More switch statement

In the last video, we looked at the switch statement, and how we can use it, instead of an if statement, if all our test conditions are testing a single variable's state.

The switch statement has seen quite a few updates through the years.

So in this video, I'm going to explore some of the new features of the switch statement.

I'll be looking at a couple of these now, and reviewing others a bit later in the course, that require some knowledge of classes and object-oriented programming.

Traditional Switch Statement vs. Enhanced Switch Statement

Traditional Switch Statement	Enhanced Switch Statement

```java
switch (switchValue) {
    case 1:
        System.out.println("Value was 1");
        break;
    case 2:
        System.out.println("Value was 2");
        break;
    case 3: case 4: case 5:
        System.out.println("Was a 3, a 4, or a 5");
        System.out.println("Actually it was a " + switchValue);
        break;
    default:
        System.out.println("Was not 1, 2, 3, 4, or 5");
        break;
}
```

```java
switch (switchValue) {
    case 1 -> System.out.println("Value was 1");
    case 2 -> System.out.println("Value was 2");
    case 3, 4, 5 -> {
        System.out.println("Was a 3, a 4, or a 5");
        System.out.println("Actually it was a " + switchValue);
    }
    default -> System.out.println("Was not 1, 2, 3, 4, or 5");
}
```

Traditional Switch Statement vs. Enhanced Switch Expression

Traditional Switch Statement used in a method, returning values	Enhanced Switch Expression

`switch (month) {`
`  case "JANUARY":`
`  case "FEBRUARY":`
`  case "MARCH":`
`  return "1st";`
`  case "APRIL":`
`  case "MAY":`
`  case "JUNE":`
`  return "2nd";`
`  case "JULY":`
`  case "AUGUST":`
`  case "SEPTEMBER":`
`  return "3rd";`
`  case "OCTOBER":`
`  case "NOVEMBER":`
`  case "DECEMBER":`
`  return "4th";`
`}`
 	
`return "bad";`
	`return switch (month) {`
	`  case "JANUARY", "FEBRUARY", "MARCH" -> "1st";`
	`  case "APRIL", "MAY", "JUNE" -> "2nd";`
	`  case "JULY", "AUGUST", "SEPTEMBER" -> "3rd";`
	`  case "OCTOBER", "NOVEMBER", "DECEMBER" -> "4th";`
	`  default -> "bad";`
	`};`

When to use yield in a switch

Your switch statement is being used as a switch expression returning a value.

Your case label uses a code block, with opening and closing curly braces.

This code	is implicitly translated to
`-> "1st";`	`-> { yield "1st"; }`