

3.1.a

The errors in this are located in case 6 and case 7.

```
In case 7:  
cout<<"Today is Sunday";
```

There is an error in the line above that deals with case 7 where they are forgetting The "break;" line of code that should be in each different case

In case 6, there should not be singular quotation marks because the cases are based Off different integer inputs rather than characters.

Case 6:

...

Case 7:

3.1.b

The error in this case is located in line 1 that they are not including the code for a Input of the week and they are not storing that input into the variable "day".

So instead of `int day = 4; //this will always run the switch for integer 4` They need to ask the user for their input on the integer for the day of the week and Then use that answer in the switch case.

Another error is forgetting the : for the default statement

Another error is brackets around the switch statement

3.1.c

The first error is located in the else if statement(parameter). Right now the code Says else if (x=90), which is assigning 90 to x. Instead it should be changed to Else if (x==90).

The second error is located in the else statement where the person who wrote the code Forgot to add an "e" on the else statement. `els ---> else{}`

The third error is that the variable "angle" should replace the variable "x" all around

3.1.d

The first error is seen throughout the whole code where they are forgetting to put {} After an if/else statement. To fix this, they change if -----> if(parameter){} And else ----->else{}

The second error is at the end in the cout statement. There should be 2 << instead Of only one "<". So it should be `cout<<value<<endl;`

The third error is on the include iostream. It should be #include <iostream>

The fourth possible error is that they are missing a test case. Since there are multiple statements , you need an if statement to check if

```
if(x==0){  
    if(y==1){  
        value = 1;}  
    Else{  
        value = 0;}  
}
```

Basically, the original code is missing a case for if x ==0 and y==1 then return 1. This case may not be necessary because the code already defines what x and y equal to when they initialize those variables.