

```

package {

    import flash.display.MovieClip;
    import RandomNumberGenerator;
    import flash.events.Event;
    import flash.events.MouseEvent;
    import com.greensock.TweenLite;

    public class Puzzle extends MovieClip {

        private var randOn:Boolean;
        private var _tiles:Array;
        private var _puzzleStarted:Boolean;
        private var _puzzleComplete:uint;
        private var _currentTile:Object;
        private var _checkMark:Object;

        public var correct0:Correct = new Correct();
        public var correct1:Correct = new Correct();
        public var correct2:Correct = new Correct();
        public var correct3:Correct = new Correct();
        public var correct4:Correct = new Correct();
        public var correct5:Correct = new Correct();

        public var sinkSFX:String = "media/sink.mp3";
        public var playSFX:PlaySound;
        public var bkgrdSFX:String = "media/thunderstorm.mp3";
        public var playBack:PlaySound = new PlaySound(bkgrdSFX, 0.1,
true);

        public function Puzzle() {
            // constructor code

            _tiles = [tile0,tile1,tile2,tile3,tile4,tile5];
            toggle();
            toggle();
            for (var t:uint = 0; t < 6; t++){
                this["tile" +
t].addEventListener(MouseEvent.MOUSE_DOWN, dragTiles);
                this["tile" + t].addEventListener(MouseEvent.MOUSE_UP,
stopDragTiles);
                this["tile" + t].addEventListener(MouseEvent.ROLL_OUT,
stopDragTiles);
                this["tile" + t].alpha=.9;
            }
        }
    }
}

```

```

        }
        this.addEventListener(Event.ENTER_FRAME, hideUI);
    }
    include "HideUI.as"
//Card
Placement=====
        private function toggle():void{
            randOn = !randOn;
            if (randOn == true){
                var generator:RandomNumberGenerator = new
RandomNumberGenerator();
                var orig:Array = [];
                getOriginalPositions();
                function getOriginalPositions(event:Event =
null):void
                {
                    for(var i:int; i<_tiles.length; i++)
                    {
                        orig.push([_tiles[i].x,
_tiles[i].y]);
                    }
                    moveCards();
                }
                function moveCards(event:Event = null):void
                {
                    for(var i:int; i<_tiles.length; i++)
                    {
                        var randomIndex:int =
generator.getInteger(0, orig.length-1);
                        orig[randomIndex][0];
                        orig[randomIndex][1];
                        _tiles[i].x =
                        _tiles[i].y =
                        orig.splice(randomIndex,1);
                        trace(randomIndex);
                    }
                }
            }
            else {}
        }
        private function dragTiles(e:MouseEvent):void{
            _puzzleStarted = true;
            e.currentTarget.startDrag();
            var mc:MovieClip=MovieClip(e.currentTarget);
            setChildIndex(mc, numChildren -1);
        }
    }

```

```

private function lockTile(e:Event = null):void
{
    _puzzleComplete++;
    _currentTile.addChild(_checkMark);
    _checkMark.x = 0; _checkMark.y = 0;

    _currentTile.removeEventListener(MouseEvent.MOUSE_DOWN, dragTiles);
    _currentTile.removeEventListener(MouseEvent.MOUSE_UP,
stopDragTiles);
    _currentTile.removeEventListener(MouseEvent.ROLL_OUT,
stopDragTiles);
    TweenLite.to(_currentTile, .1,{delay:.1,alpha:1,
scaleX:1, scaleY:1, onComplete:checkStatus});
    var mc:MovieClip = _currentTile as MovieClip
    setChildIndex(mc, 0);
    //checkStatus();
}
private function stopDragTiles(e:MouseEvent):void{
    e.currentTarget.stopDrag();
    if (e.currentTarget.x < 100){e.currentTarget.x = 6;}
    else if (e.currentTarget.x < 500){e.currentTarget.x =
430;}

    else {e.currentTarget.x = 858}

    if (e.currentTarget.y < 100){e.currentTarget.y = 6;}
    else if (e.currentTarget.y > 300){e.currentTarget.y =
402;}

    else {}

    switch (e.currentTarget){
        case tile0:
            if(e.currentTarget.x == 6 &&
e.currentTarget.y == 6 ){
                _checkMark = correct0;
                trace(_checkMark)
                _currentTile =
e.currentTarget;
                TweenLite.to(_currentTile, .1,{delay:.1, alpha:1, scaleX:1.1, scaleY:1.1,
onComplete:lockTile});
            }
            break;
        case tile1:
            if(e.currentTarget.x == 430 &&
e.currentTarget.y == 6){
                _currentTile =

```



```

TweenLite.to(_currentTile, .1,{delay:.1, alpha:1, scaleX:1.1, scaleY:1.1,
onComplete:lockTile});
}
break;
}
//END SWITCH
}
private function checkStatus():void{
trace(_puzzleComplete)
if (_puzzleComplete == 6){
showUI();//Dispatch Event to say success,
remove children listeners and unload swf
var completeStatus:Complete = new Complete();
completeStatus.x = 625; completeStatus.y =
-200;
addChild(completeStatus);
TweenLite.to(completeStatus, .5,{y:400,
onComplete:removeChecks});
playSFX = new PlaySound(sinkSFX, 0.3)
}
}
public function removeChecks(e:Event = null):void{
for (var t:uint = 0; t < 6; t++){
//this["tile" +
t].removeEventListener(MouseEvent.CLICK, dragTiles);
//this["tile" +
t].removeEventListener(MouseEvent.CLICK, stopDragTiles);
this["tile"+
t].removeChild(this["correct" + t]);
}
}
//end custom functions
}
}
}

```