire, hp(170), attack(royal-blaze, 100)).

pokemon(name(vulpix), fire, hp(60), attack(confuse-ray, 20)).
pokemon(name(ninetails), fire, hp(100), attack(fire-blast, 120)).

pokemon(name(poliwag), water, hp(60), attack(water-gun, 30)).
pokemon(name(poliwhirl), water, hp(80), attack(amnesia, 30)).
pokemon(name(poliwrath), water, hp(140), attack(dashing-punch, 50)).

pokemon(name(squirtle), water, hp(40), attack(bubble, 10)).
pokemon(name(wartortle), water, hp(80), attack(waterfall, 60)).
pokemon(name(blastoise), water, hp(140), attack(hydro-pump, 60)).

pokemon(name(staryu), water, hp(40), attack(slap, 20)).
pokemon(name(starmie), water, hp(60), attack(star-freeze, 20)).

Part 2: Interaction demo with the Initial KB

```
?- consult('/Users/LENOVO/Prolog/pokemon.pro').
true.

?- cen(pikachu).
true.

?- cen(raichu).
false.

?- cen(Names).
Names = pikachu ;
Names = bulbasaur ;
Names = caterpie ;
Names = charmander ;
Names = vulpix ;
Names = poliwag ;
Names = squirtle ;
Names = staryu.

?- cen(Names),write(Names),nl,fail.
pikachu
bulbasaur
caterpie
charmander
vulpix
poliwag
squirtle
staryu
false.

?- evolves(squirtle,wartortle).
true.

?- evolves(wartortle,squirtle).
false.

?- evolves(squirtle,blastoise).
false.

?- evolves(X,Y),evolves(Y,Z).
X = bulbasaur,
Y = ivysaur,
Z = venusaur ;
X = caterpie,
Y = metapod,
Z = butterfree ;
X = charmander,
Y = charmeleon,
Z = charizard ;
X = poliwag,
Y = poliwhirl,
Z = poliwrath ;
X = squirtle,
Y = wartortle,
Z = blastoise ;
false.
```

```

?- evolves(X,Y),evolves(Y,Z),write(X),write('->'),write(Z), nl, fail.
bulbasaur->venusaur
caterpie->butterfree
charmander->charizard
poliwag->poliwrath
squirtle->blastoise
false.

?- pokemon(name(X),_,_,_),write(X),nl,fail.
pikachu
raichu
bulbasaur
ivysaur
venusaur
caterpie
metapod
butterfree
charmander
charmeleon
charizard
vulpix
ninetails
poliwag
poliwhirl
poliwrath
squirtle
wartortle
blastoise
staryu
starmie
false.

?- pokemon(name(X),fire,_,_),write(X),nl,fail.
charmander
charmeleon
charizard
vulpix
ninetails
false.

```

```

?- pokemon(name(X),Y,_,_),write('nks(name('),write(X),write('),kind('),write(Y),write(')')',nl,fail.
nks(name(pikachu),kind(electric))
nks(name(raichu),kind(electric))
nks(name(bulbasaur),kind(grass))
nks(name(ivysaur),kind(grass))
nks(name(venusaur),kind(grass))
nks(name(caterpie),kind(grass))
nks(name(metapod),kind(grass))
nks(name(butterfree),kind(grass))
nks(name(charmander),kind(fire))
nks(name(charmeleon),kind(fire))
nks(name(charizard),kind(fire))
nks(name(vulpix),kind(fire))
nks(name(ninetails),kind(fire))
nks(name(poliwag),kind(water))
nks(name(poliwhirl),kind(water))
nks(name(poliwrath),kind(water))
nks(name(squirtle),kind(water))
nks(name(wartortle),kind(water))
nks(name(blastoise),kind(water))
nks(name(staryu),kind(water))
nks(name(starmie),kind(water))
false.

?- pokemon(name(X),_,_,attack(waterfall,_)).
X = wartortle ;
false.

?- pokemon(name(X),_,_,attack(poison-powder,_)).
X = venusaur ;
false.

?- pokemon(_,water,_,attack(A,_)),write(A),nl,fail.
water-gun
amnesia
dashing-punch
bubble
waterfall
hydro-pump
slap
star-freeze
false.

?- pokemon(name(poliwhirl),_,hp(HP),_).
HP = 80.

?- pokemon(name(butterfree),_,hp(HP),_).
HP = 130.

```

```
?- pokemon(name(X),_,hp(HP),_),HP>85,write(X),nl,fail.  
raichu  
venusaur  
butterfree  
charizard  
ninetails  
poliwrath  
blastoise  
false.  
  
?- pokemon(name(X),_,_,attack(_,D)),D>60,write(X),nl,fail.  
raichu  
venusaur  
butterfree  
charizard  
ninetails  
false.  
  
?- pokemon(name(X),_,hp(HP),_),cen(X),write(X),write(' : '),write(HP),nl,fail.  
pikachu: 60  
bulbasaur: 40  
caterpie: 50  
charmander: 50  
vulpix: 60  
poliwag: 60  
squirtle: 40  
staryu: 40  
false.  
  
?- ■
```

Part 3: KB Extension

```
display_cen :- cen(X),write(X),nl,fail.

display_not_cen :- evolves(_,X),write(X),nl,fail.

generator(Name,X) :- pokemon(name(Name),X,_,_).

display_names :- pokemon(name(X),_,_,_),write(X),nl,fail.

display_attacks :- pokemon(_,_,_,attack(X,_)),write(X),nl,fail.

display_cen_attacks :- cen(X), pokemon(name(X),_,_,attack(Y,_)),write(Y),nl,fail.

indicate_attack(Name) :- pokemon(name(Name),_,_,attack(Y,_)),write(Name),write('->'),write(Y).

indicate_attacks :- pokemon(name(Name),_,_,attack(X,_)),write(Name),write('->'),write(X),nl,fail.

powerful(Name) :- pokemon(name(Name),_,_,attack(_,Y)),Y > 55.

tough(Name) :- pokemon(name(Name),_,_,hp(HP),_),HP > 100.

awesome(Name) :- pokemon(name(Name),_,_,hp(HP),attack(_,Y)),HP > 100,Y > 55.

powerful_but_vulnerable(Name) :- pokemon(name(Name),_,_,hp(HP),attack(_,Y)),HP < 101,Y > 55.

type(Name,Type) :- pokemon(name(Name),Type,_,_).
```

```
dump_kind(Type) :- pokemon(name(Name),Type,hp(HP),attack(A,D)),write(pokemon(name(Name),Type,hp(HP),attack(A,D))),nl,fail.

family(Name) :- write(Name),write(' '),evolves(Name,X),write(X),write(' '),evolves(X,Y),write(Y).

families :- cen(X),nl,write(X),evolves(X,Y),write(' '),write(Y),write(' '),evolves(Y,Z),write(Z),fail.

lineage(Name) :- pokemon(name(Name),Type,hp(HP),attack(A,D)),write(pokemon(name(Name),Type,hp(HP),attack(A,D))),evolves(Name,X),nl,pokemon(name(X),T1,hp(HP1),attack(A1,D1)),write(pokemon(name(X),T1,hp(HP1),attack(A1,D1))),evolves(X,Y),nl,pokemon(name(Y),T2,hp(HP2),attack(A2,D2)),write(pokemon(name(Y),T2,hp(HP2),attack(A2,D2))).
```

```
?- consult('/Users/LENOVO/Prolog/pokemon.pro').
true.

?- display_cen.
pikachu
bulbasaur
caterpie
charmander
vulpix
poliwhirl
poliwrath
squirtle
staryu
false.

?- display_not_cen.
raichu
ivysaur
venusaur
metapod
butterfree
charmeleon
charizard
ninetails
poliwhirl
poliwrath
wartortle
blastoise
starmie
false.

?- generator(Name,fire).
Name = charmander ;
Name = charmeleon ;
Name = charizard ;
Name = vulpix ;
Name = ninetails.

?- generator(Name,water).
Name = poliwhirl ;
Name = poliwrath ;
Name = squirtle ;
Name = wartortle ;
Name = blastoise ;
Name = staryu ;
Name = starmie.
```

```
?- generator(Name,fire).
Name = charmander ;
Name = charmeleon ;
Name = charizard ;
Name = vulpix ;
Name = ninetails.

?- generator(Name,water).
Name = poliwag ;
Name = poliwhirl ;
Name = poliwrath ;
Name = squirtle ;
Name = wartortle ;
Name = blastoise ;
Name = staryu ;
Name = starmie.

?- generator(Name,electric).
Name = pikachu ;
Name = raichu.

?- generator(Name,grass).
Name = bulbasaur ;
Name = ivysaur ;
Name = venusaur ;
Name = caterpie ;
Name = metapod ;
Name = butterfree.

?- display_names.
pikachu
raichu
bulbasaur
ivysaur
venusaur
caterpie
metapod
butterfree
charmander
charmeleon
charizard
vulpix
ninetails
poliwag
poliwhirl
poliwrath
squirtle
wartortle
blastoise
staryu
starmie
false.
```

```
?- display_attacks.  
gnaw  
thunder-shock  
leech-seed  
vine-whip  
poison-powder  
gnaw  
stun-spore  
whirlwind  
scratch  
slash  
royal-blaze  
confuse-ray  
fire-blast  
water-gun  
amnesia  
dashing-punch  
bubble  
waterfall  
hydro-pump  
slap  
star-freeze  
false.  
  
?- display_cen_attacks.  
gnaw  
leech-seed  
gnaw  
scratch  
confuse-ray  
water-gun  
bubble  
slap  
false.  
  
?- indicate_attack(charmander).  
charmander->scratch  
true .  
  
?- indicate_attack(bulbasaur).  
bulbasaur->leech-seed  
true .
```

```
?- indicate_attacks.  
pikachu->gnaw  
raichu->thunder-shock  
bulbasaur->leech-seed  
ivysaur->vine-whip  
venusaur->poison-powder  
caterpie->gnaw  
metapod->stun-spore  
butterfree->whirlwind  
charmander->scratch  
charmeleon->slash  
charizard->royal-blaze  
vulpix->confuse-ray  
ninetails->fire-blast  
poliwag->water-gun  
poliwhirl->amnesia  
poliwrath->dashing-punch  
squirtle->bubble  
wartortle->waterfall  
blastoise->hydro-pump  
staryu->slap  
starmie->star-freeze  
false.
```

```
?- powerful(Name).  
Name = raichu ;  
Name = venusaur ;  
Name = butterfree ;  
Name = charizard ;  
Name = ninetails ;  
Name = wartortle ;  
Name = blastoise ;  
false.
```

```
?- tough(Name).
Name = venusaur ;
Name = butterfree ;
Name = charizard ;
Name = poliwraith ;
Name = blastoise ;
false.

?- awesome(Name).
Name = venusaur ;
Name = butterfree ;
Name = charizard ;
Name = blastoise ;
false.

?- powerful_but_vulnerable(Name).
Name = raichu ;
Name = ninetails ;
Name = wartortle ;
false.

?- type(squirtle,Type).
Type = water.

?- type(caterpie,Type).
Type = grass.

?- type(Name,fire),write(Name),nl,fail.
charmander
charmeleon
charizard
vulpix
ninetails
false.

?- dump_kind(water).
pokemon(name(poliwag),water,hp(60),attack(water-gun,30))
pokemon(name(poliwhirl),water,hp(80),attack(amnesia,30))
pokemon(name(poliwrath),water,hp(140),attack(dashing-punch,50))
pokemon(name(squirtle),water,hp(40),attack(bubble,10))
pokemon(name(wartortle),water,hp(80),attack(waterfall,60))
pokemon(name(blastoise),water,hp(140),attack(hydro-pump,60))
pokemon(name(staryu),water,hp(40),attack(slap,20))
pokemon(name(starmie),water,hp(60),attack(star-freeze,20))
false.

?- dump_kind(grass).
pokemon(name(bulbasaur),grass,hp(40),attack(leech-seed,20))
pokemon(name(ivysaur),grass,hp(60),attack(vine-whip,30))
pokemon(name(venusaur),grass,hp(140),attack(poison-powder,70))
pokemon(name(caterpie),grass,hp(50),attack(gnaw,20))
pokemon(name(metapod),grass,hp(70),attack(stun-spore,20))
pokemon(name(butterfree),grass,hp(130),attack(whirlwind,80))
false.
```

```
?- family(pikachu).
pikachu raichu
false.

?- family(bulbasaur).
bulbasaur ivysaur venusaur
true.

?- family(caterpie).
caterpie metapod butterfree
true.

?- families.
pikachu raichu
bulbasaur ivysaur venusaur
caterpie metapod butterfree
charmander charmeleon charizard
vulpix ninetails
poliwhirl poliwhirl poliwhirl
squirtle wartortle blastoise
staryu starmie
false.

?- lineage(pikachu).
pokemon(name(pikachu),electric, hp(60), attack(gnaw, 10))
pokemon(name(raichu),electric, hp(90), attack(thunder-shock, 90))
false.

?- lineage(squirtle).
pokemon(name(squirtle),water, hp(40), attack(bubble, 10))
pokemon(name(wartortle),water, hp(80), attack(waterfall, 60))
pokemon(name(blastoise),water, hp(140), attack(hydro-pump, 60))
true.

?- lineage(wartortle).
pokemon(name(wartortle),water, hp(80), attack(waterfall, 60))
pokemon(name(blastoise),water, hp(140), attack(hydro-pump, 60))
false.

?- lineage(blastoise).
pokemon(name(blastoise),water, hp(140), attack(hydro-pump, 60))
false.

?- lineage(charmander).
pokemon(name(charmander),fire, hp(50), attack(scratch, 10))
pokemon(name(charmeleon),fire, hp(80), attack(slash, 50))
pokemon(name(charizard),fire, hp(170), attack(royal-blaze, 100))
true.

?- ■
```

Part 5: KB Augmented by 12 Pokemon

```
% -----
% -----
% --- File: pokemon.pro
% --- Line: Just a few facts about pokemon
% -----

% -----
% --- cen(P) :: Pokemon P was "creatio ex nihilo"

cen(pikachu).
cen(bulbasaur).
cen(caterpie).
cen(charmander).
cen(vulpix).
cen(poliwag).
cen(squirtle).
cen(staryu).

% -----
% --- evolves(P,Q) :: Pokemon P directly evolves to pokemon Q

evolves(pikachu,raichu).
evolves(bulbasaur,ivysaur).
evolves(ivysaur,venusaur).
evolves(caterpie,metapod).
evolves(metapod,butterfree).
evolves(charmander,charmeleon).
evolves(charmeleon,charizard).
evolves(vulpix,ninetails).
evolves(poliwag,poliwhirl).
evolves(poliwhirl,poliwrath).
evolves(squirtle,wartortle).
evolves(wartortle,blastoise).
evolves(staryu,starmie).

% -----
% --- pokemon(name(N),T,hp(H),attach(A,D)) :: There is a pokemon with
% --- name N, type T, hit point value H, and attach named A that does
% --- damage D.

pokemon(name(pikachu), electric, hp(60), attack(gnaw, 10)).
pokemon(name(raichu), electric, hp(90), attack(thunder-shock, 90)).
pokemon(name(voltrob), electric, hp(40), attack(thunder-shock, 65)).
pokemon(name(raikou), electric, hp(90), attack(volt_switch, 85)).
```

pokemon(name(eelektross), electric, hp(85), attack(spark, 115)).

pokemon(name(bulbasaur), grass, hp(40), attack(leech-seed, 20)).
pokemon(name(ivysaur), grass, hp(60), attack(vine-whip, 30)).
pokemon(name(venusaur), grass, hp(140), attack(poison-powder, 70)).
pokemon(name(tangela), grass, hp(65), attack(grass-knot, 55)).
pokemon(name(vileplume), grass, hp(75), attack(solar-beam, 80)).

pokemon(name(caterpie), grass, hp(50), attack(gnaw, 20)).
pokemon(name(metapod), grass, hp(70), attack(stun-spore, 20)).
pokemon(name(butterfree), grass, hp(130), attack(whirlwind, 80)).
pokemon(name(jumpluff), grass, hp(75), attack(fairy-wind, 55)).

pokemon(name(charmander), fire, hp(50), attack(scratch, 10)).
pokemon(name(charmeleon), fire, hp(80), attack(slash, 50)).
pokemon(name(charizard), fire, hp(170), attack(royal-blaze, 100)).
pokemon(name(cyndaquil), fire, hp(39), attack(flame_wheel)).

pokemon(name(vulpix), fire, hp(60), attack(confuse-ray, 20)).
pokemon(name(ninetails), fire, hp(100), attack(fire-blast, 120)).
pokemon(name(blaziken), fire, hp(80), attack(blast-burn, 120)).
pokemon(name(pyroar), fire, hp(86), attack(fire-fang, 68)).

pokemon(name(poliwag), water, hp(60), attack(water-gun, 30)).
pokemon(name(poliwhirl), water, hp(80), attack(amnesia, 30)).
pokemon(name(poliwrath), water, hp(140), attack(dashing-punch, 50)).
pokemon(name(goldluck), water, hp(80), attack(confusion, 82)).
pokemon(name(inteleon), water, hp(70), attack(focus-energy, 85)).

pokemon(name(squirtle), water, hp(40), attack(bubble, 10)).
pokemon(name(wartortle), water, hp(80), attack(waterfall, 60)).
pokemon(name(blastoise), water, hp(140), attack(hydro-pump, 60)).
pokemon(name(ducklett), water, hp(62), attack(brave-bird, 44)).

pokemon(name(staryu), water, hp(40), attack(slap, 20)).
pokemon(name(starmie), water, hp(60), attack(star-freeze, 20)).

display_cen :- cen(X), write(X), nl, fail.

display_not_cen :- evolves(_, X), write(X), nl, fail.

generator(Name, X) :- pokemon(name(Name), X, _, _).

display_names :- pokemon(name(X), _, _, _), write(X), nl, fail.

display_attacks :- pokemon(_, _, _, attack(X, _)), write(X), nl, fail.

display_cen_attacks :- cen(X), pokemon(name(X),_,_,attack(Y,_)),write(Y),nl,fail.

indicate_attack(Name) :- pokemon(name(Name),_,_,attack(Y,_)),write(Name),write('->'),write(Y).

indicate_attacks :- pokemon(name(Name),_,_,attack(X,_)),write(Name),write('->'),write(X),nl,fail.

powerful(Name) :- pokemon(name(Name),_,_,attack(_,Y)),Y > 55.

tough(Name) :- pokemon(name(Name),_,hp(HP),_),HP > 100.

awesome(Name) :- pokemon(name(Name),_,hp(HP),attack(_,Y)),HP > 100,Y > 55.

powerful_but_vulnerable(Name) :- pokemon(name(Name),_,hp(HP),attack(_,Y)),HP < 101,Y > 55.

type(Name,Type) :- pokemon(name(Name),Type,_,_).

dump_kind(Type) :-

pokemon(name(Name),Type,hp(HP),attack(A,D)),write(pokemon(name(Name),Type,hp(HP),attack(A,D))),nl,fail.

family(Name) :- write(Name),write(' '),evolves(Name,X),write(X),write(' '),evolves(X,Y),write(Y).

families :- cen(X),nl,write(X),evolves(X,Y),write(' '),write(Y),write(' '),evolves(Y,Z),write(Z),fail.

lineage(Name) :-

pokemon(name(Name),Type,hp(HP),attack(A,D)),write(pokemon(name(Name),Type,hp(HP),attack(A,D))),evolves(Name,X),nl,pokemon(name(X),T1,hp(HP1),attack(A1,D1)),write(pokemon(name(X),T1,hp(HP1),attack(A1,D1))),evolves(X,Y),nl,pokemon(name(Y),T2,hp(HP2),attack(A2,D2)),write(pokemon(name(Y),T2,hp(HP2),attack(A2,D2))).

```
?- display_cen.  
pikachu  
bulbasaur  
caterpie  
charmander  
vulpix  
poliwhirl  
squirtle  
staryu  
false.  
  
?- display_not_cen.  
raichu  
ivysaur  
venusaur  
metapod  
butterfree  
charmeleon  
charizard  
ninetails  
poliwhirl  
poliwrath  
wartortle  
blastoise  
starmie  
false.  
  
?- generator(Name,fire).  
Name = charmander ;  
Name = charmeleon ;  
Name = charizard ;  
Name = cyndaquil ;  
Name = vulpix ;  
Name = ninetails ;  
Name = blaziken ;  
Name = pyroar.  
  
?- generator(Name,water).  
Name = poliwhirl ;  
Name = poliwrath ;  
Name = golduck ;  
Name = inteleon ;  
Name = squirtle ;  
Name = wartortle ;  
Name = blastoise ;  
Name = ducklett ;  
Name = staryu ;  
Name = starmie.
```

```
?- generator(Name,electric).  
Name = pikachu ;  
Name = raichu ;  
Name = voltrob ;  
Name = raikou ;  
Name = eeлектross.
```

```
?- generator(Name,grass).  
Name = bulbasaur ;  
Name = ivysaur ;  
Name = venusaur ;  
Name = tangela ;  
Name = vileplume ;  
Name = caterpie ;  
Name = metapod ;  
Name = butterfree ;  
Name = jumpluff.
```

```
?- display_names.
```

```
pikachu  
raichu  
voltrob  
raikou  
eeлектross  
bulbasaur  
ivysaur  
venusaur  
tangela  
vileplume  
caterpie  
metapod  
butterfree  
jumpluff  
charmander  
charmeleon  
charizard  
cyndaquil  
vulpix  
ninetails  
blaziken  
pyroar  
poliwag  
poliwhirl  
poliwrath  
goldluck  
inteleon  
squirtle  
wartortle  
blastoise  
ducklett  
staryu  
starmie  
false.
```

```
?- display_attacks.
```

```
gnaw  
thunder-shock  
thunder-shock  
volt_switch  
spark  
leech-seed  
vine-whip  
poison-powder  
grass-knot  
solar-beam  
gnaw  
stun-spore  
whirlwind  
fairy-wind  
scratch  
slash  
royal-blaze  
confuse-ray  
fire-blast  
blast-burn  
fire-fang  
water-gun  
amnesia  
dashing-punch  
confusion  
focus-energy  
bubble  
waterfall  
hydro-pump  
brave-bird  
slap  
star-freeze
```

```
false.
```

```
?- display_cen_attacks.
```

```
gnaw  
leech-seed  
gnaw  
scratch  
confuse-ray  
water-gun  
bubble  
slap
```

```
false.
```

```
?- indicate_attack(charmander).  
charmander->scratch  
true .
```

```
?- indicate_attacks.  
pikachu->gnaw  
raichu->thunder-shock  
voltrob->thunder-shock  
raikou->volt_switch  
eelektross->spark  
bulbasaur->leech-seed  
ivysaur->vine-whip  
venusaur->poison-powder  
tangela->grass-knot  
vileplume->solar-beam  
caterpie->gnaw  
metapod->stun-spore  
butterfree->whirlwind  
jumpluff->fairy-wind  
charmander->scratch  
charmeleon->slash  
charizard->royal-blaze  
vulpix->confuse-ray  
ninetails->fire-blast  
blaziken->blast-burn  
pyroar->fire-fang  
poliwag->water-gun  
poliwhirl->amnesia  
poliwrath->dashing-punch  
goldluck->confusion  
inteleon->focus-energy  
squirtle->bubble  
wartortle->waterfall  
blastoise->hydro-pump  
ducklett->brave-bird  
staryu->slap  
starmie->star-freeze  
false .
```

```
?- powerful(Name).  
Name = raichu ;  
Name = voltrob ;  
Name = raikou ;  
Name = eelektross ;  
Name = venusaur ;  
Name = vileplume ;  
Name = butterfree ;  
Name = charizard ;  
Name = ninetails ;  
Name = blaziken ;  
Name = pyroar ;  
Name = goldluck ;  
Name = inteleon ;  
Name = wartortle ;  
Name = blastoise ;  
false .
```

```
?- tough(Name).  
Name = venusaur ;  
Name = butterfree ;  
Name = charizard ;  
Name = poliwraith ;  
Name = blastoise ;  
false.  
  
?- awesome(Name).  
Name = venusaur ;  
Name = butterfree ;  
Name = charizard ;  
Name = blastoise ;  
false.  
  
?- powerful_but_vulnerable(Name).  
Name = raichu ;  
Name = voltrob ;  
Name = raikou ;  
Name = eelectross ;  
Name = vileplume ;  
Name = ninetails ;  
Name = blaziken ;  
Name = pyroar ;  
Name = goldluck ;  
Name = inteleon ;  
Name = wartortle ;  
false.  
  
?- type(squirtle,Type).  
Type = water.
```

```

?- type(caterpie,Type).
Type = grass.

?- type(Name,fire),write(Name),nl,fail.
charmander
charmeleon
charizard
cyndaquil
vulpix
ninetails
blaziken
pyroar
false.

?- dump_kind(water).
pokemon(name(poliwag),water,hp(60),attack(water-gun,30))
pokemon(name(poliwhirl),water,hp(80),attack(amnesia,30))
pokemon(name(poliwrath),water,hp(140),attack(dashing-punch,50))
pokemon(name(goldluck),water,hp(80),attack(confusion,82))
pokemon(name(inteleon),water,hp(70),attack(focus-energy,85))
pokemon(name(squirtle),water,hp(40),attack(bubble,10))
pokemon(name(wartortle),water,hp(80),attack(waterfall,60))
pokemon(name(blastoise),water,hp(140),attack(hydro-pump,60))
pokemon(name(ducklett),water,hp(62),attack(brave-bird,44))
pokemon(name(staryu),water,hp(40),attack(slap,20))
pokemon(name(starmie),water,hp(60),attack(star-freeze,20))
false.

?- dump_kind(grass).
pokemon(name(bulbasaur),grass,hp(40),attack(leech-seed,20))
pokemon(name(ivysaur),grass,hp(60),attack(vine-whip,30))
pokemon(name(venusaur),grass,hp(140),attack(poison-powder,70))
pokemon(name(tangela),grass,hp(65),attack(grass-knot,55))
pokemon(name(vileplume),grass,hp(75),attack(solar-beam,80))
pokemon(name(caterpie),grass,hp(50),attack(gnaw,20))
pokemon(name(metapod),grass,hp(70),attack(stun-spore,20))
pokemon(name(butterfree),grass,hp(130),attack(whirlwind,80))
pokemon(name(jumpluff),grass,hp(75),attack(fairy-wind,55))
false.

?- family(pikachu).
pikachu raichu
false.

?- family(bulbasaur).
bulbasaur ivysaur venusaur
true.

```

```
?- family(caterpie).
caterpie metapod butterfree
true.

?- families.

pikachu raichu
bulbasaur ivysaur venusaur
caterpie metapod butterfree
charmander charmeleon charizard
vulpix ninetails
poliwhag poliwhirl poliwrath
squirtle wartortle blastoise
staryu starmie
false.

?- lineage(pikachu).
pokemon(name(pikachu),electric,hp(60),attack(gnaw,10))
pokemon(name(raichu),electric,hp(90),attack(thunder-shock,90))
false.

?- lineage(wartortle).
pokemon(name(wartortle),water,hp(80),attack(waterfall,60))
pokemon(name(blastoise),water,hp(140),attack(hydro-pump,60))
false.
```

```
?- lineage(squirtle).
pokemon(name(squirtle),water,hp(40),attack(bubble,10))
pokemon(name(wartortle),water,hp(80),attack(waterfall,60))
pokemon(name(blastoise),water,hp(140),attack(hydro-pump,60))
true .

?- lineage(blastoise).
pokemon(name(blastoise),water,hp(140),attack(hydro-pump,60))
false.

?- lineage(charmander).
pokemon(name(charmander),fire,hp(50),attack(scratch,10))
pokemon(name(charmeleon),fire,hp(80),attack(slash,50))
pokemon(name(charizard),fire,hp(170),attack(royal-blaze,100))
true .

?-
```

Task 2 - List Processing

Head/Tail Exercises

```
?- [H|T] = [red, yellow, blue, green].
H = red,
T = [yellow, blue, green].

?- [H,T] = [red, yellow, blue, green].
false.

?- [F|_] = [red, yellow, blue, green].
F = red.

?- [_|[S|_]] = [red, yellow, blue, green].
S = yellow.

?- [F|[S|R]] = [red, yellow, blue, green].
F = red,
S = yellow,
R = [blue, green].

?- List = [this|[and, that]].
List = [this, and, that].

?- List = [this, and, that].
List = [this, and, that].

?- [a,[b,c]] = [a,b,c].
false.

?- [a|[b,c]] = [a, b, c].
true.

?- [cell(Row,Column)|Rest] = [cell(1,1), cell(3,2), cell(1,3)].
Row = Column, Column = 1,
Rest = [cell(3, 2), cell(1, 3)].

?- [X|Y] = [one(un,uno), two(dos, deux), three(thois, tres)].
X = one(un, uno),
Y = [two(dos, deux), three(thois, tres)].

?- ■
```

List Processing Code

```
first([H|_], H).
rest([_|T], T).

last([H|[]], H).
last([_|T], Result) :- last(T, Result).

nth(0, [H|_], H).
nth(N, [_|T], E) :- K is N - 1, nth(K, T, E).

writelist([]).
writelist([H|T]) :- write(H), nl, writelist(T).

sum([], 0).
sum([Head|Tail], Sum) :- sum(Tail, SumOfTail), Sum is Head + SumOfTail.

add_first(X, L, [X|L]).

add_last(X, [], [X]).
add_last(X, [H|T], [H|TX]) :- add_last(X, T, TX).

iota(0, []).
iota(N, IotaN) :- K is N - 1, iota(K, IotaK), add_last(N, IotaK, IotaN).

pick(L, Item) :- length(L, Length), random(0, Length, RN), nth(RN, L, Item).

make_set([], []).
make_set([H|T], TS) :- member(H, T), make_set(T, TS).
make_set([H|T], [H|TS]) :- make_set(T, TS).

product([], 1).
product([H|T], Result) :- product(T, ProductOfT), Result is H * ProductOfT.

factorial(0, 1).
factorial(N, Result) :- iota(N, IotaN), product(IotaN, Result).

make_list(0, _, []).
make_list(N, Item, [Item|ResultR]) :- K is N - 1, make_list(K, Item, ResultR).
but_first([_], []).
but_first([_|T], T).
but_last([_], []).
but_last(List, RDC) :- reverse(List, ReversedList), but_first(ReversedList, ReversedX1), reverse(ReversedX1, RDC).

is_palindrome([]).
is_palindrome([_]).
is_palindrome(List) :- first(List, First), last(List, Last), First = Last, but_first(List, ListX1), but_last(ListX1, ListX2), is_palindrome(ListX2).

noun_phrase([the, Adjective, Noun]) :- pick([hardworking, tasty, silly, witty, righteous, calm], Adjective), pick([man, woman, dog, cat, ninja, pirate, robot, student], Noun).

sentence(Sentence) :- pick([ran, walked, fought, punched, threw, mixed, ignited, jumped], Verb), noun_phrase(Phrase1), noun_phrase(Phrase2), add_last(Verb, Phrase1, PhraseWithVerb), append(PhraseWithVerb, Phrase2, Sentence).
```

Demo for Example List Processors

```
?- first([apple],First).
First = apple.

?- first([c,d,e,f,g,a,b],P).
P = c.

?- rest([apple],Rest).
Rest = [].

?- rest([c,d,e,f,g,a,b],Rest).
Rest = [d, e, f, g, a, b].

?- last([peach],Last).
Last = peach.

?- last([c,d,e,f,g,a,b],P).
P = b.

?- nth(0,[zero,one,two,three,four],Element).
Element = zero.

?- nth(3,[four,three,two,one,zero],Element).
Element = one.

?- writelist([red,yellow,blue,green,purple,orange]).
red
yellow
blue
green
purple
orange
true.

?- sum([],Sum).
Sum = 0.

?- sum([2,3,5,7,11],SumOfPrimes).
SumOfPrimes = 28.

?- add_first(thing,[],Result).
Result = [thing].

?- add_first(racket,[prolog,haskell,rust],Languages).
Languages = [racket, prolog, haskell, rust].

?- add_last(thing,[],Result).
Result = [thing].

?- add_last(rust,[racket,prolog,haskell],Languages).
Languages = [racket, prolog, haskell, rust].
```

```
?- iota(5,Iota5).  
Iota5 = [1, 2, 3, 4, 5] .  
  
?- iota(9,Iota9).  
Iota9 = [1, 2, 3, 4, 5, 6, 7, 8, 9] .  
  
?- pick([cherry,peach,apple,blueberry],Pie).  
Pie = apple .  
  
?- pick([cherry,peach,apple,blueberry],Pie).  
Pie = blueberry .  
  
?- pick([cherry,peach,apple,blueberry],Pie).  
Pie = apple .  
  
?- pick([cherry,peach,apple,blueberry],Pie).  
Pie = cherry .  
  
?- pick([cherry,peach,apple,blueberry],Pie).  
Pie = peach .  
  
?- pick([cherry,peach,apple,blueberry],Pie).  
Pie = apple .  
  
?- pick([cherry,peach,apple,blueberry],Pie).  
Pie = peach .  
  
?- pick([cherry,peach,apple,blueberry],Pie).  
Pie = peach .  
  
?- make_set([1,1,2,1,2,3,1,2,3,4],Set).  
Set = [1, 2, 3, 4] .  
  
?- make_set([bit,bot,bet,bot,bot,bit],B).  
B = [bet, bot, bit] .  
  
?-
```

Demo for List Processing Exercises

```
?- product([],P).  
P = 1.  
  
?- product([1,3,5,7,9],Product).  
Product = 945.  
  
?- iota(9,Iota),product(Iota,Product).  
Iota = [1, 2, 3, 4, 5, 6, 7, 8, 9],  
Product = 362880 .  
  
?- make_list(7,seven,Seven).  
Seven = [seven, seven, seven, seven, seven, seven, seven] .  
  
?- make_list(8,2,List).  
List = [2, 2, 2, 2, 2, 2, 2, 2] .  
  
?- but_first([a,b,c],X).  
X = [b, c].  
  
?- but_last([a,b,c,d,e],X).  
X = [a, b, c, d].
```

```
?- is_palindrome([x]).
true .

?- is_palindrome([a,b,c]).
false.

?- is_palindrome([a,b,b,a]).
true .

?- is_palindrome([1,2,3,4,5,4,2,3,1]).
false.

?- is_palindrome([c,o,f,f,e,e,e,e,f,f,o,c]).
true .

?- noun_phrase(NP).
NP = [the, calm, dog] .

?- noun_phrase(NP).
NP = [the, tasty, pirate] .

?- noun_phrase(NP).
NP = [the, tasty, ninja] .

?- noun_phrase(NP).
NP = [the, righteous, ninja] .

?- noun_phrase(NP).
NP = [the, tasty, ninja] .

?- noun_phrase(NP).
NP = [the, hardworking, cat] .

?- sentence(S).
S = [the, hardworking, woman, walked, the, righteous, cat] .

?- sentence(S).
S = [the, tasty, cat, ignited, the, righteous, woman] .

?- sentence(S).
S = [the, calm, pirate, fought, the, tasty, ninja] .

?- sentence(S).
S = [the, righteous, dog, jumped, the, hardworking, robot] .

?- sentence(S).
S = [the, calm, woman, fought, the, witty, student] .

?- sentence(S).
S = [the, tasty, ninja, threw, the, calm, robot] .

?- sentence(S).
S = [the, righteous, dog, jumped, the, calm, pirate] .
```

```
?- sentence(S).  
S = [the, hardworking, robot, jumped, the, hardworking, man] .  
  
?- sentence(S).  
S = [the, calm, cat, punched, the, silly, cat] .  
  
?- sentence(S).  
S = [the, hardworking, pirate, fought, the, witty, cat] .  
  
?- sentence(S).  
S = [the, hardworking, cat, jumped, the, hardworking, student] .  
  
?- sentence(S).  
S = [the, hardworking, woman, punched, the, calm, robot] .  
  
?- sentence(S).  
S = [the, righteous, dog, punched, the, calm, woman] .  
  
?- sentence(S).  
S = [the, silly, student, ran, the, tasty, ninja] .  
  
?- sentence(S).  
S = [the, witty, ninja, ran, the, witty, cat] .  
  
?- sentence(S).  
S = [the, tasty, cat, threw, the, calm, student] .  
  
?-
```