



In [1]: `from experta import *`

In [3]: `#CLASES
class user(Fact):
 pass
class Usuariounlock(Fact):
 pass
class Soporteaplicacion(Fact):
 pass
class Checkorder(Fact):
 pass`

In [4]: `class usuario(KnowledgeEngine):

 @Rule(AND(user(unlockus=MATCH.unlockus & L(0))))
 def usuario1(self,unlockus):
 print("no se registro usuario \n Fin del programa")

 @Rule(AND(user(unlockus=MATCH.unlockus & L(1))))
 def usuario2(self,unlockus):
 print("el usuario registrado en el sistema ya fue desbloqueado")
 print('desea que le ayudemos brindandote soporte:')
 print('0 --> NO')
 print('1 --> SI')
 self.declare(Usuariounlock(nivel=int(input())))`

```
@Rule(AND(user(unlockus=MATCH.unlockus & L(1)),Usuariounlock(nivel=MATCH.nivel & L(0))))
def usuario3(self,nivel):
    print("usuario no requiere mas ayuda \n el usuario quedo desbloqueado \n Fin del programa")
```

```
@Rule(AND(user(unlockus=MATCH.unlockus & L(1)),Usuariounlock(nivel=MATCH.nivel & L(1))))
def usuario4(self,nivel):
    print("usuario requiere ayuda de soporte")
    print('Seleccione que tipo de soporte requiere')
    print('0 --> APLICACIONES DEL CLIENTE')
    print('1 --> APLICACIONES INTERNAS')
    self.declare(Soporteaplicacion(nivel=int(input())))
```

```
@Rule(AND(user(unlockus=MATCH.unlockus & L(1)),Usuariounlock(nivel=MATCH.nivel & L(1)), Soporteaplicacion(nivel=MATCH.sop & L(1))))
def usuario5(self,nivel):
    print("soporte aplicaciones del cliente")
    print('la solicitud debe ser reportada por medio de el jefe directo al cliente \n Fin del programa')
```

```
@Rule(AND(user(unlockus=MATCH.unlockus & L(1)),Usuariounlock(nivel=MATCH.nivel & L(1)), Soporteaplicacion(nivel=MATCH.sop & L(1))))
def usuario6(self,nivel):
    print("soporte aplicaciones internas")
    print('la solicitud quedo generada en el sistema y se asigno a uno de nuestros analistas')
    print('desea ver el estado de tu ticket:')
    print('0 --> NO')
    print('1 --> SI')
    self.declare(Checkorder(nivel=int(input())))
```

```
@Rule(AND(user(unlockus=MATCH.unlockus & L(1)),Usuariounlock(nivel=MATCH.nivel & L(1)), Soporteaplicacion(nivel=MATCH.sop & L(1))))
def usuario8(self,nivel):
    print("se revisa la solicitud registrada a tu usuario y esta ya se encuentra en estado asignado")
    print("fue un gusto ayudarte \n Fin del programa")
```

```
@Rule(AND(user(unlockus=MATCH.unlockus & L(1)),Usuariounlock(nivel=MATCH.nivel & L(1)), Soporteaplicacion(nivel=MATCH.sop & L(1)))
def usuario7(self,nivel):
    print("Fin del programa")
```

```
watch('RULES', 'FACTS')
A=usuario()
A.reset()
```

```
INFO:experta.watchers.FACTS: ==> <f-0>: InitialFact()
```

```
print('quieres validar si tu usuario esta bloqueado?:')
print('0 --> NO')
print('1 --> SI')
A.declare(user(unlockus=int(input())))
```

```
quieres validar si tu usuario esta bloqueado?:
0 --> NO
1 --> SI
1
```

```
INFO:experta.watchers.FACTS: ==> <f-1>: user(unlockus=1)
```

```
user(unlockus=1)
```

```
A.run()
```

```
INFO:experta.watchers.RULES:FIRE 1 usuario2: <f-1>
```

```
INFO:experta.watchers.RULES:FIRE 1 usuario2: <f-1>
```

```
el usuario registrado en el sistema ya fue desbloqueado
desea que le ayudemos brindandote soporte:
0 --> NO
1 --> SI
1
```

```
INFO:experta.watchers.FACTS: ==> <f-2>: Usuariounlock(nivel=1)
INFO:experta.watchers.RULES:FIRE 2 usuario4: <f-2>, <f-1>
```

```
usuario requiere ayuda de soporte
Seleccione que tipo de soporte requiere
0 --> APLICACIONES DEL CLIENTE
1 --> APLICACIONES INTERNAS
1
```

```
INFO:experta.watchers.FACTS: ==> <f-3>: Soporteaplicacion(nivel=1)
INFO:experta.watchers.RULES:FIRE 3 usuario6: <f-3>, <f-2>, <f-1>
```

```
soporte aplicaciones internas
la solicitud quedo generada en el sistema y se asigno a uno de nuestros analistas
desea ver el estado de tu ticket:
0 --> NO
1 --> SI
1
```

```
INFO:experta.watchers.FACTS: ==> <f-4>: Checkorder(nivel=1)
INFO:experta.watchers.RULES:FIRE 4 usuario8: <f-3>, <f-2>, <f-1>, <f-4>
```

```
se revisa la solicitud registrada a tu usuario y esta ya se encuentra en estado asignado
fue un gusto ayudarte
```

```
Fin del programa
```

gen=e47291eae12b90f540c4e4f0c987089018ed5e47d4b5