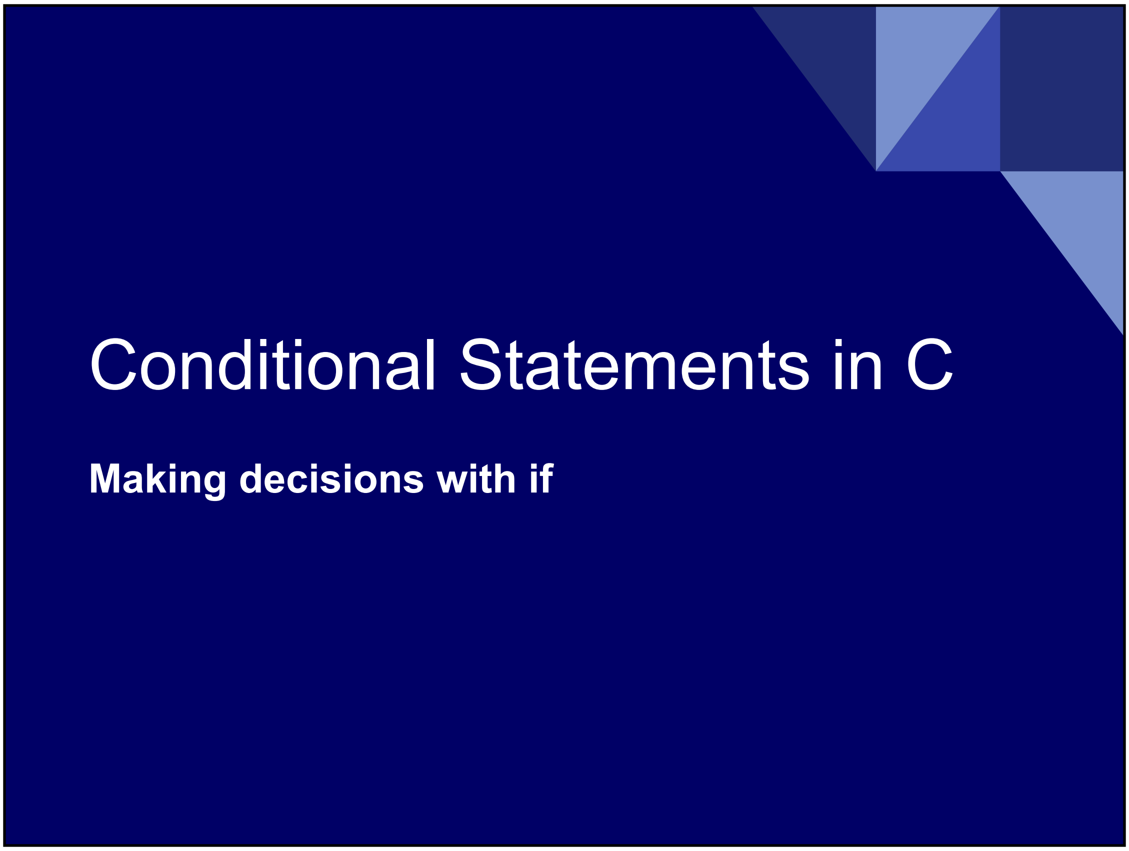
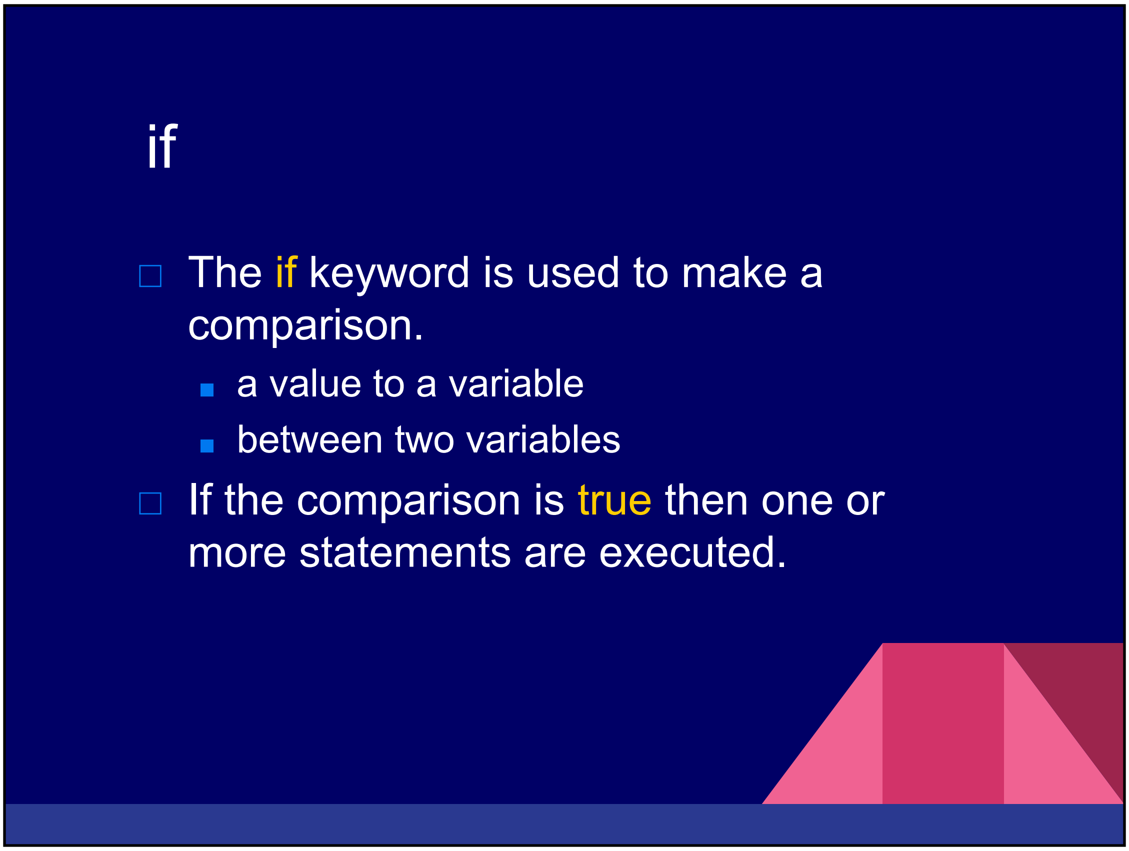


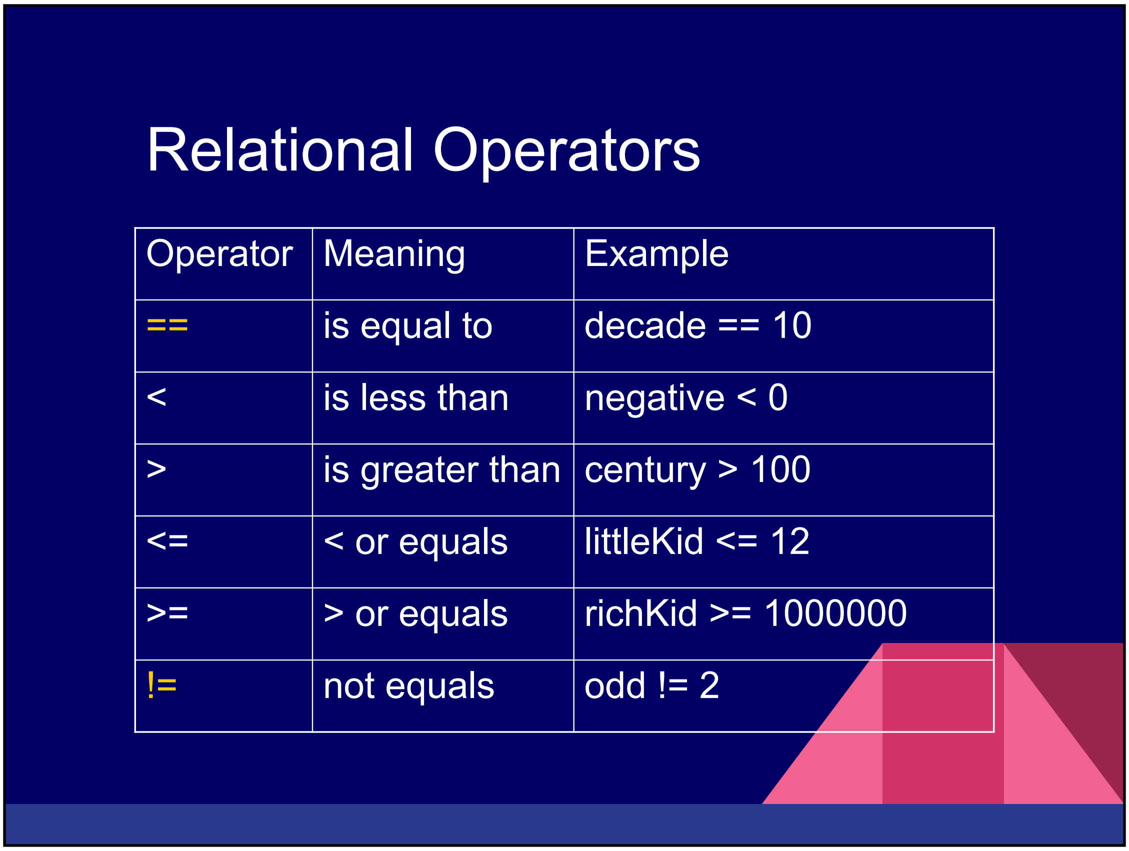
Conditional Statements in C

Making decisions with if



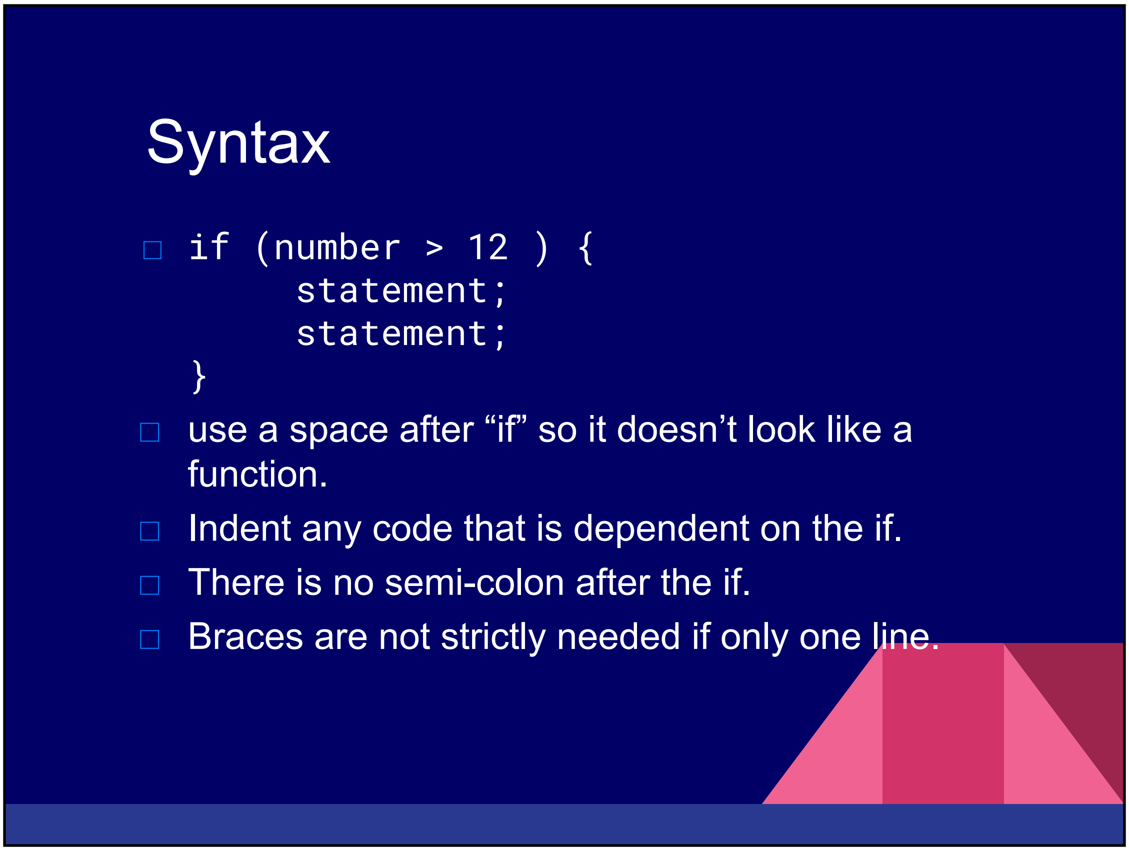
if

- □ The if keyword is used to make a comparison.
  - ■ a value to a variable
  - ■ between two variables
- □ If the comparison is true then one or more statements are executed.



Relational Operators

Operator	Meaning	Example
==	is equal to	decade == 10
<	is less than	negative < 0
>	is greater than	century > 100
<=	< or equals	littleKid <= 12
>=	> or equals	richKid >= 1000000
!=	not equals	odd != 2



Syntax

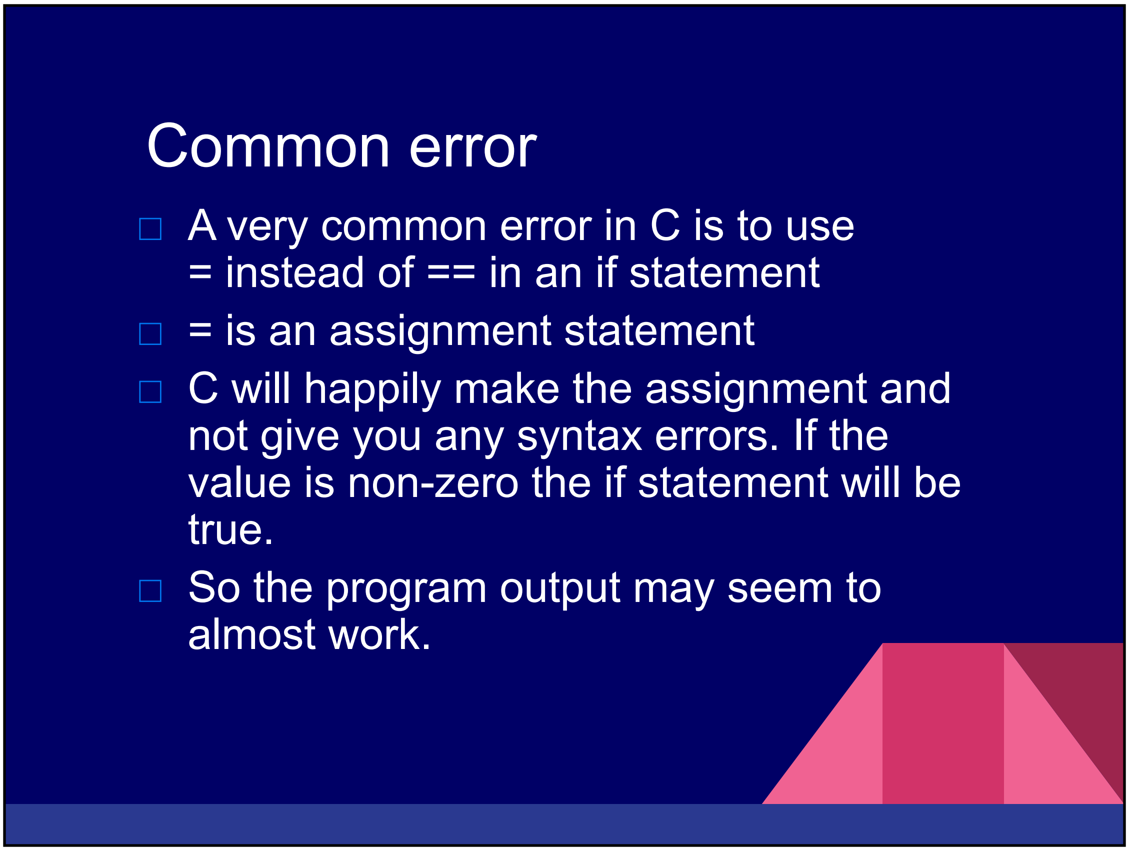
□
```
if (number > 12 ) {
        statement;
        statement;
}
```

□ use a space after “if” so it doesn’t look like a function.

□ Indent any code that is dependent on the if.

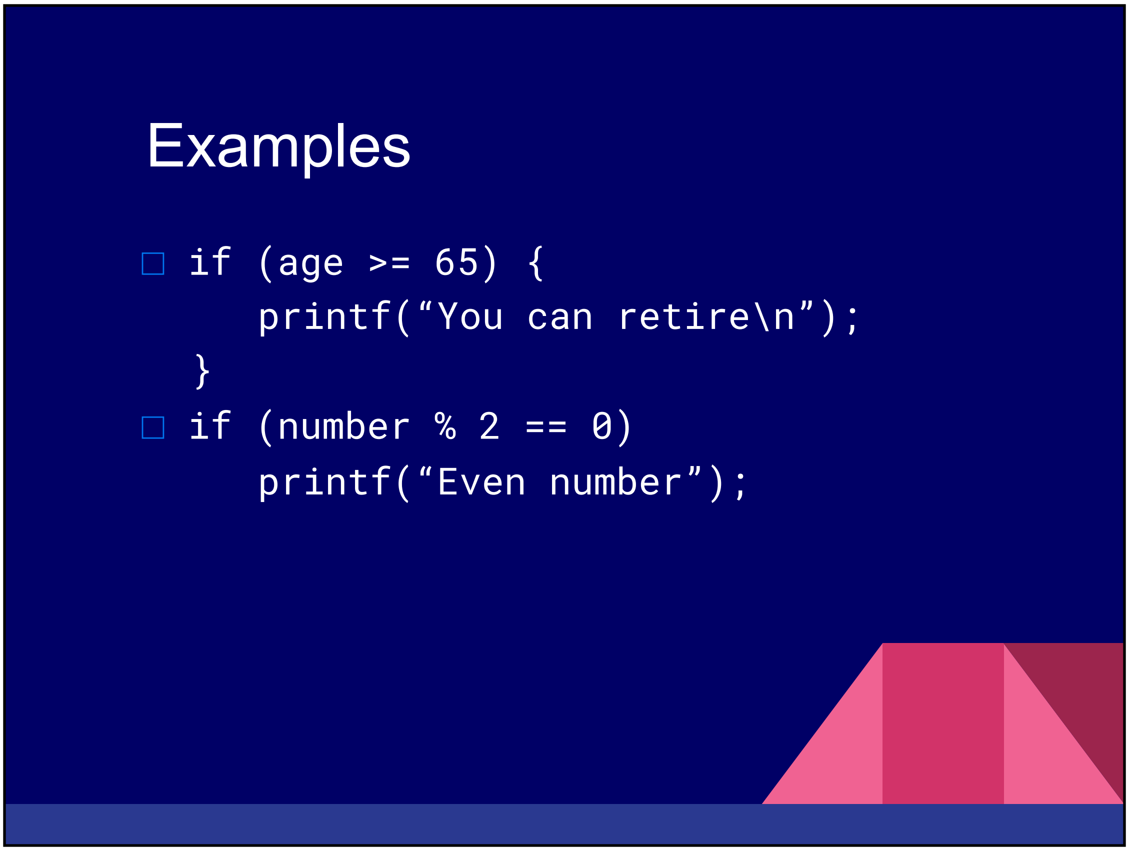
□ There is no semi-colon after the if.

□ Braces are not strictly needed if only one line.



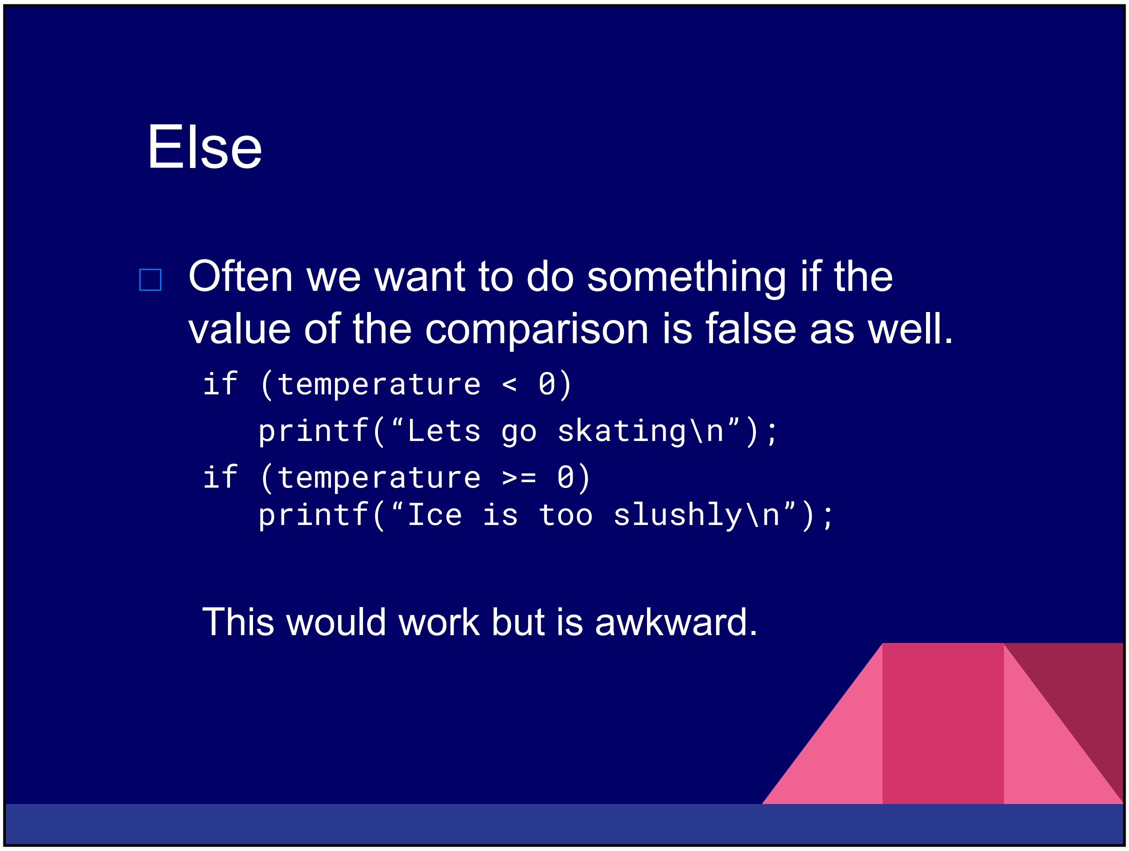
Common error

- A very common error in C is to use = instead of == in an if statement
- = is an assignment statement
- C will happily make the assignment and not give you any syntax errors. If the value is non-zero the if statement will be true.
- So the program output may seem to almost work.



Examples

```
□ if (age >= 65) {
     printf(“You can retire\n”);
  }
□ if (number % 2 == 0)
     printf(“Even number”);
```

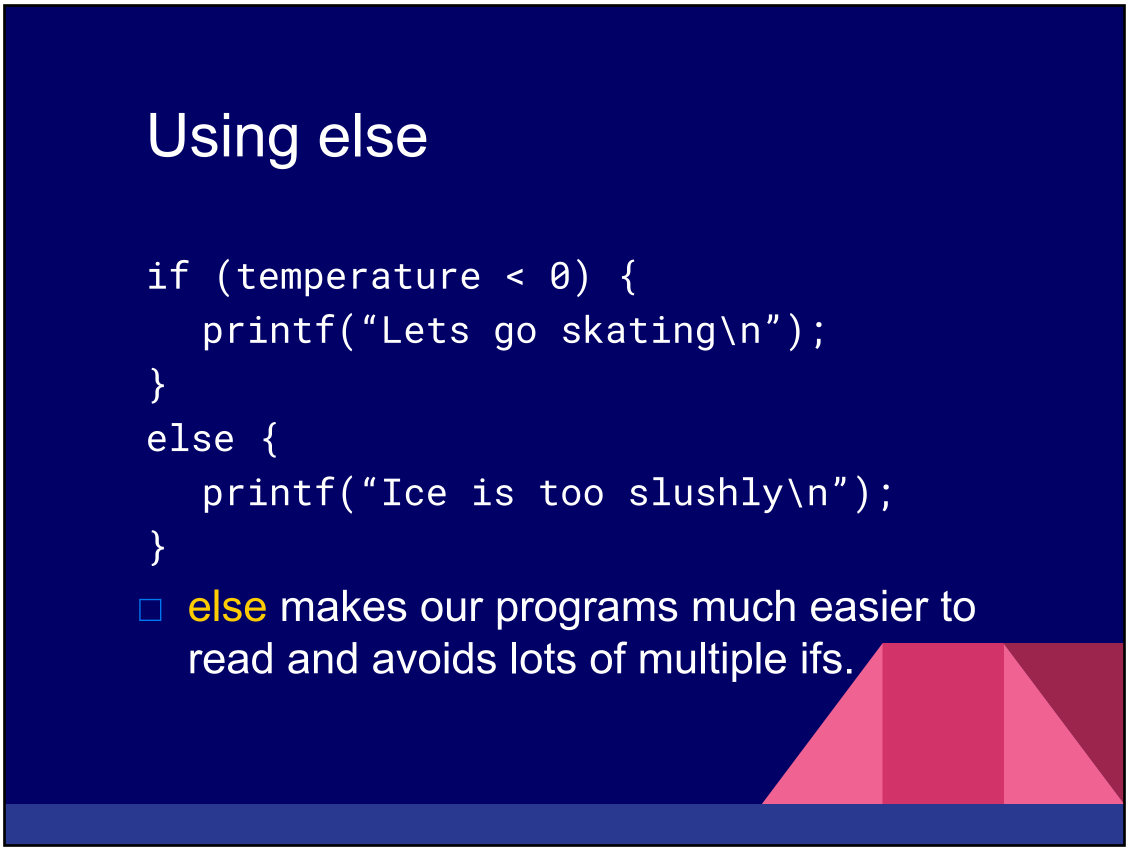


Else

- □ Often we want to do something if the value of the comparison is false as well.

```
if (temperature < 0)
   printf(“Lets go skating\n”);
if (temperature >= 0)
   printf(“Ice is too slushly\n”);
```

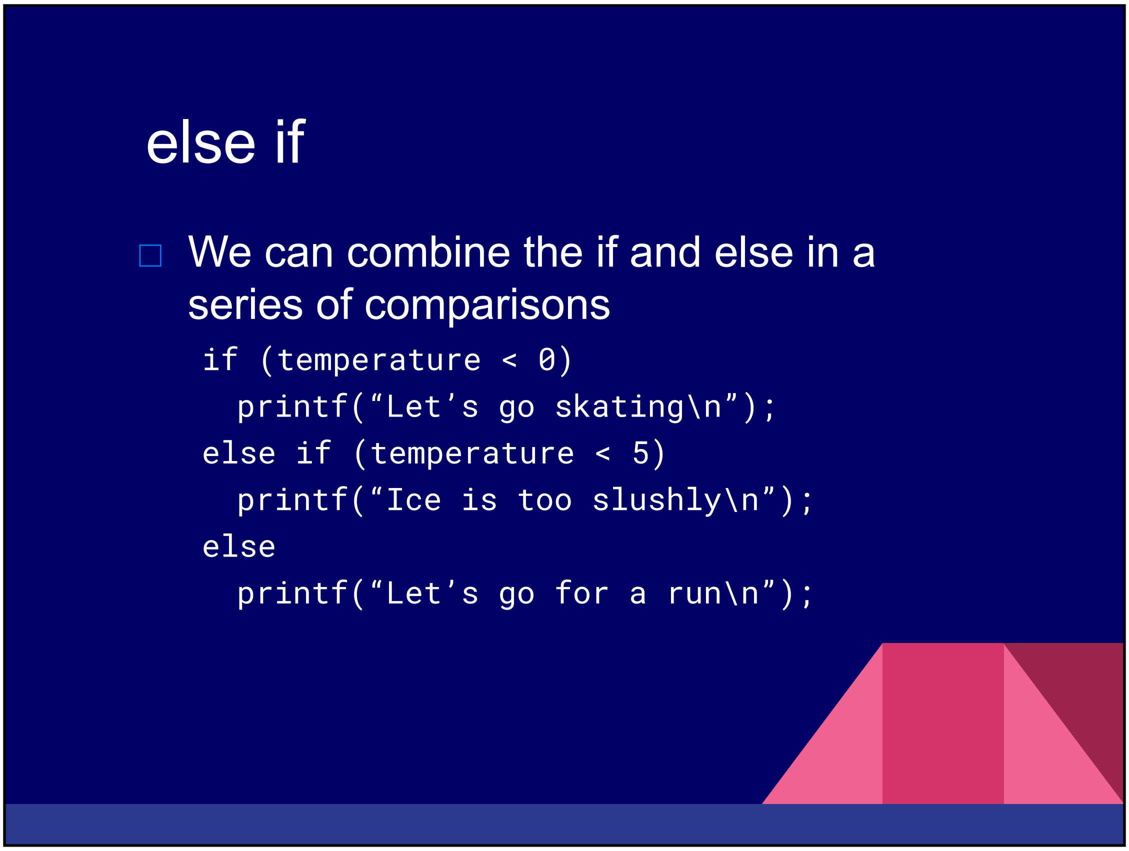
This would work but is awkward.



Using else

```
if (temperature < 0) {
   printf(“Lets go skating\n”);
}
else {
   printf(“Ice is too slushly\n”);
}
```

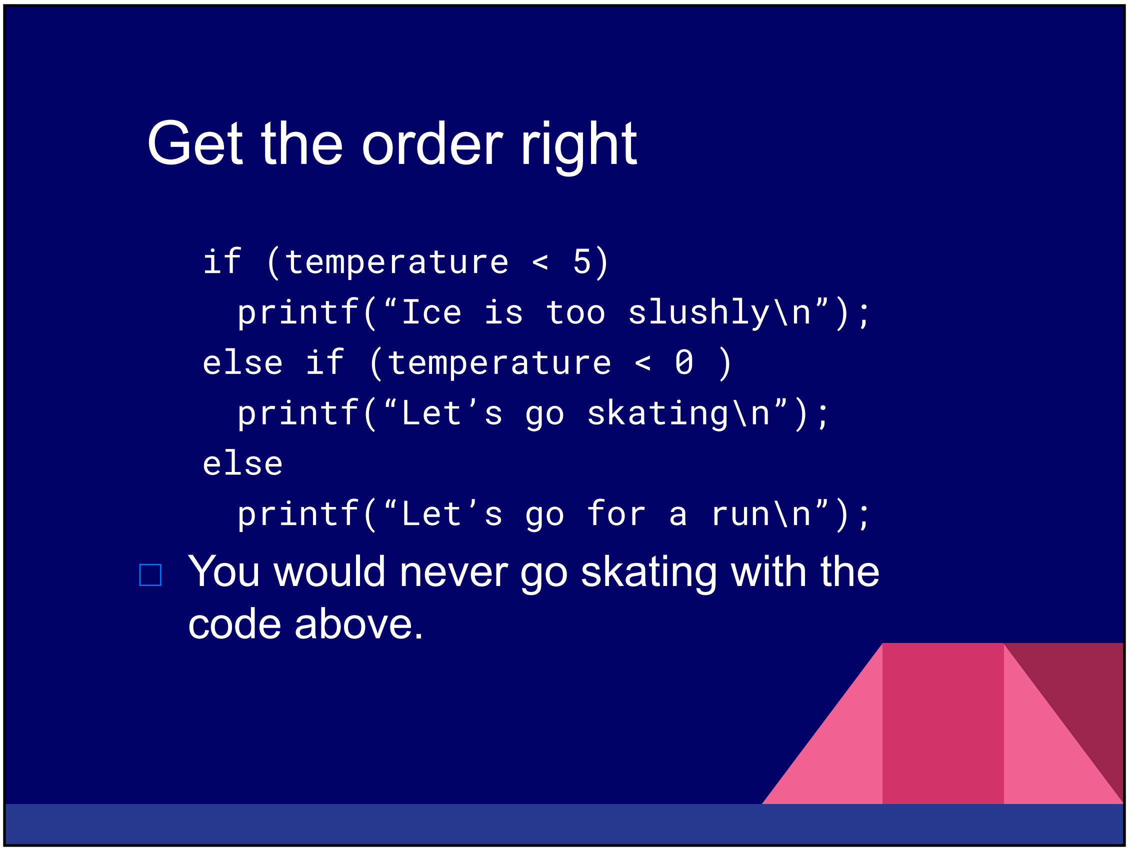
- □ else makes our programs much easier to read and avoids lots of multiple ifs.



else if

- We can combine the if and else in a series of comparisons

```
if (temperature < 0)
  printf(“Let’s go skating\n”);
else if (temperature < 5)
  printf(“Ice is too slushly\n”);
else
  printf(“Let’s go for a run\n”);
```



Get the order right

```
if (temperature < 5)
  printf(“Ice is too slushly\n”);
else if (temperature < 0 )
  printf(“Let’s go skating\n”);
else
  printf(“Let’s go for a run\n”);
```

- You would never go skating with the code above.

Characters

We can compare characters

-
  ```
  if (letter > ‘M’)
      printf(“See Mrs. Smith\n”);
  ```
- Letters are compared based on their Ascii code. ‘a’ is greater than ‘Z’.
- toupper and tolower functions can be used to change the case of a character.

Using toupper / tolower

- You can convert this:

```
if (answer == 'Y' || answer == 'y') {
   //do something
}
```

into

```
if (tolower(answer) == 'y') {
   //do something
}
```