Writing Python Simulations

Puzzle #1: Find the expected output of the following code:

(a):

Python:
```python
red = 4
if red < 3:
    red = 5
if red > 3:
    red = 3

print(f"Value of red: {red}")
```

Description of Output:

(b):

Python:
```python
coin = random.choice(["head", "tail"])
if coin == "head":
    print("You won!")
else:
    print("You lost.")
```

Description of Output:

(c):

Python:
```python
data = []
for i in range(1000):
    guess = random.randint(1, 10)
    if guess == 7:
        d = {"win": 1}
    else:
        d = {"win": 0}
    data.append(d)
df = pd.DataFrame(data)
```

Description of Program:

Puzzle #2: Write the Python code to simulate 100,000 generic tests to test someone's relation to Taylor Swift. The test has the following parameters: There is a 99% probability that an individual related to Taylor Swift will get a positive result ("true positive"). There is a 6% probability that an individual NOT related to Taylor Swift will get a positive result ("false positive"). About 1% of the world population is related to Taylor Swift.

Algorithm:	
Simulation:	

Analysis:

(a): In our simulation of 100,000 tests, how many people were **actually** related to Taylor?

(b): In our simulation of 100,000 tests, how many people **tested** to be **related** to Taylor?

(c): How many people tested positive to be related, but weren't actually related?