

# Quiz 6 (Prolog)

**Due** Mar 12 at 11:59pm**Points** 100**Questions** 10**Available** Mar 10 at 12am - Mar 17 at 11:59pm**Time Limit** 90 Minutes**Allowed Attempts** 2

This quiz was locked Mar 17 at 11:59pm.

## Attempt History

	Attempt	Time	Score
LATEST	<u>Attempt 1</u>	30 minutes	100 out of 100

⚠️ Answers will be shown after your last attempt

Score for this attempt: **100** out of 100

Submitted Mar 12 at 2:22pm

This attempt took 30 minutes.

### Question 1

**10 / 10 pts**

Consider the following predicate `plays/2`.

```
plays(john,piano).
plays(john,cello).
plays(jill,cello).
plays(mike,piano).
plays(jane,guitar).
```

Write a goal that finds all people who play the cello.

☐ `Plays(cello).`

☐ `plays(X,Y).`

☒ `plays(X, cello).`

☐ `plays(People, X).`

☐ none of these answers

## Question 2

10 / 10 pts

Consider the following predicate plays/2.

```
plays(john,piano).  
plays(john,cello).  
plays(jill,cello).  
plays(mike,piano).  
plays(jane,guitar).
```

Define a predicate duet/2 that finds two (different) people who can play a duet with the same instrument.

- ☐ `duet(X,Y) :- plays(Z,X), plays(Z,X), X\=Y.`
- ☐ `duet(X,Y) :- plays(X,A), plays(Y,B).`
- ☒ `duet(X,Y) :- plays(X,Z), plays(Y,Z), X\=Y.`
- ☐ `duet(X,Y,Z) :- plays(X,Z), plays(Y,Z), X\=Y.`
- ☐ none of these

## Question 3

10 / 10 pts

Consider the following predicate plays/2.

```
plays(john,piano).  
plays(john,cello).  
plays(jill,cello).  
plays(mike,piano).  
plays(jane,guitar).
```

Define a predicate talent/1 that yields true for persons who can play more than one instrument.

☒ talent(X) :- plays(X,Y), plays(X,Z), Y\=Z.

☐ none of these

☐ talent(X) :- plays(X,Y), plays(X,Z).

☐ talent(X) :- plays(Y,X), plays(X,Y), Y\=Z.

☐ talent(X) :- plays(X,Y), plays(X,Z), Y=Z.

### Question 4

10 / 10 pts

The following predicates that describe books (first argument) and their authors (second argument) that can be borrowed from two different libraries (OSU Library and the Public Library).

```
osu(b,a).
osu(d,z).
osu(d,s).
osu(e,z).
osu(f,q).

public(b,a).
public(d,v).
public(c,a).
public(e,z).
public(h,z).
```

Define a predicate book/1 that can produce all the books available in any of the two libraries.

☐ book(X) :- osu(X,Y), public(X,Z).

☐ none of these

☐ book(X) :- osu(X,Y), public(X,Y).

☐ book(X) :- osu(X,Y).

☒ book(X) :- public(X,Z).

**Question 5****10 / 10 pts**

The following predicates that describe books (first argument) and their authors (second argument) that can be borrowed from two different libraries (OSU Library and the Public Library).

```
osu(b,a).  
osu(d,z).  
osu(d,s).  
osu(e,z).  
osu(f,q).  
  
public(b,a).  
public(d,v).  
public(c,a).  
public(e,z).  
public(h,z).
```

Define a predicate `writtenBy/2` that lists all books by a specific author that are available in any of the two libraries.

☐ `written(Author,Book) :- osu(Book,Author), public(Book,Author).`

☐ none of these

☐ `writtenBy(Author,Book) :- osu(Book,Author) or public(Book,Author).`

`writtenBy(Author,Book) :- osu(Book,Author).`

☒ `writtenBy(Author,Book) :- public(Book,Author).`

**Question 6****10 / 10 pts**

The following predicates that describe books (first argument) and their authors (second argument) that can be borrowed from two different libraries (OSU Library and the Public Library).

```
osu(b,a).  
osu(d,z).  
osu(d,s).  
osu(e,z).  
osu(f,q).
```

```
public(b,a).
public(d,v).
public(c,a).
public(e,z).
public(h,z).
```

Assume you defined a predicate `writtenBy/2` that lists all books by a specific author that are available in any of the two libraries.

Using `writtenBy`, write a goal/query that lists all books written by "z".

☐ none of these

☐ `writtenBy(z).`

☐ `writtenBy(Books,z).`

☐ `writtenBy("z",books).`

☒ `writtenBy(z,Books).`

## Question 7

10 / 10 pts

The predicate `majority/2` shows how the majority of all votes has voted on each measure.

```
poll(benton,1,yes). poll(lane,1,no). majority(1,no).
poll(benton,2,yes). poll(lane,3,no). majority(2,yes).
poll(benton,3,no). poll(polk,3,abstain). majority(3,no).
```

These predicates are the basis for the following question.

Define a predicate `agree/2` that is true for a county and measure if the county's vote agrees with the majority vote.

☐ `agree(C,M) :- poll(C,M,no), majority(M,no).`

☐ `agree(C,M) :- poll(C,M,_), majority(M,_).`

☐ none of these

☐ agree(C,M) :- poll(C,M,yes), majority(M,yes).

☒ agree(C,M) :- poll(C,M,Z), majority(M,Z).

## Question 8

10 / 10 pts

The predicate majority/2 shows how the majority of all votes has voted on each measure.

```
poll(benton,1,yes). poll(lane,1,no). majority(1,no).
poll(benton,2,yes). poll(lane,3,no). majority(2,yes).
poll(benton,3,no). poll(polk,3,abstain). majority(3,no).
```

These predicates are the basis for the following question.

Define a predicate diff/2 that produces pairs of counties who have voted differently on a measure.

☐ diff(C1,C2) :- poll(C1,M,Z), poll(C2,M,Z).

☐ diff(C1,C2) :- poll(C1,M,Z1), poll(C2,M,Z2).

☐ none of these

☐ diff(C1,C2) :- poll(C1,M1,Z1), poll(C2,M2,Z2), Z1\=Z2.

☒ diff(C1,C2) :- poll(C1,M,Z1), poll(C2,M,Z2), Z1\=Z2.

## Question 9

10 / 10 pts

For following program:

```
teacher(joe).
doctor(jane).
healthy(joe).
healthy(jane).
wealthy(jane).
```

```
lucky(X) :- healthy(X), wealthy(X).
```

What will be the solution of query:

?- lucky(X).

☐ none of the above

☐ X = joe

☒ X = jane

☐ Jane

☐ true.

### Question 10

10 / 10 pts

Select the correct predicate `sum/2` that computes the sum of the elements in a list.

☐ `sum([x],1).`  
☐ `sum([H|T], S) :- sum(T,S2), S = S + S2.`

☐ `sum([],_) = 0.`  
☐ `sum([H|T], S) :- sum(T,S2), S = H + S2.`

☐ `sum([],0).`  
☐ `sum([H|T], S) :- sum(T,S2), S = H + S2.`

☒ `sum([],0).`  
☒ `sum([H|T], S) :- sum(T,S2), S is H + S2.`

☐ none of the above.

Quiz Score: **100** out of 100