



Tutorial Link <https://codequotient.com/tutorials/if statement mistakes/5a22a06dc66cfe38f296224f>

TUTORIAL

if statement mistakes

Chapter

1. if statement mistakes

One common problem with if statement is if you put a semicolon after if condition, things will change considerable. For example,

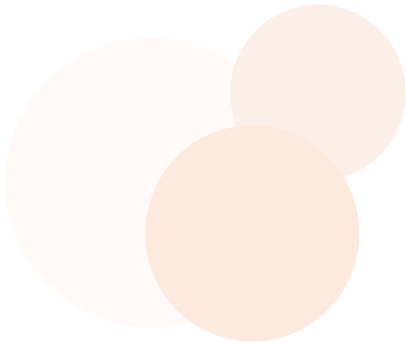
```
1  #include <stdio.h>
2
3  int main()
4  {
5      int i = 6;
6      if(i == 5);
7      printf ("It will be executed, as if already ends with
8      semicolon.\n");
9      return 0;
10 }
```

The statement `if(i ==5);` is ended with a semicolon. It is interpreted as below code by compiler:

```
1  #include <stdio.h>
2
3  int main()
4  {
5      int i = 6;
```

```
6   if(i == 5)
7       ;           // it is valid statement, that do nothing.
      NULL statement.
8   printf ("It will be executed, as if already ends with
      semicolon.\n");
9   return 0;
10  }
11
```

So the if condition will be evaluated and if it evaluates to true the NULL statement will be executed and after that printf() function will be called. So this can be tiny mistake while typing and cause the undesired response.



Tutorial by codequotient.com | All rights reserved, CodeQuotient 2023

codequotient.com