

*Universidad Politécnica Salesiana*

*Sistemas Expertos*

***Tema:***

Animales en Clips

***Integrantes:***

*Carlos Álvarez*

***Docente:***

*Ing. Diego Quisi*



# Animales en Clips

- **Código en Clips**

- Definición de las reglas

```
(defrule mamifero-1
(tiene-pelos)
=>
(assert (es-mamifero))
(printout t " es un mamifero " crlf)
)
```

```
(defrule mamifero-2
(da-leche)
=>
(assert (es-mamifero))
(printout t " es un mamifero " crlf)
)
```

```
(defrule aves-1
(tiene-plumas)
=>
(assert (es-ave))
(printout t " es una ave " crlf)
)
```

```
(defrule reptiles-1
(tiene-sangre-fria)
=>
(assert (es-reptil))
(printout t " es un reptil " crlf)
)
```

```
(defrule peces-1
(vive-oceano)
=>
(assert (es-pezu))
(printout t " es un pez " crlf)
)
```

```
(defrule ungulado-1
(es-mamifero)
(tiene-pezuñas)
=>
(assert (es-ungulado))
(printout t " es un ungulado " crlf)
)
```

```
(defrule ungulado-2
(es-mamifero)
(rumia)
=>
```



```
(assert (es-ungulado))  
(printout t " es un ungulado " crlf)  
)
```

```
(defrule carnivoros-1  
(es-mamifero)  
(come-carne)  
=>  
(assert (es-carnivoro))  
(printout t " es un mamifero carnivoro " crlf)  
)
```

```
(defrule aulla-1  
(es-mamifero)  
(aulla)  
=>  
(assert (es-canino))  
(printout t " es un mamifero canino " crlf)  
)
```

```
(defrule paquidermos-1  
(es-mamifero)  
(tiene-colmillos)  
=>  
(assert (es-paquidermo))  
(printout t " es un mamifero paquidermo " crlf)  
)
```

```
(defrule es-rapaz  
(es-ave)  
(tiene-garras)  
=>  
(assert (es-rapaz))  
(printout t " es una ave rapaz " crlf)  
)
```

```
(defrule es-carronera  
(es-ave)  
(come-desechos)  
=>  
(assert (es-carronera))  
(printout t " es una ave carronera " crlf)  
)
```

```
(defrule es-grande  
(es-ave)  
(mide-3metros)  
=>  
(assert (es-grande))  
(printout t " es una ave grande " crlf)  
)
```

```
(defrule pico-curvado  
(es-ave)  
(garras-prensiles)  
=>
```



```
(assert (pico-curvado))  
(printout t " es una ave con pico curvado " crlf)  
)
```

```
(defrule es-domestica  
(es-ave)  
(no-vuela)  
=>  
(assert (es-domestica))  
(printout t " es una ave domestica " crlf)  
)
```

```
(defrule es-lagarto  
(es-reptil)  
(traga-carne)  
=>  
(assert (es-lagarto))  
(printout t " es un reptil lagarto " crlf)  
)
```

```
(defrule es-ojosGrandes  
(es-reptil)  
(es-color)  
=>  
(assert (es-ojosGrandes))  
(printout t " es un reptil cameleon " crlf)  
)
```

```
(defrule es-gecko  
(es-reptil)  
(es-arcoiris)  
=>  
(assert (es-gecko))  
(printout t " es un reptil gecko " crlf)  
)
```

```
(defrule es-negro  
(es-mamifero)  
(tiene-colmillo)  
=>  
(assert (es-negro))  
(printout t " es un mamifero negro " crlf)  
)
```

```
(defrule tiene-caparazon  
(es-reptil)  
(come-vegetacion)  
=>  
(assert (tiene-caparazon))  
(printout t " es un reptil con caparazon " crlf)  
)
```

```
(defrule nada-rapido  
(es-pezo)  
(no-duerme)  
=>
```



```
(assert (nada-rapido))  
(printout t " es un pez que nada rapido " crlf)  
)
```

```
(defrule es-cetaceo  
(es-mamifero)  
(es-enorme)  
=>  
(assert (es-cetaceo))  
(printout t " es un mamifero cetaceo" crlf)  
)
```

```
(defrule es-colorido  
(es-pezo)  
(es-tomate)  
=>  
(assert (es-colorido))  
(printout t " es un pez pomacentridae" crlf)  
)
```

```
(defrule vive-frio  
(es-pezo)  
(es-comida)  
=>  
(assert (vive-frio))  
(printout t " es un pez que vive en el frio" crlf)  
)
```

```
(defrule es-trofeo  
(es-pezo)  
(tiene-pico-largo)  
=>  
(assert (es-trofeo))  
(printout t " es un pez trofeo" crlf)  
)
```

```
(defrule jirafa  
(es-ungulado)  
(tiene-cuello-largo)  
=>  
(assert (es-jirafa))  
(printout t " es una jirafa" crlf)  
)
```

```
(defrule cebra  
(es-ungulado)  
(tiene-rayas)  
=>  
(assert (es-cebra))  
(printout t " es una cebra" crlf)  
)
```

```
(defrule leon  
(es-carnivoro)  
(tiene-melena)  
=>
```



```
(assert (es-leon))  
(printout t " es un leon" crlf)  
)
```

```
(defrule lobo  
(es-canino)  
(vive-alaska)  
=>  
(assert (es-lobo))  
(printout t " es un lobo" crlf)  
)
```

```
(defrule elefante  
(es-paquidermo)  
(tiene-trompa)  
=>  
(assert (es-elefante))  
(printout t " es un elefante" crlf)  
)
```

```
(defrule aguila  
(es-rapaz)  
(come-carne)  
=>  
(assert (es-aguila))  
(printout t " es una aguila" crlf)  
)
```

```
(defrule buitre  
(es-carronera)  
(no-caza)  
=>  
(assert (es-buitre))  
(printout t " es un buitre" crlf)  
)
```

```
(defrule condor  
(es-grande)  
(vive-andes)  
=>  
(assert (es-condor))  
(printout t " es un condor" crlf)  
)
```

```
(defrule loro  
(pico-curvado)  
(es-de-color)  
=>  
(assert (loro))  
(printout t " es un loro" crlf)  
)
```

```
(defrule gallina  
(es-domestico)  
(come-maiz)  
=>
```



```
(assert (gallina))  
(printout t " es una gallina" crlf)  
)
```

```
(defrule cocodrilo  
(es-lagarto)  
(tiene-cola)  
=>  
(assert (cocodrilo))  
(printout t " es un cocodrilo" crlf)  
)
```

```
(defrule serpiente  
(es-toxico)  
(envenena)  
=>  
(assert (serpiente))  
(printout t " es una serpiente" crlf)  
)
```

```
(defrule caiman  
(es-sauropsida)  
(es-pequenio)  
=>  
(assert (caiman))  
(printout t " es un caiman" crlf)  
)
```

```
(defrule pantera  
(es-negro)  
(es-peligrosa)  
=>  
(assert (pantera))  
(printout t " es un pantera" crlf)  
)
```

```
(defrule tortuga  
(tiene-caparazon)  
(es-lenta)  
=>  
(assert (es-totuga))  
(printout t " es una totuga" crlf)  
)
```

```
(defrule tiburon  
(nada-rapido)  
(come-carne)  
=>  
(assert (es-tiburon))  
(printout t " es un tiburon" crlf)  
)
```

```
(defrule ballena  
(es-cetaceo)  
(come-banco-peces)  
=>
```



```
(assert (es-ballena))  
(printout t " es un ballena" crlf)  
)
```

```
(defrule pez-payaso  
  (es-colorido)  
  (es-blanco)  
=>  
  (assert (es-pez-payaso))  
  (printout t " es un pez payaso" crlf)  
)
```

```
(defrule trucha  
  (vive-frio)  
  (se-puede-criar)  
=>  
  (assert (es-trucha))  
  (printout t " es una trucha" crlf)  
)
```

```
(defrule pez-espada  
  (es-trofeo)  
  (es-perciforme)  
=>  
  (assert (es-pez-espada))  
  (printout t " es un pez espada" crlf)  
)  
;
```

- Definición de los parámetros de entrada

```
(deffacts inicio  
  (tiene-pelos)  
  (tiene-colmillo)  
  (es-peligrosa)  
)
```