Planning the development of animated menu

At this step we are going to build an animated menu using jQuery's `animate` method.

The animation will include the following stages:

- Stage 1 is the closed state, where the menu is not visible.
- Stage 2 is the transitional phase where the two `div`s spring out from behind the button to reach a preset distance on the vertical axis. Note that at this time, their opacities are also animated from zero to one.
- Stage 3 is when these `div`s expand horizontally.
- At stage 4 their heights are animated to reveal two individual panels. Once these panels are revealed, the unordered list that represents the menu is faded in.

When the menu button is pressed again, this entire process should repeat in the reverse.

Building the animated menu

Open the *animatedMenu_Begin.htm* file.

Start by creating two functions, namely `openMenu` and `closeMenu`. They will be used to trigger the animation sequences for the open and closed states of the panels:

```
function openMenu() {}

function closeMenu() {}
```

Create the click event handler for the `toggleMenu` hyperlink and set it to trigger the `openMenu` function:

```
$('#toggleMenu').on('click', function() { openMenu() });
```

Code the `openMenu` function:

```
var menuStatus = false;

function openMenu() {
// Open Menu A (Top Menu)
  $('.menuA')
    .animate({
      top:'0px',
      opacity:1
    },400,"swing")
    .animate({
      width:'140px',
      left:'120px',
      'border-radius':'5px'
    },200)
    .animate({
      height:'150px'
    },400, function() {
// Set Menu Open Status to True
```

```
        menuStatus = true;
// Fade In the Menu
        $('.submenu').fadeIn(100);
    });
// Open Menu B (Bottom Menu)
  $('.menuB')
      .animate({
        top:'370px',
        opacity:1
      },400)
      .animate({
        width:'140px',
        left:'120px',
        'border-radius':'5px'
      },200)
      .animate({
        height:'150px',
        top:'230px'
      },400);
}
```

The `menuStatus` variable is used to set and get the current state of the menu panels so they can be opened or closed as needed. When the menu is opened, this variable will be set to `true`, to denote that the menu is open. So now we can utilize this variable to call the required function:

```
$('#toggleMenu').on('click', function() {
    if(menuStatus === false){
        openMenu();
    } else {
        closeMenu();
    }
});
```

Now let's work on the `closeMenu` function:

```
function closeMenu() {
// FadeOut the Menu Text
  $('.submenu').fadeOut(100);
// Close Menu A (Top Menu)
  $('.menuA')
      .animate({
        height:'10px'
      },400)
      .animate({
        width:'10px',
        left:'185px',
        'border-radius':'50px'
      },200)
      .animate({
        top:'185px',
        opacity:1
      },400, function() {
// Set the Menu Open Status to False
        menuStatus = false;
    });
// Close Menu B (Bottom Menu)
  $('.menuB')
      .animate({
        height:'10px',
        top:'370px'
      },400)
      .animate({
        width:'10px',
        left:'185px',
        'border-radius':'50px'
      },200)
      .animate({
```

```
        top:'185px',
        opacity:0
      },400);
  }
```

Rotating the button on menu open/close:

Let's also rotate the button while the menu opens or closes. To do this, write an `animate` method in the `openMenu` function first:

```
function openMenu() {
// Open Menu A (Top Menu)
  $('.menuA')
      .animate({
        top:'0px',
        opacity:1
      },400,"swing")
      .animate({
        width:'140px',
        left:'120px',
        'border-radius':'5px'
      },200)
      .animate({
        height:'150px'
      },400, function() {
// Set Menu Open Status to True
        menuStatus = true;
// Fade In the Menu
        $('.submenu').fadeIn(100);
      });
// Open Menu B (Bottom Menu)
  $('.menuB')
      .animate({
        top:'370px',
        opacity:1
      },400)
      .animate({
        width:'140px',
        left:'120px',
        'border-radius':'5px'
      },200)
      .animate({
        height:'150px',
        top:'230px'
      },400);
// Rotate the Menu Button Clockwise
  $('#toggleMenu').animate({
    textIndent:1
  },{
    duration:1000,
    easing:'linear',
    step:function(now) {
      $(this).css('transform', 'rotateZ(' + (now * 360) + 'deg)');
    }
  });
}
```

Once again we animate the `text-indent` and use the `step` function.

Rotate the button counter-clockwise when we close the menu panels:

```
function closeMenu() {
// FadeOut the Menu Text
  $('.submenu').fadeOut(100);
// Close Menu A (Top Menu)
  $('.menuA')
```

```
      .animate({
        height:'10px'
      },400)
      .animate({
        width:'10px',
        left:'185px',
        'border-radius':'50px'
      },200)
      .animate({
        top:'185px',
        opacity:1
      },400, function() {
// Set the Menu Open Status to False
        menuStatus = false;
      });
// Close Menu B (Bottom Menu)
  $('.menuB')
      .animate({
        height:'10px',
        top:'370px'
      },400)
      .animate({
        width:'10px',
        left:'185px',
        'border-radius':'50px'
      },200)
      .animate({
        top:'185px',
        opacity:0
      },400);
  $('#toggleMenu').animate({
    textIndent:0
  },{
    duration:1000,
    easing:'linear',
    step:function(now) {
      $(this).css('transform', 'rotateZ(' + (now * 360) + 'deg)');
    }
  });
}
```

Now our menu animation should look great!