01_04 UNDERSTANDING EVENT PROPAGATION

- Event propagation allows an element to capture all the events of its children elements.
- Consider the example in the previous chapter. Suppose we want an event to occur when we click at any of the images.
- In the older model (where there is no event propagation parameter) we would need to register an event for each of the list items. With event propagation we can add an event listener to ***ul*** and check for any events happening inside the unordered list.

```
<!doctype html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <title>JavaScript Events</title>
  <link rel="stylesheet" href="style.css">
</head>
<body>
  <ul id = "grid">
    <li><img src="images/ylw.gif" alt="yellow"></li>
    <li><img src="images/org.gif" alt="orange"></li>
    <li><img src="images/ppl.gif" alt="purple"></li>
    <li><img src="images/blu.gif" alt="blue"></li>
    <li><img src="images/pnk.gif" alt="pink"></li>
    <li><img src="images/grn.gif" alt="green"></li>
    <li><img src="images/ygr.gif" alt="ygreen"></li>
    <li><img src="images/gry.gif" alt="gray"></li>
    <li><img src="images/red.gif" alt="red"></li>
  </ul>

  <script>
    document.getElementById("grid").addEventListener("click",
    function(e)
    {
        //console.log(e);
        console.log(e.toElement.alt);
    },false);
  </script>
</body>
</html>
```

- **Capturing vs Bubbling Propagation models**
    - Capturing goes down the DOM while Bubbling goes up the DOM tree.
- capturing example :

```
<!doctype html>
```

```
<html lang="en">
<head>
  <meta charset="UTF-8">
  <title>JavaScript Events</title>
  <link rel="stylesheet" href="style.css">
</head>
<body>
  <ul id = "grid">
    <li><img src="images/ylw.gif" alt="yellow"></li>
    <li><img src="images/org.gif" alt="orange"></li>
    <li><img src="images/ppl.gif" alt="purple"></li>
    <li><img src="images/blu.gif" alt="blue"></li>
    <li><img src="images/pnk.gif" alt="pink" id = "pink"></li>
    <li><img src="images/grn.gif" alt="green"></li>
    <li><img src="images/ygr.gif" alt="ygreen"></li>
    <li><img src="images/gry.gif" alt="gray"></li>
    <li><img src="images/red.gif" alt="red"></li>
  </ul>

  <script>
    document.getElementById("grid").addEventListener("click",
    function(e)
    {
       console.log("clicked inside UL");
    },true);
    document.getElementById("pink").addEventListener("click",
    function(e)
    {
        console.log(e.toElement.alt);
    },true);
  </script>
</body>
</html>
```

output in console when clicked on pink : clicked inside UL
pink

- Bubbling example :

```
<!doctype html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <title>JavaScript Events</title>
  <link rel="stylesheet" href="style.css">
</head>
<body>
  <ul id = "grid">
    <li><img src="images/ylw.gif" alt="yellow"></li>
    <li><img src="images/org.gif" alt="orange"></li>
    <li><img src="images/ppl.gif" alt="purple"></li>
    <li><img src="images/blu.gif" alt="blue"></li>
    <li><img src="images/pnk.gif" alt="pink" id = "pink"></li>
    <li><img src="images/grn.gif" alt="green"></li>
    <li><img src="images/ygr.gif" alt="ygreen"></li>
    <li><img src="images/gry.gif" alt="gray"></li>
    <li><img src="images/red.gif" alt="red"></li>
  </ul>

  <script>
    document.getElementById("grid").addEventListener("click",
    function(e)
    {
       console.log("clicked inside UL");
    },false);
    document.getElementById("pink").addEventListener("click",
    function(e)
    {
        console.log(e.toElement.alt);
    },false);
  </script>
</body>
</html>
```

output in console when clicked on pink : pink clicked inside UL

- Note that some browsers like IE support only events that bubble while some support both propagation models.
- Remember that the event propaagation parameter in ***addEventListener*** method can have two possible values :
    1. **true** - The event handler is executed in the capturing phase.
    2. **false** - Default. The event handler is executed in the bubbling phase