Do the same rules for Independence hold for FACTS?

According to the slides:

No, the same rules of independence do not hold for facts. When several facts with matching parameters are set they do not overwrite each other. Regardless of whether there is an empty line between them or not.

Code

```
(assert(color(banana, yellow))).
(assert(color(banana, teal))).

(assert(color(banana, orange))).
(assert(color(banana, orange))).
(assert(color(orange_tree, orange))).

findall(X,color(banana,X),Z).
findall(X,color(X, orange),Y).

halt.
```

Output

```
Z = [yellow, teal, orange, orange].

Y = [banana, banana, orange_tree].
```

According to on-line research:

Yes, the same rules of independence hold for facts. The code below demonstrates that it is possible to have several facts with matching arguments and have them occupy separate entries in the database just like rules are treated. Each variable inside a rule/fact has a consistent value only on the same rule/fact. Outside of it, variables with the same name are treated like two independent variables.

Code

```
(assert(color(banana, yellow))).
(assert(color(banana, teal))).

(assert(color(banana, orange))).
(assert(color(banana, orange))).
(assert(color(orange_tree, orange))).

(assert((eats(bob, X) :-color(X, green), vegetable(X)))).
(assert((eats(bob, X) :-color(X, yellow), fruit(X)))).

(assert((eats(bob, X) :-color(X, green), fruit(X)))).

listing.

halt.
```

Output

```
...

:-dynamic color/2.

color(banana, yellow).
color(banana, teal).
color(banana, orange).
color(banana, orange).
color(orange_tree, orange).

:-dynamic eats/2.

eats(bob, A) :-
        color(A, green),
        vegetable(A).
eats(bob, A) :-
        color(A, yellow),
        fruit(A).
eats(bob, A) :-
        color(A, green),
        fruit(A).

...
```