Variable Dry Run Table Puzzles (for Python)

The simplest and most reliable way to work out what a code fragment does is to draw a table. This is called tracing or dry running. Let's work out what the following fragment does:

```
x = 5
y = 7
x = 10
y = x
```

Each time we come across a new variable we label a new column with that variable.

The first line creates a column for x and the second a column for y.
We get this table:

x	y

Each assignment places a value in the appropriate column.

So continuing to dry run where we left off

```
x = 5
y = 7
```

We first put a value 5 in to the column for x.

Then executing the next instruction, we put a 7 in the column for y

x	y
5	
	7

We cross out any old value in a column when that variable is assigned to again.

So continuing to dry run the next part where we left off

```
x = 10
```

We cross out the 5 in the column for x and put 10 in that column instead.

x	y
~~5~~	
	7
10	

To get the value of a variable we take the last value put there (i.e. ignore values crossed out)

So if the above program continues

```
y = x
```

We go to the x column and find the LAST value placed there which is a 10
We place the 10 in the y column, crossing out the 7 that was already there.

x	y
~~5~~	
	~~7~~
10	
	10

The final values in each variables (here x and y) are then the last uncrossed out ones, so here x is 10 and y is 10 after the fragment has executed.

1. **Fill in the dry run table then state the final values stored in each variable after the code fragment has executed.**

```
x = 5
y = 7
x = y
```

x	y

The final value of x is _____ The final value of y is _____

2. **Fill in the dry run table then state the final values stored in each variable after the code fragment has executed.**

```
x = 5
y = 7
y = x
```


The final value of x is _____ The final value of y is _____

3. **Fill in the dry run table then state the final values stored in each variable after the code fragment has executed.**

```
y = 7
x = 5
y = x
```


The final value of x is _____ The final value of y is _____

4. **Fill in the dry run table then state the final values stored in each variable after the code fragment has executed.**

```
red = "red"
blue = "yellow"
red = blue
```


The final value of red is _____ The final value of blue is _____

5. **Fill in the dry run table then state the final values stored in each variable after the code fragment has executed.**

```
y = 7
x = 5
y = x
x = 3
```


The final value of x is _____ The final value of y is _____

6. **Fill in the dry run table then state the final values stored in each variable after the code fragment has executed.**

```
x = 1
y = 2
z = 3
y = x
x = y
```


The final value of x is _____ The final value of y is _____ The final value of z is _____

7. **Fill in the dry run table then state the final values stored in each variable after the code fragment has executed.**

```
one = 1
two = 3
three = 2
one = two
two = three
```


The final value of one is _____

The final value of two is _____

The final value of three is _____

8. **Fill in the dry run table then state the final values stored in each variable after the code fragment has executed.**

```
a = 9
b = 7
c = 8
a = c
b = a
c = b
```


The final value of a is _____ **The final value of b is** _____ **The final value of c is** _____

9. **Fill in the dry run table then state the final values stored in each variable after the code fragment has executed.**

```
a = 1
b = 2
c = 3
c = b
c = a
a = b
```


The final value of a is _____ The final value of b is _____ The final value of c is _____

10. **Fill in the dry run table then state the final values stored in each variable after the code fragment has executed.**

```
x = 3
y = 2
z = 3
z = y
y = x
x = z
```


The final value of x is _____ The final value of y is _____ The final value of z is _____