

# Switch Case vs If Else

---

## Switch Case vs If Else

Java allows you to use either `switch case` or a series of `if else` to handle decisions with multiple outcomes. There are a couple reasons you would use one over the other.

### #1: If Else is used for ranges - Switch Case is for values

Switch case can only check for equality (i.e. `num == 5`) so if you need to check for a range (i.e. `num > 50 && num <= 60`).

```
int grade = 62;
int letterGrade = grade / 10;
switch(letterGrade) {
    case 10: case 9: System.out.print("A");
        break;
    case 8: System.out.print("B");
        break;
    case 7: System.out.print("C");
        break;
    case 6: System.out.print("D");
        break;
    default : System.out.print("F");
}

int grade = 62;
if(grade < 60) {
    System.out.println("F"); }
else if(grade < 70) {
    System.out.println("D"); }
else if(grade < 80) {
    System.out.println("C"); }
else if(grade < 90) {
    System.out.println("B"); }
else if(grade <= 100) {
    System.out.println("A"); }
```

[.guides/img/Switch-Case-vs-If-Else](#)

#### ▼ What is case 10: case 9:?

Sometimes, the code for multiple cases is the same. Instead of repeating code, you can list multiple cases before the code. Here is another example:

```

int month = 2;
int year = 2000;
int numDays = 0;

switch (month) {
    case 1: case 3: case 5:
    case 7: case 8: case 10:
    case 12:
        numDays = 31;
        break;
    case 4: case 6:
    case 9: case 11:
        numDays = 30;
        break;
    case 2:
        if ((year % 4 == 0) &&
            !(year % 100 == 0))
            || (year % 400 == 0))
            numDays = 29;
        else
            numDays = 28;
        break;
    default:
        System.out.println("Invalid month.");
        break;
}
System.out.println("Number of Days = "
    + numDays);

```

In some cases, as shown above, you can exploit patterns to force ranges into a switch case, but frequently that is not possible and it also makes the code less readable. For example, above, the user has to realize that letterGrade is using integer division to retrieve the ten's place of the original grade.

```

int grade = 62;
int letterGrade = grade / 10;
switch(letterGrade) {
    case 10: case 9: System.out.print("A");
        break;
    case 8: System.out.print("B");
        break;
    case 7: System.out.print("C");
        break;
    case 6: System.out.print("D");
        break;
    default : System.out.print("F");
}

```

[Code Visualizer](#)

## #2: If Else is used for handling multiple variables

Switch case can only compare against values - not variables. For example, if you wanted to compare the inputted day of the week with the current day of the week you would need to use an if else. Switch can handle values (dayOfWeek == "Sunday") but not variables (dayOfWeek == today).

### #3: If Else is used for compound conditionals

To check multiple conditions, an if else is needed.

An example would be a multiple choice grader:

```
int studentAnswer = 3;
String feedback1 = "This answer is wrong because...";
String feedback2 = "This answer is correct! You know this
because...";
String feedback3 = "This answer is wrong. While the first part
is correct...";
String feedback;

int correctAnswer = 2;
int points = 0;

switch(studentAnswer) {
    case 1: feedback = feedback1; break;
    case 2: feedback = feedback2; break;
    case 3: feedback = feedback3; break;
    default: feedback = "Invalid answer choice";
}

System.out.println(feedback);
```

challenge

#### Convert to an If Else

- Change the switch statement above into an if else
- Add a check to see if studentAnswer == correctAnswer
- If the student's answer is correct, increment (++) the points variable.
- Print out the students earned points at the end of the program using the points variable

[Code Visualizer](#)