Relational (Testing Cond)

>

<

>=

<=

==

!=

= v/s ==

① Assignment op

```
int a=10, b=7;
b = a;
printf("%d %d", a, b);
```

10 10

① Equality Testing op

```
int a=10,b=7,c;
c = a == b;
printf("%d %d %d", a,b,c);
```

True → 1
False → 0

10 7 0

```
int a = 5, b = 9, c;
```

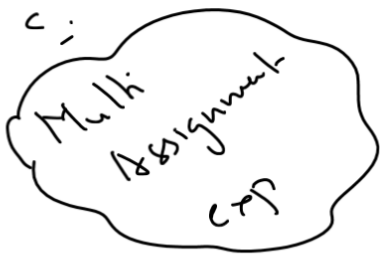
① `c = a > b;`

```
printf("%d %d %d", a, b, c);
        5  9  0
```

② `c = a < b;`

```
printf("%d %d %d", a, b, c);
        5  9  1
```

```
int a = 10, b = 7, c;
```



`c = a = b;`

```
printf("%d %d %d", a, b, c);
        7  7  7
```

```
int a, b, c, d, e;

a = 10;
b = 10;
c = 10;
d = 10;
e = 10;
```

OK

```
a = b = c = d = e = 10;
```

This is also multi-assignment expression.

```
int a, b, c, d, e;

a = b = c = d = 10 = e;
```

✗ l-value reqd

```
a==b=c;
```

✗

```
int x;
x=10=10;
x=10=11;
x=10==10;
x=10==11;
```

✗

✗

✓

✓

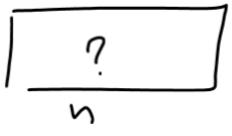
```
int a, b;

a = b;

printf("%d %d", a, b);
```

garbage (Any unpredictable number)

```
int n;
```




```
printf("%d", n);
```

garbage value

```
n = 10;
printf("%d", n);
```