**Learning Objectives**

* To grasp uses of if-else statement;
* To grasp uses of switch statement.

**Learning Contents**

* There is such an episode on Internet.
  + The wife said, “Could you go to the food market to buy a watermelon? If you see tomato, then buy two. ” A moment later, the husband came back with two watermelons. The wife asked, “Why did you buy two watermelons?” The programmer husband answered, “Because I see tomato at the food market!” People normally feel consumed about it. After studying this section, you will say, “Well……There is no problem!”
* To understand conditional branch statement
  + First of all, let’s learn about what is a conditional branch statement for.
  + A conditional branch statement execute different **actions** based on different **conditions**.
  + We will translate the above episode by standing in the shoes of a programmer. Behind "if" is a "condition" and behind "condition" is corresponding "action". Then we get the following two sets of “condition -> action”.
    - When the condition is satisfied: Saw the tomato -> Action 1：Buy two watermelons
    - When the condition is not satisfied: Did not see the tomato -> Action 2：Buy one watermelon
  + Next, let’s see how a conditional branch statement is used in JavaScript.
* **if** statement
  + Syntax statement of if statement is as below
  + **if (condition){**
  + **// Statement block to be executed when condition==true**
  + **}**
  + It should be noted that, the identity of the condition is a judgment statement. There are only two results, true or false. That is to say, the condition is either satisfied or not satisfied.
  + For example:
  + **var seeTomato = 'yes';**
  + **if (seeTomato == 'yes'){**
  + **console.log('Buy two watermelons!');**
  + **}**
  + ***//Output Buy two watermelons!***
  + You might ask: if the condition is not satisfied, how the codes will be executed? The answer is that be statement behind the if statement block will be executed by jumping over the if statement block.
  + For example:
  + **var seeTomato = 'no';**
  + **if (seeTomato == 'yes'){**
  + **console.log('Buy two watermelons!');**
  + **}**
  + **console.log('Buy nothing!');**
  + ***//Output Buy nothing！***
  + So far we cannot yet translate the above episode with JavaScript, so if-else statement is required.
* **if-else** statement
  + syntax format of if-else statement is as below:
  + **if (condition){**
  + **// Statement block 1 executed when condition==true**
  + **} else {**
  + **// Statement block 2 executed when condition==false**
  + **}**
  + When the condition is satisfied, statement block 1 is executed; when the condition is not satisfied, statement block 2 is executed.
  + OK. Let’s translate the above episode.
  + For example:
  + **var seeTomato = 'yes';**
  + **if (seeTomato == 'yes'){**
  + **console.log('Buy two watermelons!');**
  + **} else {**
  + **console.log('Buy one watermelon!');**
  + **}**
  + ***// Output Buy two watermelons!***
  + **var seeTomato = 'no';**
  + **if (seeTomato == 'yes'){**
  + **console.log('Buy two watermelons!');**
  + **} else {**
  + **console.log('Buy one watermelon!');**
  + **}**
  + ***// Output Buy one watermelon!***
  + So the programmer husband is right？！
  + You might also ask: Only one condition can be set above, how can we set multiple conditions? The answer is to combine if-else statements.
* **else-if** statement
  + Syntax format of else-if statement is as below:
  + **if (condition1){**
  + **// Statement block 1 to be executed when condition1==true**
  + **} else if (condition2){**
  + **// Statement block 2 executed when condition2==true**
  + **} else {**
  + **// Statement block 3 executed when condition1==false && condition2==false**
  + **}**
  + You may feel confused. Now lets’ make a conversion.
  + **if (condition1){**
  + **// Statement block 1 executed when condition1==true**
  + **} else {**
  + **if (condition2){**
  + **// Statement block 2 executed when condition2==true**
  + **} else {**
  + **// Statement block 3 executed when condition1==false && condition2==false**
  + **}**
  + **}**
  + It is easier to understand it. The second if-else statement is embedded in the first if-else. However, we would use the first method for the purpose of convenience, beauty and good code writing habit.
  + For example:
  + **var date = 3;**
  + **if (date == 1){**
  + **console.log('Monday');**
  + **} else if (date == 2){**
  + **console.log('Tuesday');**
  + **} else {**
  + **console.log('unknown');**
  + **}**
  + ***// Output unknown***
  + Looking at these codes, you might think back what about Wednesday and Thursday? Embed them again!
  + For example:
  + **var date = 5;**
  + **if (date == 1){**
  + **console.log('Monday');**
  + **} else if (date == 2){**
  + **console.log('Tuesday');**
  + **} else if (date == 3){**
  + **console.log('Wednesday');**
  + **} else if (date == 4){**
  + **console.log('Thursday');**
  + **} else if (date == 5){**
  + **console.log('Friday');**
  + **} else if (date == 6){**
  + **console.log('Saturday');**
  + **} else if (date == 7){**
  + **console.log('Sunday');**
  + **} else {**
  + **console.log('unknown');**
  + **}**
  + ***// Output Friday***
  + You might ask: the codes are so long. Is there any simplification method. Then switch statement will help you.
* **swich** statement
  + Syntax format of switch statement is show as below:
  + **switch(n){**
  + **case n1:**
  + **Execute code block 1**
  + **break;**
  + **case n2:**
  + **Execute code block 2**
  + **break;**
  + **default:**
  + **Execute code block 3 if it is not equivalent to either case n1 or case n2.**
  + **}**
  + Here n is a variable. If n is equivalent to n1, code block 1 will be executed; if n is equivalent to n2, code block 2 will be executed; if it is equivalent to neither one, code block 3 will be executed. Then the codes can be written as below:
  + **var date = 5;**
  + **switch(date){**
  + **case 1:**
  + **console.log("Monday");**
  + **break;**
  + **case 2:**
  + **console.log("Tuesday");**
  + **break;**
  + **case 3:**
  + **console.log("Wednesday");**
  + **break;**
  + **case 4:**
  + **console.log("Thursday");**
  + **break;**
  + **case 5:**
  + **console.log("Friday");**
  + **break;**
  + **case 6:**
  + **console.log("Saturday");**
  + **break;**
  + **case 7:**
  + **console.log("Sunday");**
  + **break;**
  + **default:**
  + **console.log("unknown");**
  + **}**
  + ***//Output Friday***
  + Finally, what’s the role of break statement? **break** statement is used to make the **switch** statement block jump out. If it is not used, the satisfied code behind **case** will be executed continuously. If it is used, the **switch** statement block will jump out.
  + For example:
  + **var date = 5;**
  + **switch(date){**
  + **case 1:**
  + **console.log("Monday");**
  + **case 2:**
  + **console.log("Tuesday");**
  + **case 3:**
  + **console.log("Wednesday");**
  + **case 4:**
  + **console.log("Thursday");**
  + **case 5:**
  + **console.log("Friday");**
  + **case 6:**
  + **console.log("Saturday");**
  + **case 7:**
  + **console.log("Sunday");**
  + **default:**
  + **console.log("unknown");**
  + **}**
  + ***// Output:***
  + **Friday**
  + **Saturday**
  + **Sunday**
  + **unknown**
  + This is all about this section. Finally I would like to stress that a conditional branch statement can execute one case every time if there are multiple cases.

**Recommended Resources**

* JavaScript If...Else Statement(<http://www.runoob.com/js/js-if-else.html>)
* JavaScript switch Statement(<http://www.runoob.com/js/js-switch.html>)