#Function 모던웹을위한 Javascript jQuery입문\_제 5장,

일반함수의 형식

Function 함수명(매개변수1, 매개변수2){

}

참조변수를 이용한 함수사용

Var 참조변수이름 = function(){};

일반함수와 참조변수로 선언된 함수의 차이: 참조변수로 선언된 함수는 호이스팅에의해 선언이전에 사용하면 오류발생한다, 하지만 일반함수 형식으로 선언된 함수는 선언 이전에서도 사용가능하다

-일반적인 함수형식

<head>

    <meta charset="UTF-8">

    <meta http-equiv="X-UA-Compatible" content="IE=edge">

    <meta name="viewport" content="width=device-width, initial-scale=1.0">

    <title>Document</title>\

    <script>

        var color = ["white","yellow","aqua","purple"]

        var i =0;

        function changeColor(){

            window.setInterval(function(){

            i++

            if(i>=color.length){

                i=0;

            }

            var bodyTag = document.getElementById("the Body");

            bodyTag.style.backgroundColor=color[i];

             },2000);

        }

        changeColor();

    </script>

</head>

<body id = "the Body">

</body>

매개변수 없이 값 전달하기

        function sum(){

            var sum =0;

            for(var i = 0; i < arguments.length; i++) {

                sum+=arguments[i];

            }

            document.write(sum);

        }

        sum(10,30,50);

받을 매개변수를 저장할 arguments로받아 for문을 통해 처리해준다

        function sum(){

            var sum =0;

            for(var i = 0; i < arguments.length; i++) {

                if(i ==0){

                    document.write("인덱스0이 수행할일")

                }

                else if(i ==1){

                    document.write("인덱스1이 수행할일")

                }

                else if(i ==2){

                    document.write("인덱스2이 수행할일")

                }

                else if(i ==3){

                    document.write("인덱스3이 수행할일")

                }

                else if(i ==4){

                    document.write("인덱스4이 수행할일")

                }

                else if(i ==5){

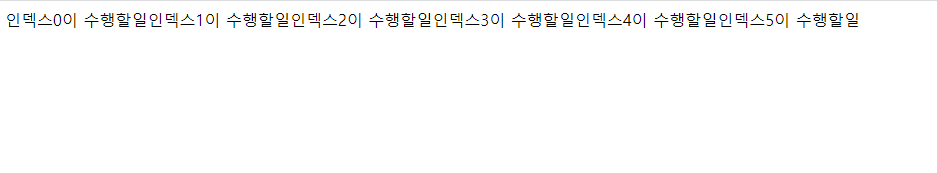
                    document.write("인덱스5이 수행할일")

                }

            }

        }

        sum(10,30,50,1,1,1,1,1);



위처럼 사용할수도있다

-재귀함수

var num = 0;

        window.setTimeout(function recrusion(){

            num++;

            document.write("재귀함수 실행중 : "+num );

            if(num==10){

                return;

            }

            window.setTimeout(recrusion,1000)

        },0)

이 구문은 재귀를 통해 호출하므로 스택 영역에 할당된 메모리가 해제되지 않는 치명적인 단점을 가지고 있다. if문을통해 특정 조건일 때 탈출하는 탈출조건을 부여

-this 식별자와 prototype을이용한 함수생성

function TestScore(name,kor,eng){

            this.username=name;

            this.userkor=kor;

            this.usereng=eng;

            this.getTestInfo = function(){

                var str= "";

                str += "이름 :" + this.username + "<br>";

                str += "국어성적 :"+this.userkor+"<br>";

                str += "영어성적 :"+this.usereng+"<br>";

                return str;

            }

            this.getavg = function(){

                return (this.userkor+this.usereng)/2

            }

        }

        // TestScore.prototype.getavg = function(){

        //     var kor1 = this.userkor;

        //     var eng1 = this.usereng;

        //     return (kor1+eng1)/2;

        // }

        var kimgun = new TestScore("김군",90,80);

        document.write(kimgun.getTestInfo());

        document.write(kimgun.avg());

이벤트 모던웹을위한 Javascript jQuery입문\_제 10장

1.윈도우가 로드되자마자 이벤트를 달아주기위해 window.onload 함수에 정의

2.body의 event를 달아주고싶은곳의 id를얻어와야한다

Document.getelementbyId(“해당id”)

3.얻어온 아이디에 onclick을 달아주고 이벤트발생시 실행시킬 함수를 정의한다

Document.getelementbyId(“id”).onclick = function(){

실행시킬 이벤트정의

}  
4. 이벤트 강제실행, 종료는 각각 참조변수.onclick(); , Document.getelementbyId(“id”).onclick = null; or 참조변수.onclick = null;

5 이벤트 버블 발생시

인터넷 익스플로러 : 이벤트 객체의 cancelBubble속성을 true로변경

Document.getElementById(“id”).onclick = function(e){

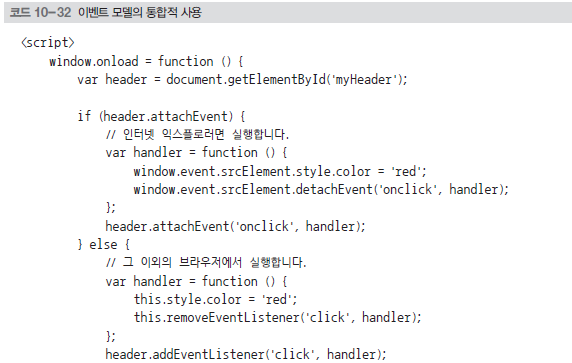
Var event = e::window.evnet

Event.cancelBubble = true;

그 이외의 브라우저 : 이벤트 객체의 stopPropagation()메서드를 사용

If(event.stopPropagation){

Event.stopPropagation();

}

