1 The Module `zebra_puzzle.pl`

`next_to(?X, ?Y, ?List)`   zebra_puzzle.pl

The element *X* is next to *Y* in *List*, i.e. *Y* directly follows *X* or *X* directly follows *Y*. Not to be confused with nextto/3.

```
:- module(zebra_puzzle).
next_to(X, Y, List) :-
    ( nextto(X, Y, List)
    ; nextto(Y, X, List)
    ).
```

`puzzle(?Houses)`   zebra_puzzle.pl

The puzzle consists of 15 clues about *Houses*.

```
puzzle(Houses) :-
```

1. There are five houses.

```
length(Houses, 5),
```

2. The Englishman lives in the red house.

```
member(house(englishman, red, _, _, _), Houses),
```

3. The Spaniard owns the dog.

```
member(house(spainard, _, dog, _, _), Houses),
```

4. Coffee is drunk in the green house.

```
member(house(_, green, _, coffee, _), Houses),
```

5. The Ukrainian drinks tea.

```
member(house(ukrainian, _, _, tea, _), Houses),
```

6. The green house is immediately to the right of the ivory house.

```
nextto(house(_, ivory, _, _, _), house(_, green, _, _, _), Houses),
```

7. The Old Gold smoker owns snails.

```
member(house(_, _, snails, _, old_gold), Houses),
```

8. Kools are smoked in the yellow house.

```
member(house(_, yellow, _, _, kool), Houses),
```

9. Milk is drunk in the middle house.

```
[_, _, house(_, _, _, milk, _), _, _] = Houses,
```

10. The Norwegian lives in the first house.

```
[house(norwegian, _, _, _, _) | _ ] = Houses,
```

11. The man who smokes Chesterfields lives in the house next to the man with the fox.

```
next_to(house(_, _, _, _, chesterfield), house(_, _, fox, _, _), Houses),
```

12. Kools are smoked in the house next to the house where the horse is kept.

```
next_to(house(_, _, _, _, kool), house(_, _, horse, _, _), Houses),
```

13. The Lucky Strike smoker drinks orange juice.

```
member(house(_, _, _, orange_juice, lucky_strike), Houses),
```

14. The Japanese smokes Parliaments.

```
member(house(japanese, _, _, _, parliament), Houses),
```

15. The Norwegian lives next to the blue house.

```
next_to(house(norwegian, _, _, _, _), house(_, blue, _, _, _), Houses).
```

---

zebra_owner(-Owner)   zebra_puzzle.pl

Determine the nationality of the zebra *Owner*.

```
zebra_owner(Owner) :-
    puzzle(Houses),
    member(house(Owner, _, zebra, _, _), Houses).
```

---

water_drinker(-Drinker)   zebra_puzzle.pl

Determine the nationality of the water *Drinker*.

```
water_drinker(Drinker) :-
    puzzle(Houses),
    member(house(Drinker, _, _, water, _), Houses).
```