

U19CS082

SOURABH PATEL

Q) Here are some tables representing what we know about various products sold by the XYZ Equipment Company, and about XYZ's customer orders.

part-name part-number price

-----

Anvil 423 10.99

Stick of dynamite 567 1.97

Shotgun 128 99.99

Broccoli 256 0.99

Carrot stick 511 0.99

customer-name part-number quantity

-----

Coyote 423 300

Fudd 567 1

Bunny 511 94

Bunny 256 9723

Coyote 128 12

1. Represent this information as a Prolog knowledge base; during that process, you should change all upper-case letters to lowercase, combine multiple words into one, and omit punctuation. For example, "Stick of dynamite" will become stick of dynamite. Use predicate names, part and order for the tables mentioned above.

CODE:

```
parts(anvil,423,10.99).
parts(stick_of_dynamite,567,1.97).
parts(shotgun,128,99.99).
parts(broccoli,256,0.99).
parts(carrot_stick,511,0.99).

customer(coyote,423,300).
customer(fudd,567,1).
customer(bunny,511,94).
```

```
customer(bunny,256,9723).
customer(coyote,128,12).
```

#### OUTPUT:

```
?- parts(anvil,423,10.99).
true.

?- parts(shotgun,128,99.99).
ERROR: Unknown procedure: parts/4
ERROR:         However, there are definitions for:
ERROR:         parts/3
false.

?- parts(shotgun,128,99.99).
true.

?- ■
```

2. Write a predicate called `bigorder(cust)` that succeeds if `cust` is the name of a customer who has ordered more than 100 of any item in a single order.

#### CODE:

```
bigorder:-customer(X,Y,Z),
    (Z>100)->write(X),write(" "),write(Y),write(" "),write(Z),nl
    ).
```

#### OUTPUT:

```
?- bigorder.
coyote 423 300
true ;
bunny 256 9723
true ;
false.

?-
```

3. Write a predicate `pricey(part-name)` that evaluates to true if the `part-name` costs more than 50 \$.

#### CODE:

```
pricey:-parts(X,Y,Z),
    (Z>50)->write(X),write(" "),write(Y),write(" "),write(Z),nl
```

```
).  
)
```

OUTPUT:

```
?- pricey.  
shotgun 128 99.99  
true ;  
false.  
  
?-
```

4. Write a predicate that finds if the product 'Anvil' exists in a list of products given by {Anvil, Shotgun , Broccoli}.

CODE:

```
finds(X):-parts(Y,_,_),Y==X.
```

OUTPUT:

5. Write a predicate compareprod (Prod1,Prod2) that returns true if product 1 is more expensive than product 2.

CODE:

```
compareprod(Prod1,Prod2):-parts(Prod1,_,X),parts(Prod2,_,Y),x>Y.
```

OUTPUT: