Explanation A

The method *getFormat(date)* calculates the format of a given date.

The assertion in line 8 checks if the method outputs the same value as in the variable *format*.

Internally, the method computes and returns the value of the variable *result*. However,
1. In V1, *result* is assigned in line 15 to be *"yyyy"* as *date.year* causes the *"if"* statement in line 12 to evaluate to *false*.

2. In V2, despite the change in line 12, the *"if"* statement in line 12 also evaluates to *false*.

The variable *result* is, thus, also assigned in line 15 but, due to the change in this line, its value is *"yy"*.
Then, in both versions, the variable *result* is further reassigned in
lines 17, 19, and 21, based on its previous value and the output of the methods *getMonthFormat* (line 17), *getDayFormat* (line 19), and *getTimeFormat* (line 21) applied on *tokenLen*.
In summary, the change leads to a difference in computing the value of *result* in V1 and V2:

in V1, *result* for such inputs is prefixed with *"yyyy"* and is equal to the prefix of the variable *format* defined in line 2. In V2, it is rather prefixed with *"yy"*.
This difference causes the output of *getFormat(date)* to differ from *format* in line 8.

Explanation B

The method *getFormat(date)* calculates the format of a given date.

The assertion in line 8 checks if the method outputs the same value as in the variable *format*.

Internally, the method computes and returns the value of the variable *result*. However,
1. In V1, *result* is assigned in line 15 to be *"yyyy"* as *date.year* causes the *"if"* statement in line 12 to evaluate to *false*.

2. In V2, despite the change in line 12, the *"if"* statement in line 12 also evaluates to *false*.

The variable *result* is, thus, also assigned in line 15 but, due to the change in this line, its value is *"yy"*.
Then, in both versions, the variable *result* is further reassigned in
line 21 to be a function of its previous value.

In summary, the change leads to a difference in computing the value of *result* in V1 and V2:

in V1, *result* for such inputs is prefixed with *"yyyy"* and is equal to the prefix of the variable *format* defined in line 2. In V2, it is rather prefixed with *"yy"*.
This difference causes the output of *getFormat(date)* to differ from *format* in line 8.

Explanation C

The method *getFormat(date)* calculates the format of a given date.
In this code, the year of the input parameter *date* is initialized in line 4 as "003".
The assertion in line 8 checks if the method outputs the same value as in the variable *format*,
i.e., "yyyyMMddHH:mm:ss" set in line 2.
Internally, the method computes and returns the value of the variable *result*. However,
1. In V1, *result* is assigned in line 15 to be *"yyyy"* as *date.year* causes the *"if"* statement in line 12 to evaluate to *false:*
   date.year is *"003"* and, thus, its length is not equal to 2.
2. In V2, despite the change in line 12, the *"if"* statement in line 12 also evaluates to *false:*
   date.year is *"003"* and, thus, its length is not ≧ 4.

The variable *result* is, thus, also assigned in line 15 but, due to the change in this line, its value is *"yy"*.
Then, in both versions, the variable *result* is further reassigned in
line 21 to be a function of its previous value.

In summary, the change leads to a difference in computing the value of *result* in V1 and V2
for inputs with years given as 3 digits, e.g., "003":
in V1, *result* for such inputs is prefixed with *"yyyy"* and is equal to the prefix of the variable *format* defined in line 2. In V2, it is rather prefixed with *"yy"*.
This difference causes the output of *getFormat(date)* to differ from *format* in line 8.

Notations: Colored backgrounds highlight the differences between the views.

FYI: Views are given below again, for your reference.

View A

V1

```
1  public static void main(String[] args){
2
3
4
5
6
7
8    assertEquals(format, getFormat(date));
9  }
10 public String getFormat(Date date){
11
12   if (date.year.length() == 2) [false]
13     result = "yy";
14   else
15     result = "yyyy";
16
17   result = result+getMonthFormat(tokenLen);
18
19   result = result+getDayFormat(tokenLen);
20
21   result = result+getTimeFormat(tokenLen);
22   return result;
23 }
```

Update

V2

```
1  public static void main(String[] args){
2
3
4
5
6
7
8    assertEquals(format, getFormat(date));
9  }
10 public String getFormat(Date date){
11
12   if (date.year.length() >= 4) [false]
13     result = "yyyy";
14   else
15     result = "yy";
16
17   result = result+getMonthFormat(tokenLen);
18
19   result = result+getDayFormat(tokenLen);
20
21   result = result+getTimeFormat(tokenLen);
22   return result;
23 }
```

View B

V1

```
1  public static void main(String[] args){
2
3
4
5
6
7
8    assertEquals(format, getFormat(date));
9  }
10 public String getFormat(Date date){
11
12   if (date.year.length() == 2) [false]
13     result = "yy";
14   else
15     result = "yyyy";
16
17
18
19
20
21   result = Func1(result);
22   return result;
23 }
```

Update

V2

```
1  public static void main(String[] args){
2
3
4
5
6
7
8    assertEquals(format, getFormat(date));
9  }
10 public String getFormat(Date date){
11
12   if (date.year.length() >= 4) [false]
13     result = "yyyy";
14   else
15     result = "yy";
16
17
18
19
20
21   result = Func1(result);
22   return result;
23 }
```

View C

V1

```
1  public static void main(String[] args){
2    String format =  "yyyyMMddHH:mm:ss";
3    Date date = new Date();
4    date.year = "003";
5
6
7
8    assertEquals(format, getFormat(date));
9  }
10 public String getFormat(Date date){
11
12   if (date.year.length() == 2) [false]
13     result = "yy";
14   else
15     result = "yyyy";
16
17
18
19
20
21   result = Func1(result);
22   return result;
23 }
```

Update

V2

```
1  public static void main(String[] args){
2    String format =  "yyyyMMddHH:mm:ss";
3    Date date = new Date();
4    date.year = "003";
5
6
7
8    assertEquals(format, getFormat(date));
9  }
10 public String getFormat(Date date){
11
12   if (date.year.length() >= 4) [false]
13     result = "yyyy";
14   else
15     result = "yy";
16
17
18
19
20
21   result = Func1(result);
22   return result;
23 }
```