

ÖVNING 3

ÖVNING 1 AV 2 – MOUSE EVENT OCH IF-SATSER

I förra övningen skapade du ett nytt Movie Clip som du döpte till IfPage och bland annat lade ut lite texturor på. I den här övningen ska vi med hjälp av det du gjorde i förra övningen skapa ett spel där användaren ska gissa ett nummer, och spelet skriver ut om det var rätt eller ej.

1. Börja med att skapa en ny mapp. Kopiera över både InteraktivBok.fla och Main.as från förra övningen till den nya mappen. Öppna båda filerna (du behöver inte ta bort någon kod från förra veckan).
2. Först måste vi deklarera IfPage precis som vi gjorde med HillPage. Lägg därför till följande kod:

```
public class Main extends MovieClip {  
  
    var hillPage:HillPage;  
    var ifPage:IfPage;  
  
    //-----  
    //Constructor  
    public function Main() {  
        init();  
    }  
  
    //-----  
    //Initializes game  
    private function init():void {  
        ifPage = new IfPage(); //New IfPage instance  
        hillPage = new HillPage(); //New hillPage instance  
        addChild(hillPage); //Adds hillPage to stage  
    }  
}
```

3. Vi ska nu lägga till event listeners för knapparna HillButton och IfButton som du skapade förra gången. Meningen är att du med hjälp av dessa knappar ska kunna hoppa mellan HillPage och IfPage. Vad vi måste göra är alltså följande:
 - a. Ersätta addChild(hillPage) som just nu står i init() med addChild(ifPage) (vi vill ju inte att det är HillPage som visas, utan nu ska vi jobba med IfPage).
 - b. Lägga till en event listener som lyssnar efter om användaren klickar på hillButton och i så fall visar HillPage.
 - c. Lägga till en event listener som lyssnar eftersom om användaren klickar på ifButton och i så fall visar IfPage.

En god idé kan vara att använda removeChild() för att ta bort den sida som visas på lämpligt ställe i koden (bara ett tips).

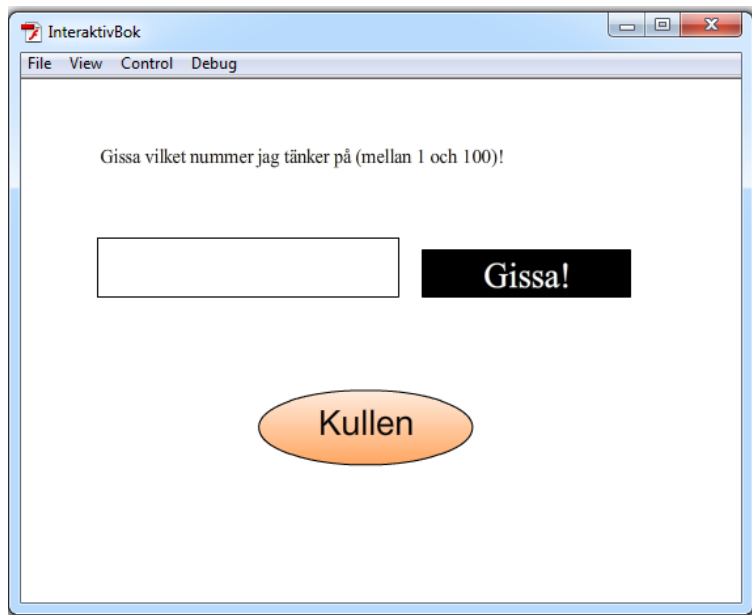
4. Genomför ändringarna som krävs enligt steg 3. Om du inte kommer ihåg hur man lade in event listeners kan du kika på förra veckans övning, eller tjuvkika på min lösning på nästa sida.

```
1 package {
2     import flash.display.MovieClip;
3     import flash.events.MouseEvent;
4
5     public class Main extends MovieClip {
6
7         var hillPage:HillPage;
8         var ifPage:IfPage;
9
10        //-----
11        //Constructor
12        public function Main() {
13            init();
14        }
15
16        //-----
17        //Initializes game
18        private function init():void {
19            ifPage = new IfPage();
20            hillPage = new HillPage(); //New hillPage instance
21            addChild(ifPage); //Adds hillPage to stage
22
23            //Adds event listeners to buttons
24            hillPage.downButton.addEventListener(MouseEvent.CLICK, onDownButtonClick);
25            hillPage.growButton.addEventListener(MouseEvent.CLICK, onGrowButtonClick);
26            hillPage.rotateButton.addEventListener(MouseEvent.CLICK, onRotateButtonClick);
27            hillPage.ifButton.addEventListener(MouseEvent.CLICK, onIfButtonClick);
28            ifPage.hillButton.addEventListener(MouseEvent.CLICK, onHillButtonClick);
29        }
30
31        //-----
32        //Handles click on ifButton
33        private function onIfButtonClick(e:MouseEvent):void {
34            addChild(ifPage);
35            removeChild(hillPage);
36        }
37
38        //-----
39        //Handles click on hillButton
40        private function onHillButtonClick(e:MouseEvent):void {
41            addChild(hillPage);
42            removeChild(ifPage);
43        }
44
```

5. Vi ska nu skriva en metod som heter `initGame()`. Den metoden ska välja vilket nummer användaren ska gissa (d.v.s. vad rätt svar är), och också skriva ut instruktioner till användaren i ett av textfälten. `initGame()` ska köras från `init()`. Du får färdig kod till `initGame()` den här gången:

```
1 package {
2     import flash.display.MovieClip;
3     import flash.events.MouseEvent;
4
5     public class Main extends MovieClip {
6
7         //Declares variables
8         var hillPage:HillPage;
9         var ifPage:IfPage;
10        var startMessage:String;
11        var mysteryNumber:uint;
12
13        /*Uint (unsigned integer) är endast positiv
14        och är den snabbaste (minsta) formen av integer.
15        Används mest för bäst prestanda.
16        Default värde för uint är 0.*/
17
18
19        //-----
20        //Initializes application
21        private function init():void {
22            ifPage = new IfPage(); //Adds new IfPage instance
23            hillPage = new HillPage(); //Adds new HillPage instance
24            addChild(ifPage); //Adds ifPage to stage
25
26            //Adds event listeners to buttons
27            hillPage.downButton.addEventListener(MouseEvent.CLICK, onDownButtonClick);
28            hillPage.growButton.addEventListener(MouseEvent.CLICK, onGrowButtonClick);
29            hillPage.rotateButton.addEventListener(MouseEvent.CLICK, onRotateButtonClick);
30            hillPage.ifButton.addEventListener(MouseEvent.CLICK, onIfButtonClick);
31            ifPage.hillButton.addEventListener(MouseEvent.CLICK, onHillButtonClick);
32
33            initGame(); //Initializes game
34        }
35
36
37        //-----
38        //Initializes game
39        private function initGame():void {
40            //Output message telling user about game
41            startMessage = "Gissa vilket nummer jag tänker på (mellan 1 och 100)!";
42
43            mysteryNumber = 50; //The number the user has to guess
44
45            ifPage.output.text = startMessage; //Sets text for output textbox
46            ifPage.input.text = ""; //Empties input textbox if necessary
47        }
48
49
50
51
52
53
```

När jag testar min applikation (Ctrl + Enter) ser det ut så här:



Dock händer det ju ingenting just nu. Det behövs lite mer kod för det.

6. Just nu går det att skriva in både siffror och bokstäver i input-rutan. För att ändra på det lägg till följande i `initGame()`:

```
ifPage.input.restrict = "0-9";
```

7. Nu kan användaren bara skriva in siffror. Dock händer det ju fortfarande ingenting. Vi måste lägga in en event listener på `guessButton` och i event handler-metoden måste programmet läsa av vad användaren har gissat och jämföra gissningen med `mysteryNumber`. Jämförelsen görs enklast med en `if/else`-sats. Testa att skriva koden själv, och om du fastnar finns min lösning på nästa sida.

```
1 package {
2     import flash.display.MovieClip;
3     import flash.events.MouseEvent;
4
5     public class Main extends MovieClip {
6
7         //Declares variables
8         var hillPage:HillPage;
9         var ifPage:IfPage;
10        var startMessage:String;
11        var mysteryNumber:uint;
12        var currentGuess:uint;
13
14
15
16
17
18
19
20
21
22
23
24
25
26
27
28
29
30
31
32
33
34
35
36
37
38
39
40
41
42        //-----
43        //Initializes game
44        private function initGame():void {
45            //Output message telling user about game
46            startMessage = "Gissa vilket nummer jag tänker på (mellan 1 och 100)!";
47
48            mysteryNumber = 50; //The number the user has to guess
49
50            ifPage.output.text = startMessage; //Sets text for output textbox
51            ifPage.input.text = ""; //Empties input textbox if necessary
52            ifPage.input.restrict = "0-9";
53
54            ifPage.guessButton.addEventListener(MouseEvent.CLICK, onGuessButtonClick);
55        }
56
57        //-----
58        //Handles click on guessButton
59        private function onGuessButtonClick(e:MouseEvent):void {
60            currentGuess = uint(ifPage.input.text); //Gets current guess from textbox (converts to uint)
61
62            //Compares current guess with correct number
63            if(currentGuess > mysteryNumber) {
64                ifPage.output.text = "Du gissade för högt!";
65            } else if(currentGuess < mysteryNumber) {
66                ifPage.output.text = "Du gissade för lågt!";
67            } else {
68                ifPage.output.text = "Du gissade rätt!";
69            }
70        }
71    }
```

8. Det jag gjort i min if-sats är att först jämföra gissningen mot svaret och se om det är för högt. Är det för högt skrivs en text ut, och inget mer händer. Är det däremot inte för högt fortsätter programmet med att jämföra gissningen mot svaret och ser om det är för lågt. Är det inte för lågt heller drar vi slutsatsen att det är rätt gissat.
9. För närvarande är ju inte spelet speciellt kul – om du har gissat rätt en gång kan du ju svaret. Därför ska vi nu slumpa fram ett tal till mysteryNumber istället. För att göra det används metoden random() i klassen Math. Metoden returnerar ett decimaltal mellan 0 och 1, vilket gör att vi måste lägga till vissa uträkningar för att få en siffra mellan 1 och 100. Vi kommer dessutom att använda metoden ceil() i Math-klassen för att avrunda talet uppåt.

Det här med slumpmässiga tal är komplicerat i programmering eftersom en dator är logiskt byggd och inte bara kan komma på vad som helst. Du kommer att märka att slumpmässiga tal inte är riktigt helt slumpmässiga, utan de räknas ut med väldigt komplicerade algoritmer. Det fungerar dock bra i alla fall, speciellt för ett så pass enkelt spel som vi gör här.

10. Ersätt raden:

```
mysteryNumber = 50;
```

i `initGame()` med följande:

```
mysteryNumber = Math.ceil(Math.random() * 100);
```

11. Testa ditt spel. Om du tycker det är jobbigt att gissa kan du lägga in en `trace()` som skriver ut vilket tal det är som är rätt svar. Det vi gör är:

- a. Vi använder `Math.random()` och får ett slumpmässigt tal mellan 0 och 1.
- b. Vi multiplicerar talet med 100 för att vi vill ha ett tal mellan 1 och 100.
- c. Vi avrundar talet till heltal med `Math.ceil()` för vi vill ju inte ha några decimaler.

12. Vi ska även lägga in en funktion som gör att användaren kan gissa genom att trycka på enter-knappen, istället för att behöva trycka på `guessButton` hela tiden. Det vi behöver göra är:

- a. Importera `flash.events.KeyboardEvent` och `flash.ui.Keyboard`.
- b. Lägg till en event listener på scenen för att känna av om användaren trycker på en tangent (d.v.s. lyssnar efter `KeyboardEvent.KEY_DOWN`).
- c. Lyft ut det som händer i `onGuessButtonClick` till en ny metod som du döper till `gameEvents()`.
- d. Lägg till en event handler (`onKeyPress`) som hanterar gissningen, precis som när användaren tryckte på `guessButton`. Dock måste du först kolla så att användaren använt enter-knappen och inte någon annan av knapparna. Detta kan du göra genom att i en if-sats kolla om `e.keyCode` (`e` är den parameter som krävs för din metod) är detsamma som `Keyboard.ENTER`.

13. Försök göra detta själv. Behöver du hjälp finns min kod på nästa sida.

```

1 package {
2     import flash.display.MovieClip;
3     import flash.events.MouseEvent;
4     import flash.events.KeyboardEvent;
5     import flash.ui.Keyboard;
6
7     public class Main extends MovieClip {
8
9         //Declares variables
10        var hillPage:HillPage;
11        var ifPage:IfPage;
12        var startMessage:String;
13        var mysteryNumber:uint;
14        var currentGuess:uint;
15
16        //-----
17        //Initializes game
18        private function initGame():void {
19            //Output message telling user about game
20            startMessage = "Gissa vilket nummer jag tänker på (mellan 1 och 100)!";
21
22            mysteryNumber = Math.ceil(Math.random() * 100); //The number the user has to guess
23            trace(mysteryNumber);
24
25            ifPage.output.text = startMessage; //Sets text for output textbox
26            ifPage.input.text = ""; //Empties input textbox if necessary
27            ifPage.input.restrict = "0-9";
28
29            ifPage.guessButton.addEventListener(MouseEvent.CLICK, onGuessButtonClick);
30            addEventListener(KeyboardEvent.KEY_DOWN, onKeyPress);
31        }
32
33        //-----
34        //Handles key press for submitting guess
35        private function onKeyPress(e:KeyboardEvent):void {
36            if(e.keyCode == Keyboard.ENTER) {
37                gameEvents();
38            }
39        }
40
41        //-----
42        //Handles click on guessButton
43        private function onGuessButtonClick(e:MouseEvent):void {
44            gameEvents();
45        }
46
47        //-----
48        //Works as game engine
49        private function gameEvents():void {
50            currentGuess = uint(ifPage.input.text); //Gets current guess from textbox (converts to uint)
51
52            //Compares current guess with correct number
53            if(currentGuess > mysteryNumber) {
54                ifPage.output.text = "Du gissade för högt!";
55            } else if(currentGuess < mysteryNumber) {
56                ifPage.output.text = "Du gissade för lågt!";
57            } else {
58                ifPage.output.text = "Du gissade rätt!";
59            }
60        }
61    }
62}

```

14. Testa din applikation. Nu ska det gå att gissa även genom att trycka på enter-knappen. Du är nu egentligen färdig med ditt första spel, men det finns utrymme för förbättringar. På egen hand (jag kommer inte att lägga ut en lösning, du får försöka helt själv) ska du nu lägga till en funktion som begränsar antalet gissningar till 10 st. För det behöver du göra följande:

- a. Ha koll på hur många gånger användaren har gissat. Har användaren gissat 10 gånger ska spelet avslutas. Du kommer att behöva några nya variabler, t.ex.:

```
var guessesMade:uint;  
var guessesRemaining:int;
```

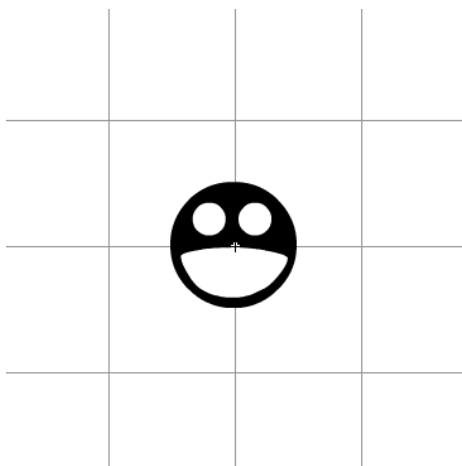
Du behöver med andra ord lägga in någon slags räknings- och jämförelsefunktion, förslagsvis i metoden gameEvents().

- b. Du behöver lägga till en knapp som startar om spelet när det är slut (d.v.s. nollställer antalet gissningar och genererar ett nytt nummer).
- c. Du måste ge feedback till användaren hur många gissningar de gjort och hur många de har kvar.

ÖVNING 2 AV 2 – FÖRBEREDELSE INFÖR NÄSTA ÖVNING

Under nästa övning ska du träna på att styra en figur med piltangenterna. För att göra det behöver du förbereda några saker.

1. Skapa en ny .FLA-fil (döp den till characterControl fla) och en .AS-fil (döp den till Main_Character.as). Ange Main_Character som Class i Properties i .FLA-filen.
2. För att underlätta arbetet, gå in på View → Grid → Edit Grid. Kryssa för "Show grid" och ställ storleken till 50x50 pixlar.
3. Skapa ett nytt Movie Clip som du döper till Player (glöm inte "Export for ActionScript" – detta är sista gången jag påminner om det!). Rita en figur som precis fyller en ruta.
4. För att centrera figuren (d.v.s. få den precis mitt på krysset), markera den och i Properties ange x: -25 och y: -25 (anledningen till att det är just -25 är att det är hälften av figuren som är 50 px eftersom vi ställde in griden på 50 px).
5. Dra ut figuren på scenen och ge den instansnamnet player. Så här ser min figur ut:



6. Leta upp en bild som är mycket större än scenen, minst 1000 x 1000 pixlar. Importera den till biblioteket.

7. Skapa ett Movie Clip som du döper till Background. Dra ut bilden du precis importerade i Movie Clipet. Detta Movie Clip kommer vi att lägga ut på scenen nästa gång, så just nu behöver du inte göra något mer med det.