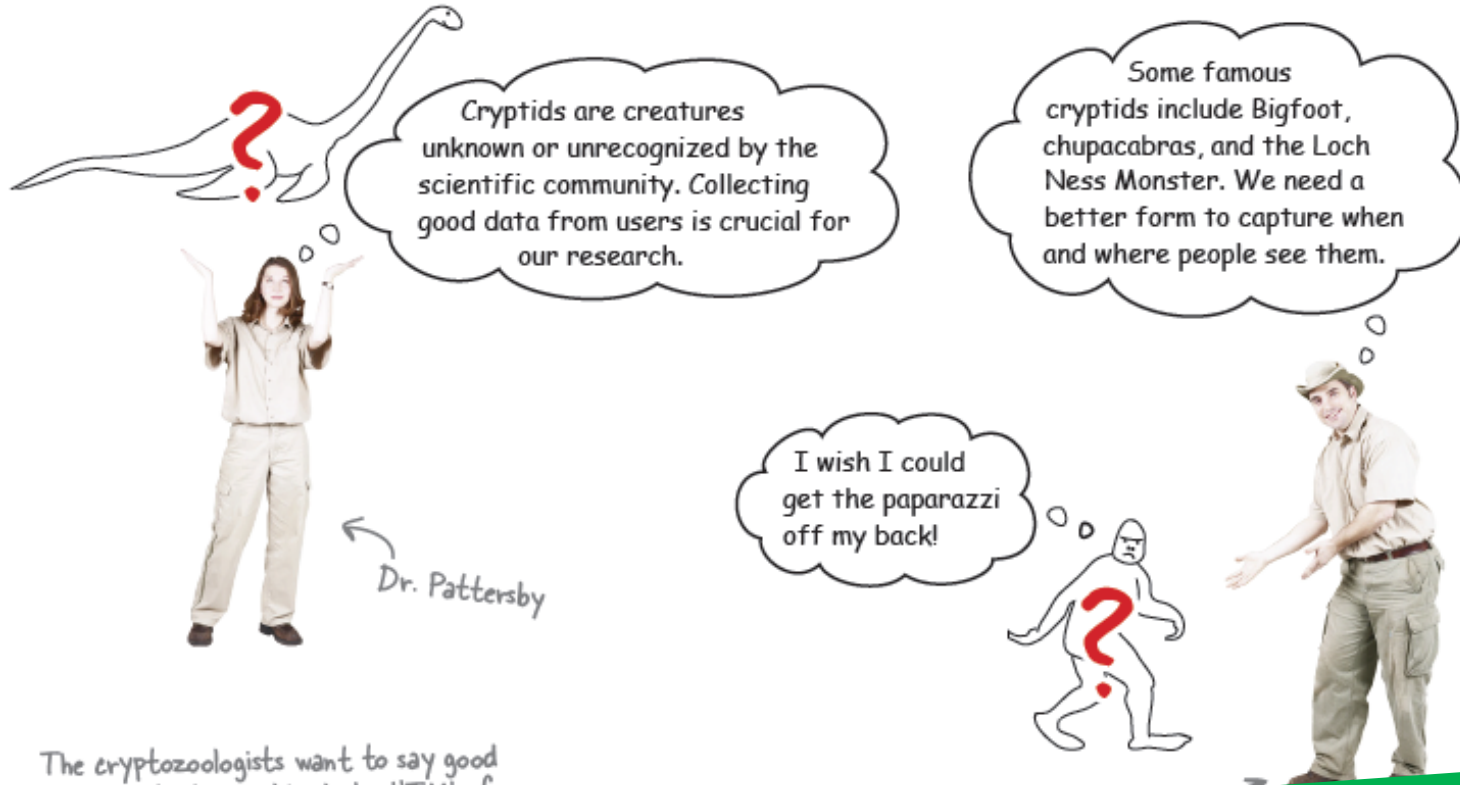


jQuery UI: Extreme form makeover

Samkeun Kim <skim@hknu.ac.kr>

<http://cyber.hknu.ac.kr/>



The cryptozoologists want to say good riddance to their old, clunky HTML form.

Cryptozoologists.org needs a makeover

Submit Cryptid Sighting

Please provide the following information about your cryptid sighting below along with your contact information.

Fill out the form below and click "Enter" to Submit

Date of Sighting:

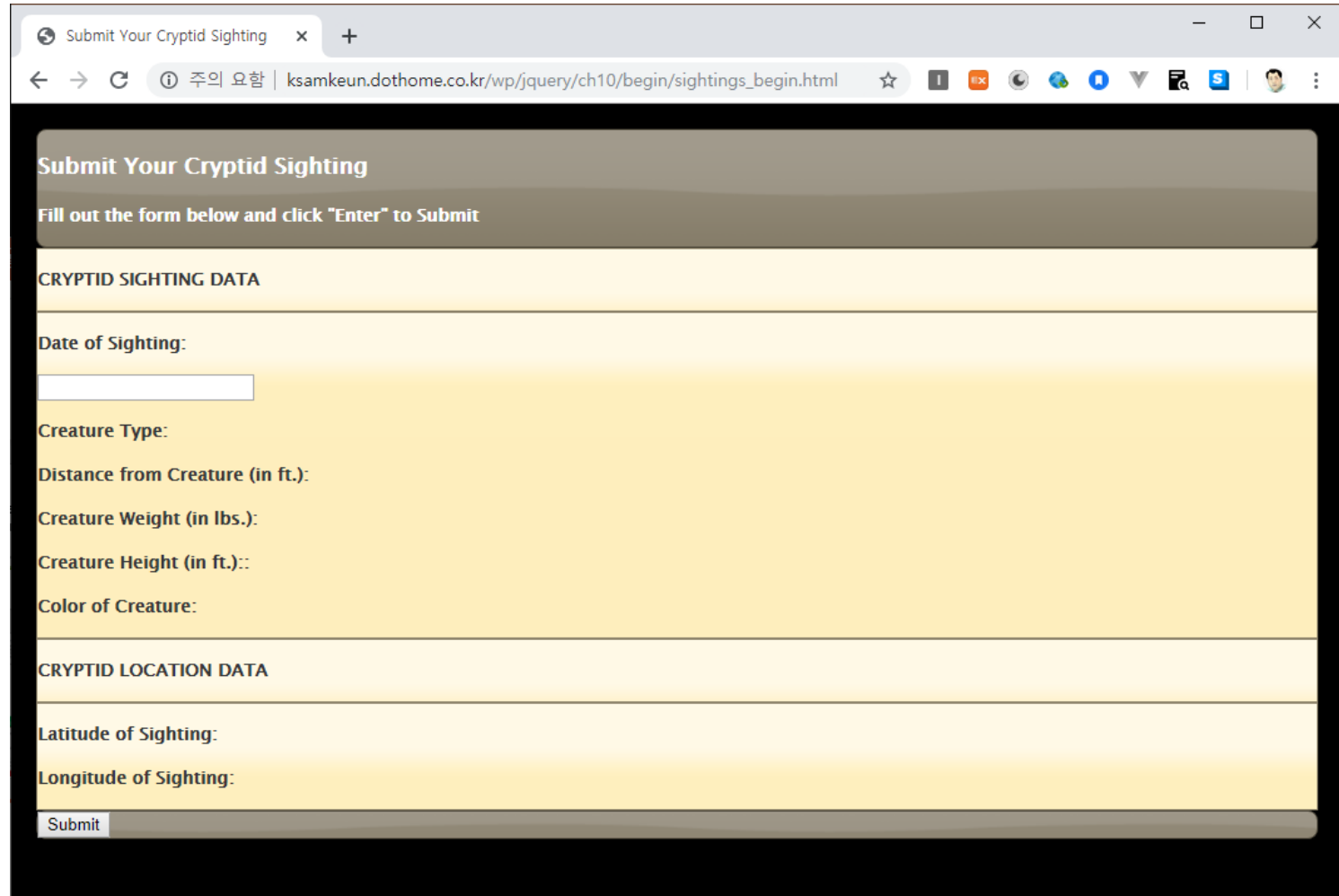
Where You Saw the Creature:

What Type: ☐ Mammal ☐ Reptile ☐ Bird

The look and feel of the form is not engaging. It looks like it's from the late 90s.

There's nothing visual that assists the user in entering quality data, and HTML doesn't offer a lot of options for that. As a result, the doctors get a lot of junk data.

현재 사이트



The screenshot shows a web browser window with the title 'Submit Your Cryptid Sighting'. The address bar displays the URL 'ksamkeun.dothome.co.kr/wp/jquery/ch10/begin/sightings_begin.html'. The page content includes a header with the title and a instruction: 'Fill out the form below and click "Enter" to Submit'. The form is divided into two main sections: 'CRYPTID SIGHTING DATA' and 'CRYPTID LOCATION DATA'. The first section contains input fields for 'Date of Sighting:', 'Creature Type:', 'Distance from Creature (in ft.):', 'Creature Weight (in lbs.):', 'Creature Height (in ft.):', and 'Color of Creature:'. The second section contains input fields for 'Latitude of Sighting:' and 'Longitude of Sighting:'. A 'Submit' button is located at the bottom left of the form.

Submit Your Cryptid Sighting

Fill out the form below and click "Enter" to Submit

CRYPTID SIGHTING DATA

Date of Sighting:

Creature Type:

Distance from Creature (in ft.):

Creature Weight (in lbs.):

Creature Height (in ft.):

Color of Creature:

CRYPTID LOCATION DATA

Latitude of Sighting:

Longitude of Sighting:

Submit

http://ksamkeun.dothome.co.kr/wp/jquery/ch10/begin/sightings_begin.html

Update your HTML form

Submit Your Cryptid Sighting

CRYPTID SIGHTING DATA

Date of Sighting:

Month						

In the current form, this is a text field, which gets us a lot of sloppy data. We want the user to pick from a pop-up calendar so that the data is as accurate as it can be.

Creature Type:

We use radio buttons for this one on the current form; can we make them look more inviting?

Distance from Creature (estimate in feet):

Creature Weight:

Creature Height:

Color of Creature Hide/Fur/Covering:

R

G

B

How about a color mixer here?

We'd like our users to click and drag to enter the numerical data fields. More accurate for us. Easier for our users.

SIGHTING LOCATION DATA

Latitude of Sighting Location:

Longitude of Sighting Location:

Enter your sighting data:

Can you make the button look nice? The pill-shaped button on our current form makes it look like we're way out of date.

Wouldn't it be dreamy if
there were some sort of library of
jQuery interface plug-ins?
But I know it's just a fantasy...



Save coding headaches with jQuery UI

Fortunately for developers everywhere, jQuery has an official library of **user interface plug-ins** for just this kind of project.

Effects plug-ins

jQuery UI extends jQuery by adding more effects. Make your elements bounce, explode, pulsate, or shake. jQuery UIs also includes *easing functions*, complex mathematical operations that make animations look more realistic.

Interaction plug-ins

Interactions add more complex behavior to web apps. You can enable users to interact with elements by making those elements draggable, droppable, or sortable, just to name a few of the options.

Widget plug-ins

A web widget is a self-contained component that adds functionality to your web app. Widgets save you tons of coding time and complexity while creating usable and responsive user interface elements.

We're going to focus mostly on widgets for our UI work in this chapter.

jQuery offers a plug-in architecture that allows web developers to extend (or add onto) the core jQuery library.

Exercise

Try out some **jQuery UI effects**, **interactions**, and **widgets** by visiting the following URLs and doing what the instructions say.

<http://jqueryui.com/animate/>

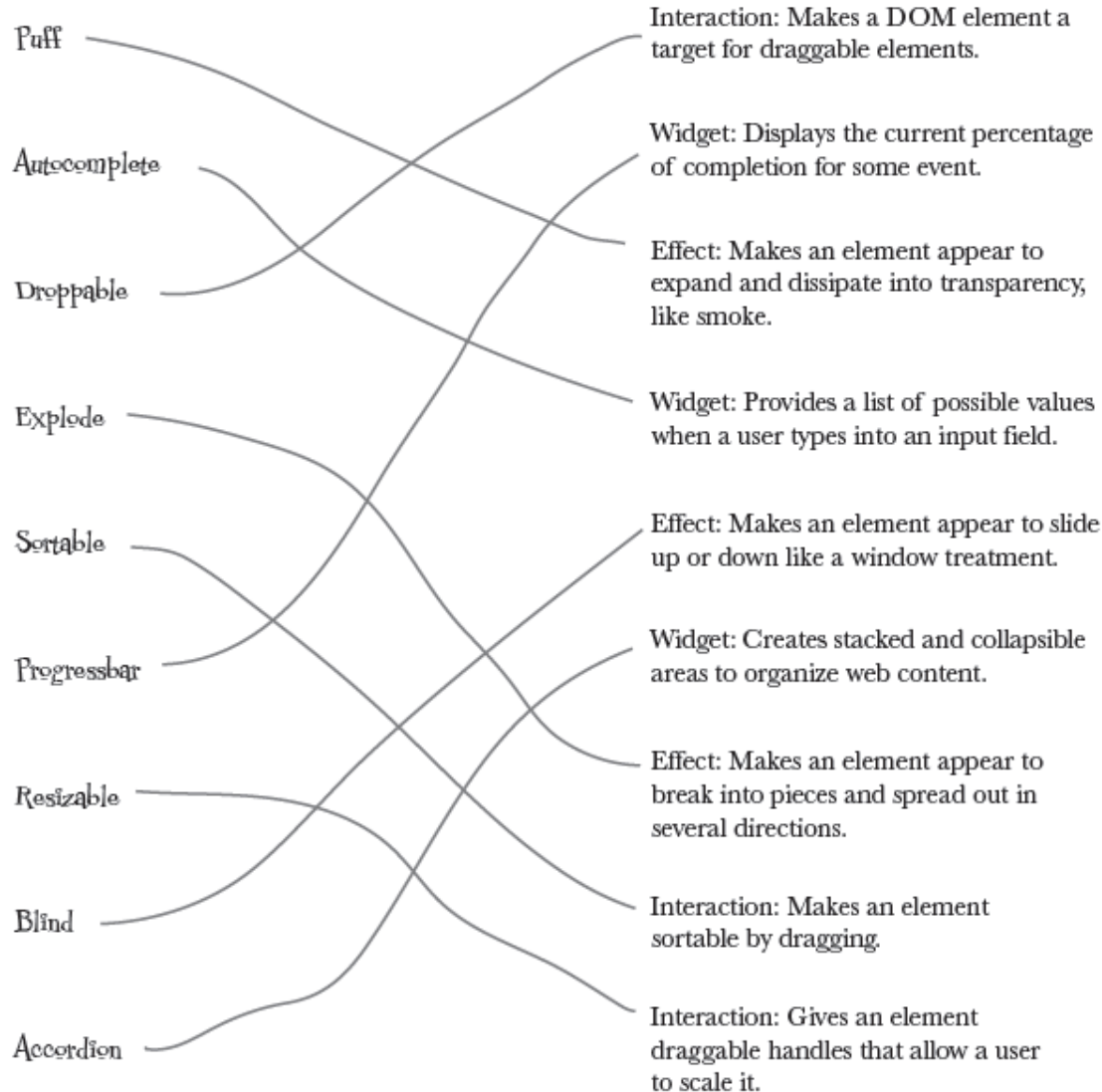
<http://jqueryui.com/effect/>

<http://jqueryui.com/draggable/>

<http://jqueryui.com/dialog/>

WHO DOES WHAT? SOLUTION

Match each jQuery UI plug-in to what type of plug-in it is and what it does.



<https://jqueryui.com/download/>

CDN 사용:

```
<script src="https://code.jquery.com/jquery-1.12.4.min.js"> </script>  
<link rel="stylesheet" href="//code.jquery.com/ui/1.12.1/themes/sunny/jquery-ui.css">  
<script src="http://ajax.googleapis.com/ajax/libs/jqueryui/1.10.3/jquery-ui.js"> </script>
```

Our project checklist

1. Build a datepicker for users to enter the date of the sighting.
2. Build more engaging radio buttons for users to choose the creature type.
3. Build number-entry sliders for users to enter distance from creature, creature weight, creature height, latitude, and longitude.
4. Build a color mixer interface component for the user to enter creature color.
5. Build a nicer-looking submit button for the sightings form.

1. 사용자가 신비생물을 본 날짜를 입력하는 **날짜 선택기**를 만든다.
2. 사용자가 본 신비생물이 어떤 타입인지 선택하는 좀 더 매력적인 **라디오 버튼**을 만든다.
3. 신비생물을 얼마나 멀리서 봤는지, 신비생물의 키와 몸무게는 얼마나 되는지, 신비생물이 있던 장소의 위도/경도 값은 얼마인지 입력할 숫자 필드를 **슬라이더**로 바꾼다.
4. 신비생물이 무슨 색이었는지 입력할 수 있도록 **색상 믹서**를 만든다.
5. **제출 버튼**을 더 멋져 보이게 만든다.

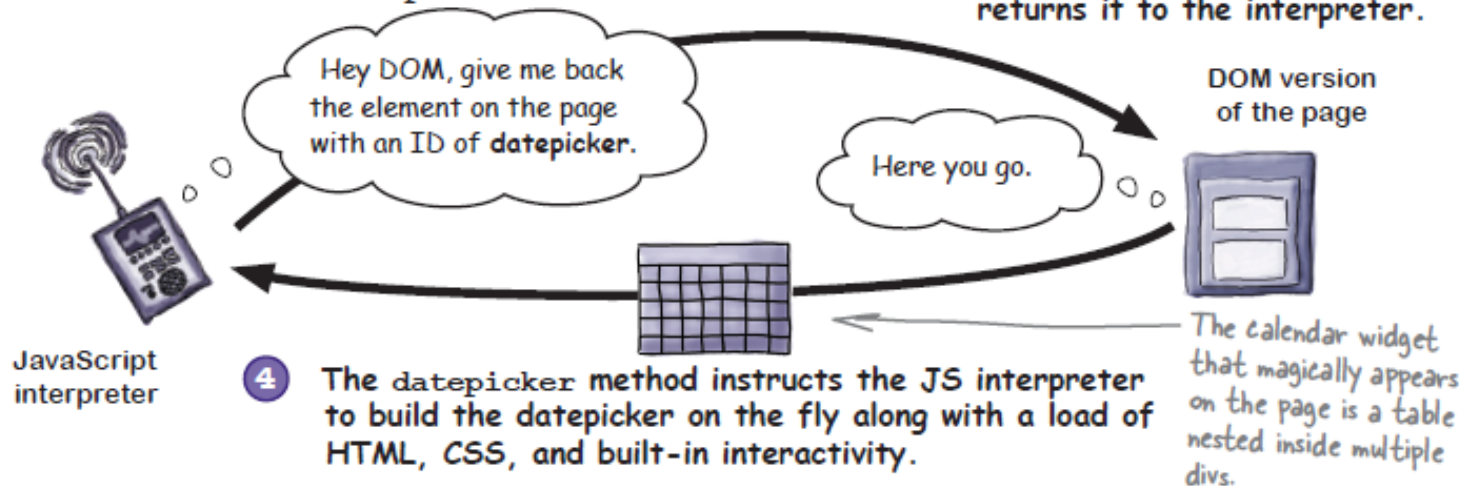
jQuery UI behind the scenes

- 1 Just like all the other jQuery code you've written, the datepicker uses a selector and a method.



- 2 Using the jQuery and jQuery UI library, the JavaScript interpreter asks the DOM for the element whose ID is `datepicker`.

- 3 The DOM grabs the selected element, runs the datepicker method on that element, and returns it to the interpreter.



- 4 The datepicker method instructs the JS interpreter to build the datepicker on the fly along with a load of HTML, CSS, and built-in interactivity.

Widgets have customizable options



Customize the datepicker with options

The datepicker widget comes with tons of configurable options. The `stepMonths` option lets you change how many months you want to hop.

```
$("#datepicker").datepicker({  
    stepMonths: 3  
});
```

If you start on August, when you press the "previous" or "next" button, you'll hop three months backward or three months forward.

```
$("#datepicker").datepicker({  
    changeMonth: true  
});
```

If you set the `changeMonth` option to `true`, the user can choose the month from a drop-down list.



Exercise:

사용자가 년과 월을 모두 드롭다운 목록을 통해 선택할 수 있게 하는 코드를 만들어 보자. 옵션을 여러 개 쓸 때는 옵션을 콤마로 구분한다.

```
$('#datepicker').datepicker({  
    changeMonth: true, changeYear: true  
});
```

sightings.html, my_scripts.js

<http://ksamkeun.dothome.co.kr/wp/jquery/ch10/end/style/form.css>

```
1  <!DOCTYPE html>
2  <html>
3  <head>
4      <title>Submit Your Cryptid Sighting</title>
5      <script src="https://code.jquery.com/jquery-1.12.4.min.js"></script>
6      <link rel="stylesheet" href="//code.jquery.com/ui/1.12.1/themes/sunny/jquery-ui.css">
7      <script src="http://ajax.googleapis.com/ajax/libs/jqueryui/1.10.3/jquery-ui.js"></script>
8      <link rel="stylesheet" type="text/css" href="style/form.css" />
9      <script src="scripts/my_scripts.js"></script>
10 </head>
11
12 <body>
13     <h3>Date of Sighting:</h3>
14     <form>
15         <input type="text" name="sighting_date" id="datepicker" />
16     </form>
17 </body>
18 </html>
```

sightings.html

```
$(document).ready(function() {
    $('#datepicker').datepicker({ changeMonth: true, changeYear: true });
}); //end doc ready
```

↖ The datepicker code



my_scripts.js

실습과제 21-1 Test Drive

Submit Your Cryptid Sighting

Fill out the form below and click "Enter" to Submit

CRYPTID SIGHTING DATA

Date of Sighting:

Nov 2019

Su	Mo	Tu	We	Th	Fr	Sa
					1	2
3	4	5	6	7	8	9
10	11	12	13	14	15	16
17	18	19	20	21	22	23
24	25	26	27	28	29	30

CRYPTID LOCATION DATA

Latitude of Sighting:

Longitude of Sighting:

Submit

<http://ksamkeun.dothome.co.kr/wp/jquery/ch10/step1/sightings.html>

Check it off

- ~~1. 사용자가 신비생물을 본 날짜를 입력하는 날짜 선택기를 만든다.~~
2. 사용자가 본 신비생물이 어떤 타입인지 선택하는 좀 더 매력적인 라디오 버튼을 만든다.
3. 신비생물을 얼마나 멀리서 봤는지, 신비생물의 키와 몸무게는 얼마나 되는지, 신비생물이 있던 장소의 위도/경도 값은 얼마인지 입력할 숫자 필드를 슬라이더로 바꾼다.
4. 신비생물이 무슨 색이었는지 입력할 수 있도록 색상 믹서를 만든다.
5. 제출(submit) 버튼을 더 멋져 보이게 만든다.

Styling up your buttons

Here's the HTML for a single button.

`<input type="radio" id="radio1" name="radio" />`

This input will be updated when the user clicks.

`<label for="radio1">Choice 1</label>`

The button widget styles the label to appear as a button.

And the corresponding jQuery statement

`$ ("#radio1").button();`

The button method turns a plain old HTML radio button into a nice-looking, more interactive button.

Grouping button widgets

Put your radio button group inside a container element.

```
<div id="radio">
```

```
    <input type="radio" id="radio1" name="radio" />
```

```
    <label for="radio1">Choice 1</label>
```

```
    <input type="radio" id="radio2" name="radio" />
```

```
    <label for="radio2">Choice 2</label>
```

```
    <input type="radio" id="radio3" name="radio" />
```

```
    <label for="radio3">Choice 3</label>
```

```
</div>
```

In your jQuery code, select the container element.

```
$( "#radio" ).buttonset();
```

The buttonset method groups the buttons for you and runs the button method on each element.

jQuery UI Magnets

```
<div id="type_select">
```

```
<input type="radio" id="radio1" name="creature_type" />
```

```
<label for="radio1">Chupacabras</label>
```

```
<input type="radio" id="radio2" name="creature_type" />
```

```
<label for="radio2">Jersey Devil</label>
```

```
<input type="radio" id="radio3" name="creature_type" />
```

```
<label for="radio3">Loch Ness Monster</label>
```

```
<input type="radio" id="radio4" name="creature_type" />
```

```
<label for="radio4">Sasquatch</label>
```

```
</div>
```



sightings.html

In the code folder, you'll find that we added buttons for Yeti and Other too. We just ran out of space in this exercise.

```
$( "#type_select" ).buttonset();
```



my_scripts.js

실습과제 21-2 Test Drive

Creature Type:

☐ Chupacabras ☐ Jersey Devil ☐ Loch Ness Monster ☒ Sasquatch ☐ Yeti ☐ Other

*You took these plain old
HTML radio buttons...*

Creature Type:

<http://ksamkeun.dohome.co.kr/wp/jquery/ch10/step2/sightings.html>

Check it off

- ~~1. 사용자가 신비생물을 본 날짜를 입력하는 날짜 선택기를 만든다.~~
- ~~2. 사용자가 본 신비생물이 어떤 타입인지 선택하는 좀 더 매력적인 라디오 버튼을 만든다.~~
3. 신비생물을 얼마나 멀리서 봤는지, 신비생물의 키와 몸무게는 얼마나 되는지, 신비생물이 있던 장소의 위도/경도 값은 얼마인지 입력할 숫자 필드를 **슬라이더**로 바꾼다.
4. 신비생물이 무슨 색이었는지 입력할 수 있도록 색상 믹서를 만든다.
5. 제출(submit) 버튼을 더 멋져 보이게 만든다.

Control numerical entries with a slider

jQuery UI slider plug-in

사용자가 마우스나 키보드로 조작할 수 있는 슬라이더 인터페이스를 만들자:

This sets up the slider in your HTML file.

```
<div id="slide_me"></div>
```

And here's the related jQuery code

```
$("#slide_me").slider();
```

The user can move the slider handle with the mouse cursor or with the arrow keys on the keyboard.



As the user slides the slider handle along the slider track, the slider's value changes

Control numerical entries with a slider

슬라이더 위젯의 옵션들:

```
$( "#slide_me" ).slider({
```

The **value** option tells the slider what value to start with.

```
  value: 0,
```

The **min** option tells the slider the lowest value a user can enter.

```
  min: 0,
```

```
  max: 100,
```

The **max** option tells the slider the highest value a user can enter.

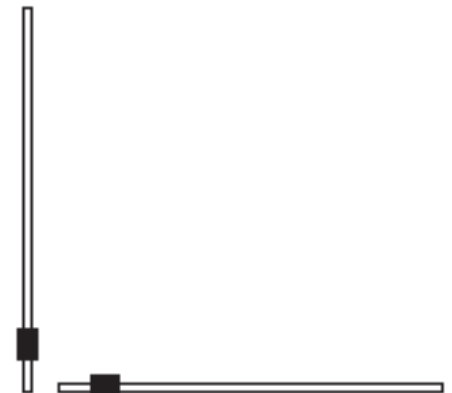
The **step** option tells the slider what increments we want the values in.

```
  step: 5,
```

```
  orientation: 'vertical'
```

```
});
```

The **orientation** option can be set to horizontal or vertical.





슬라이더 값을 어떻게
폼 input에 연결할 수
있을까?

슬라이더를 **슬라이더 위젯의 이벤트 핸들러** 중의 하나와 연결
하면 된다.

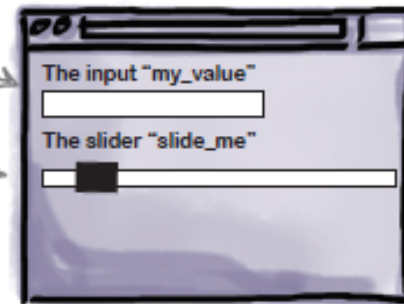
jQuery UI의 또다른 강력한 특징
⇒ 많은 이벤트 핸들러 제공

슬라이더 이벤트 핸들러
⇒ 슬라이더를 폼 input에 연결

To connect the slider to a form input, let's try out the `slide` event handler

The HTML for a slider widget

```
<input type="text" id="my_value" readonly="readonly"/>
<div id="slide_me"></div>
```



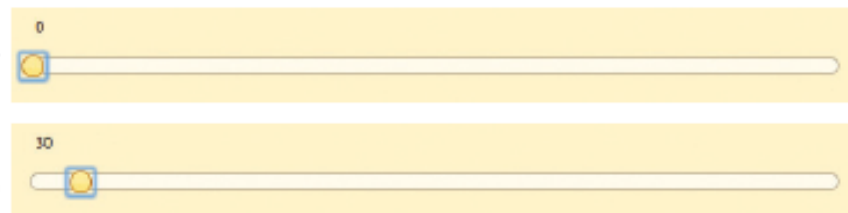
The JQuery script for a slider widget

```
$( "#slide_me" ).slider({
  slide: function( event, ui ) {
    $( "#my_value" ).val( ui.value );
  }
});
```

This is the slide event handler. The user triggers the slide event when she moves the slide handle.

The slide event is attached to a function callback. When the function runs, this sets the input with the jquery `val` method.

When the user slides the slide handle, the function is called and the input is updated to the slider's value.



Exercise

Distance from Creature (in ft.):
Starting value should be 0.
Minimum distance should be 0.
Maximum distance should be 500.
Use increments of 10 feet.

```
<h3>Distance from Creature (in ft.):</h3>
<input type="text" id="distance" class="just_display" name="creature_distance"
      readonly="readonly"/>

  <div id="slide_dist"></div>
</div>
</div>
```



sightings.html

```
$( "#slide_dist" ).slider({
  value:0,
  min:0,
  max:500,
  step:10,
  slide: function( event, ui ) {
    $( "#distance" ).val( ui.value );
  }
});
```



my_scripts.js

Exercise

Creature Weight (in lbs.):
Starting value should be 0.
Minimum weight should be 0.
Maximum weight should be 5,000.
Use increments of 5 pounds.

Creature Height (in ft.):
Starting value should be 0.
Minimum height should be 0.
Maximum height should be 20.
Use increments of 1 foot

```
<h3>Creature Weight (in lbs.)</h3>
  <input type="text" id="weight" class="just_display" name="creature_weight"
        readonly="readonly"/>
  <div id="slide_weight"></div>

<h3>Creature Height (in ft.)</h3>
  <input type="text" id="height" class="just_display" name="creature_height"
        readonly="readonly"/>
  <div id="slide_height"></div>
```



sightings.html

```
$( "#slide_height" ).slider({
    value:0,
    min:0,
    max:20,
    step:1,
    slide: function( event, ui ) {
        $( "#height" ).val( ui.value );
    }
});
```



my_scripts.js

```
$( "#slide_weight" ).slider({
    value:0,
    min:0,
    max:5000,
    step:10,
    slide: function( event, ui ) {
        $( "#weight" ).val( ui.value );
    }
});
```



my_scripts.js

실습과제 21-3 Test Drive

Submit Your Cryptid Sighting

Fill out the form below and click "Enter" to Submit

CRYPTID SIGHTING DATA

Date of Sighting:
11/21/2019

Creature Type:
Chupacabras Jersey Devil Loch Ness Monster Sasquatch Yeti Other

Distance from Creature (in ft.):
60

Creature Weight (in lbs.):
1365

Creature Height (in ft.):
12

Color of Creature:

CRYPTID LOCATION DATA

Latitude of Sighting:

Longitude of Sighting:

Submit

<http://ksamkeun.dothome.co.kr/wp/jquery/ch10/step3/sightings.html>



경위도 값은 음수를 가질 수도 있고 0.0000이처럼
아주 작은 십진수 값을 가질 수도 있다. 슬라이더
위젯이 이러한 숫자를 다룰 수 있나요?

jQuery UI has you covered there, too.

The slider widget can deal with negative numbers and decimal numbers. You can enter negative numbers as values, as minimums, and as maximums. Give that a try below to see it in action.

슬라이더 위젯 => 음수 값 및 아주 작은 십진수 값도 다룰 수 있다!

Sharpen your pencil

경도, 위도에 대한 슬라이더를 만드시오.

```
$( "#slide_lat" ).slider({  
    value:0,  
    min: -90,  
    max: 90,  
    step: 0.00001,  
    slide: function( event, ui ) {  
        $( " latitude " ).val( ui.value );  
    }  
});
```

```
$( "#slide_long" ).slider({  
    value:0,  
    min: -180,  
    max: 180,  
    step: 0.00001,  
    slide: function( event, ui ) {  
        $( " longitude " ).val( ui.value );  
    }  
});
```



my_scripts.js

실습과제 21-4 Test Drive

Submit Your Cryptid Sighting x +

← → ↻ 주의 요함 | ksamkeun.dothome.co.kr/wp/jquery/ch10/step3.1/sightings.html ☆ 1 2 3 4 5 6 7 8 9 10 11 12 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 44 45 46 47 48 49 50 51 52 53 54 55 56 57 58 59 60 61 62 63 64 65 66 67 68 69 70 71 72 73 74 75 76 77 78 79 80 81 82 83 84 85 86 87 88 89 90 91 92 93 94 95 96 97 98 99 100 101 102 103 104 105 106 107 108 109 110 111 112 113 114 115 116 117 118 119 120 121 122 123 124 125 126 127 128 129 130 131 132 133 134 135 136 137 138 139 140 141 142 143 144 145 146 147 148 149 150 151 152 153 154 155 156 157 158 159 160 161 162 163 164 165 166 167 168 169 170 171 172 173 174 175 176 177 178 179 180 181 182 183 184 185 186 187 188 189 190 191 192 193 194 195 196 197 198 199 200 201 202 203 204 205 206 207 208 209 210 211 212 213 214 215 216 217 218 219 220 221 222 223 224 225 226 227 228 229 230 231 232 233 234 235 236 237 238 239 240 241 242 243 244 245 246 247 248 249 250 251 252 253 254 255 256 257 258 259 260 261 262 263 264 265 266 267 268 269 270 271 272 273 274 275 276 277 278 279 280 281 282 283 284 285 286 287 288 289 290 291 292 293 294 295 296 297 298 299 300 301 302 303 304 305 306 307 308 309 310 311 312 313 314 315 316 317 318 319 320 321 322 323 324 325 326 327 328 329 330 331 332 333 334 335 336 337 338 339 340 341 342 343 344 345 346 347 348 349 350 351 352 353 354 355 356 357 358 359 360 361 362 363 364 365 366 367 368 369 370 371 372 373 374 375 376 377 378 379 380 381 382 383 384 385 386 387 388 389 390 391 392 393 394 395 396 397 398 399 400 401 402 403 404 405 406 407 408 409 410 411 412 413 414 415 416 417 418 419 420 421 422 423 424 425 426 427 428 429 430 431 432 433 434 435 436 437 438 439 440 441 442 443 444 445 446 447 448 449 450 451 452 453 454 455 456 457 458 459 460 461 462 463 464 465 466 467 468 469 470 471 472 473 474 475 476 477 478 479 480 481 482 483 484 485 486 487 488 489 490 491 492 493 494 495 496 497 498 499 500 501 502 503 504 505 506 507 508 509 510 511 512 513 514 515 516 517 518 519 520 521 522 523 524 525 526 527 528 529 530 531 532 533 534 535 536 537 538 539 540 541 542 543 544 545 546 547 548 549 550 551 552 553 554 555 556 557 558 559 560 561 562 563 564 565 566 567 568 569 570 571 572 573 574 575 576 577 578 579 580 581 582 583 584 585 586 587 588 589 590 591 592 593 594 595 596 597 598 599 600 601 602 603 604 605 606 607 608 609 610 611 612 613 614 615 616 617 618 619 620 621 622 623 624 625 626 627 628 629 630 631 632 633 634 635 636 637 638 639 640 641 642 643 644 645 646 647 648 649 650 651 652 653 654 655 656 657 658 659 660 661 662 663 664 665 666 667 668 669 670 671 672 673 674 675 676 677 678 679 680 681 682 683 684 685 686 687 688 689 690 691 692 693 694 695 696 697 698 699 700 701 702 703 704 705 706 707 708 709 710 711 712 713 714 715 716 717 718 719 720 721 722 723 724 725 726 727 728 729 730 731 732 733 734 735 736 737 738 739 740 741 742 743 744 745 746 747 748 749 750 751 752 753 754 755 756 757 758 759 760 761 762 763 764 765 766 767 768 769 770 771 772 773 774 775 776 777 778 779 780 781 782 783 784 785 786 787 788 789 790 791 792 793 794 795 796 797 798 799 800 801 802 803 804 805 806 807 808 809 810 811 812 813 814 815 816 817 818 819 820 821 822 823 824 825 826 827 828 829 830 831 832 833 834 835 836 837 838 839 840 841 842 843 844 845 846 847 848 849 850 851 852 853 854 855 856 857 858 859 860 861 862 863 864 865 866 867 868 869 870 871 872 873 874 875 876 877 878 879 880 881 882 883 884 885 886 887 888 889 890 891 892 893 894 895 896 897 898 899 900 901 902 903 904 905 906 907 908 909 910 911 912 913 914 915 916 917 918 919 920 921 922 923 924 925 926 927 928 929 930 931 932 933 934 935 936 937 938 939 940 941 942 943 944 945 946 947 948 949 950 951 952 953 954 955 956 957 958 959 960 961 962 963 964 965 966 967 968 969 970 971 972 973 974 975 976 977 978 979 980 981 982 983 984 985 986 987 988 989 990 991 992 993 994 995 996 997 998 999 1000

Submit Your Cryptid Sighting

Fill out the form below and click "Enter" to Submit

CRYPTID SIGHTING DATA

Date of Sighting:
11/21/2019

Creature Type:
Chupacabras Jersey Devil Loch Ness Monster **Sasquatch** Yeti Other

Distance from Creature (in ft.):
90

Creature Weight (in lbs.):
1450

Creature Height (in ft.):
9

Color of Creature:

CRYPTID LOCATION DATA

Latitude of Sighting:
38.03475

Longitude of Sighting:
128.05505

Submit

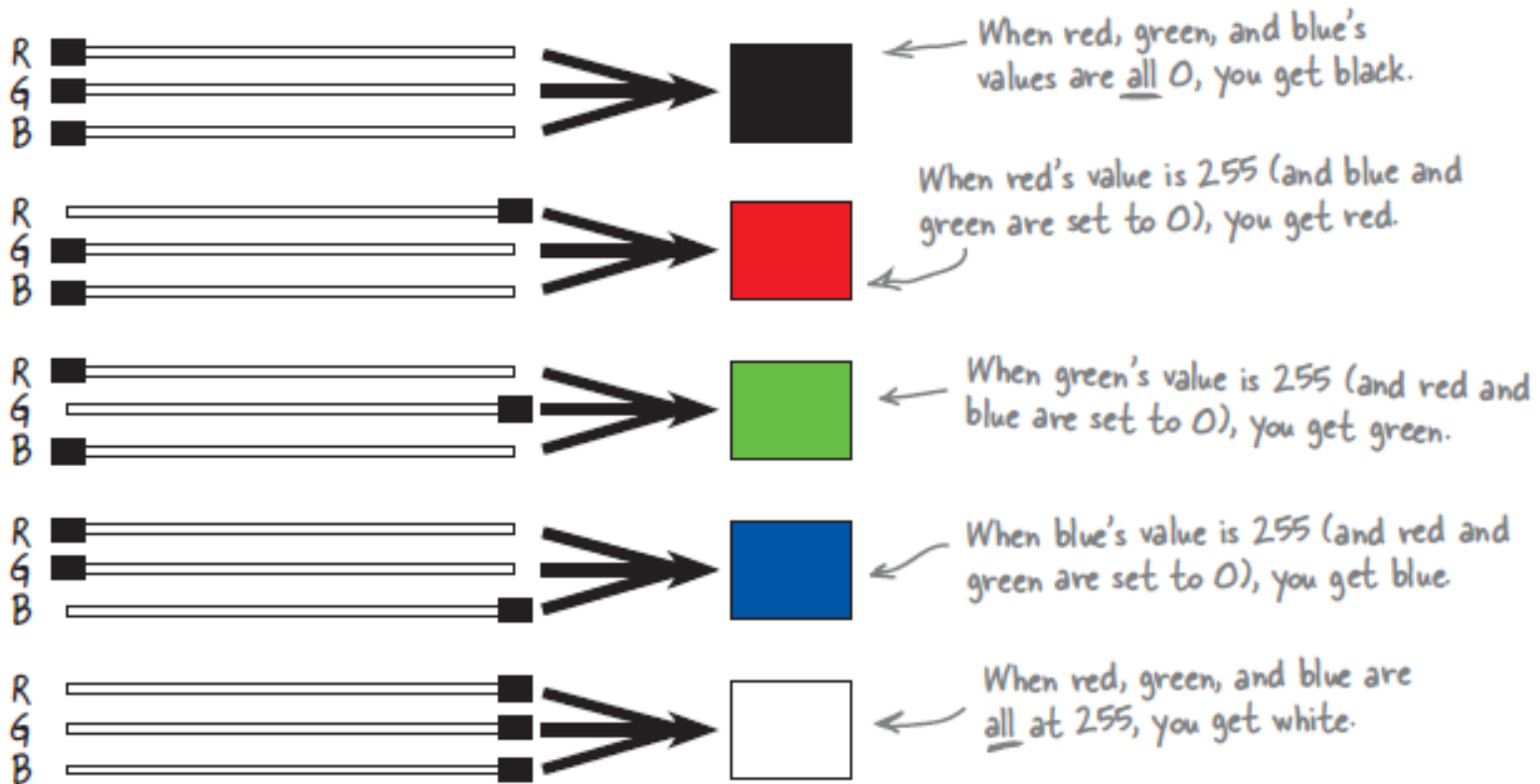
<http://ksamkeun.dothome.co.kr/wp/jquery/ch10/step4/sightings.html>

Check it off

- ~~1. 사용자가 신비생물을 본 날짜를 입력하는 날짜 선택기를 만든다.~~
- ~~2. 사용자가 본 신비생물이 어떤 타입인지 선택하는 좀 더 매력적인 라디오 버튼을 만든다.~~
- ~~3. 신비생물을 얼마나 멀리서 봤는지, 신비생물의 키와 몸무게는 얼마나 되는지, 신비생물이 있던 장소의 위도/경도 값은 얼마인지 입력할 숫자 필드를 슬라이더로 바꾼다.~~
4. 신비생물이 무슨 색이었는지 입력할 수 있도록 색상 믹서를 만든다.
5. 제출 버튼을 더 멋지게 만든다.

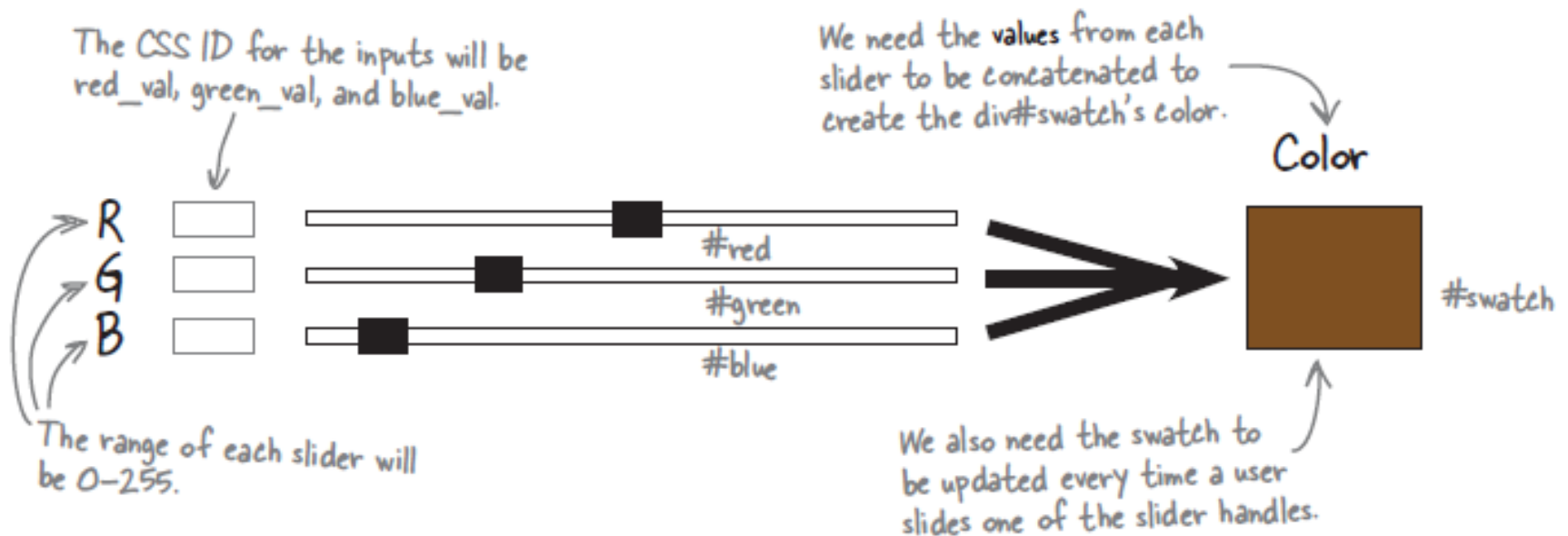
Computers mix color using red, green, and blue

The values of red, green, and blue each have a minimum of 0 and a maximum of 255. When each color is at its minimum—in other words, when red's value is 0, green's is 0, and blue's is 0—you get black. When each color is at its maximum—when red's value is 255, green's is 255, and blue's is 255—you get white.



Your sliders need to do the same

So you need to build **three** different sliders: one for red, one for green, and one for blue. Then you'll **combine** each of the slider's values to become one color. Let's look at what we need each slider widget to do.



Ready Bake Code

```
//color slider styles
#slide_dist, #slide_weight, #slide_height {
margin-bottom:14px;
}
```

```
#swatch {
width: 75px;
height: 75px;
background-image: none;
}
```

This CSS ID style defines the swatch that will show the color as the user changes the slider.

```
#red .ui-slider-range { background: #ef2929; }
#red .ui-slider-handle { border-color: #ef2929; }
```

```
#green .ui-slider-range { background: #8ae234; }
#green .ui-slider-handle { border-color: #8ae234; }
```

```
#blue .ui-slider-range { background: #729fcf; }
#blue .ui-slider-handle { border-color: #729fcf; }
```



form.css

Each of these styles puts the color of the slider in the slider track.

Ready Bake Code

```
<h3>Color of Creature (use the color sliders to enter):</h3>  
Color (in hexadecimal):<input type="text" class="just_display"  
name="creature_color" id="color_val" readonly="readonly"/><br /><br />  
  <div id="swatch" class="ui-widget-content ui-corner-all"></div>  
Red:<input type="text" class="just_display" name="creature_color" id="red_val"  
readonly="readonly"/>  
  <div id="red"></div>  
Green:<input type="text" class="just_display" name="creature_color" id="green_val"  
readonly="readonly"/>  
  <div id="green"></div>  
Blue:<input type="text" class="just_display" name="creature_color" id="blue_val"  
readonly="readonly"/>  
  <div id="blue"></div>
```

The input field that will hold the hex value of the color

The div for the color swatch

The div for the red slider

The div for the green slider

The div for the blue slider



sightings.html

Sharpen your pencil

The script below sets up the red, green, and blue color sliders.

<pre><code>\$("#red, #green, #blue").slider({ orientation: "horizontal", range: "min", max: 255, value: 127, slide: <u>refreshSwatch</u>, change: refreshSwatch });</code></pre>	<p>Turns each of the divs for R,G,B into slider widgets.</p> <p>Makes them into horizontal sliders rather than vertical.</p> <p>Sets the slider range so the user can only select a maximum.</p> <p>Sets the max value to 255 to stick within the color limits.</p> <p>Sets the value so the slider handles are roughly in the middle.</p> <p>Calls a function named refreshSwatch when the user slides.</p> <p>Calls the same function when any value changes.</p>
--	---

my_scripts.js

Build the refreshSwatch function

컬러 믹서를 완성하기 위해 swatch 설정 및 refresh할 JavaScript 함수가 필요하다.

```
function refreshSwatch() {  
    var red = ???  
    var green = ???  
    var blue = ???  
    var my_rgb = ???  
    $( "#swatch" ). ???;  
    $( "#red_val" ).val( red );  
    $( "#blue_val" ).val( blue );  
    $( "#green_val" ).val( green );  
    $( "#color_val" ).val( my_rgb );  
}
```

How will you get the values of each slider into each of these variables?

We need to concatenate the RGB values in this variable so that we can set the color swatch.

What jQuery method will let us set the color swatch's color?

No big questions here. We can simply use jQuery's val method to set the input fields to show the value of the sliders as they change. That way, the user will know what the values are.

my_scripts.js

우리는 이미 슬라이더로부터 값을 얻어오는 방법을 알고 있다.

그러나 컬러 swatch는 어떻게 생성할 것인가? 16진수

○ ○ 웹 컬러를 변환하기 위한 한 묶음의 코드가 필요하지 않나?



Good catch! We *could* write a decimal-to-hexadecimal conversion function, or we could use the decimal values straight from the sliders.

Remember that the CSS `background-color` property allows us to specify colors like this:

R G B
↓ ↓ ↓
`background-color: rgb(255, 0, 255)`

But that's just a hint for one of the questions. To write the whole function, you'll have to do a little brain workout.

Exercise

Fill in the blank lines of code to finish off the refreshSwatch function.

```
function refreshSwatch() {  
    var red = $( "#red" ).slider( "value" );  
    var green = $( "#green" ).slider( "value" );  
    var blue = $( "#blue" ).slider( "value" );  
    var my_rgb = "rgb(" + red + "," + green + "," + blue + ")";  
    $( "#swatch" ). $( "#swatch" ).css( "background-color", my_rgb );  
    $( "#red_val" ).val( red );  
    $( "#blue_val" ).val( blue );  
    $( "#green_val" ).val( green );  
    $( "#color_val" ).val( my_rgb );  
}
```

By concatenating the RGB values into this variable...

...we can set the CSS for the swatch to the combined values of the three colors.

my_scripts.js

Ready Bake Code

Add the lines below too. They will trigger the slider values to run the **refreshSwatch** function when the web page loads, which will start the web page with a colored swatch rather than an empty one.

```
$( "#red" ).slider( "value", 127 );  
$( "#green" ).slider( "value", 127 );  
$( "#blue" ).slider( "value", 127 );
```

my_scripts.js

실습과제 21-5 Test Drive

Submit Your Cryptid Sighting x +

← → ↻ 주의 요약 | ksamkeun.dothome.co.kr/wp/jquery/ch10/step5/sightings.html

Submit Your Cryptid Sighting

Fill out the form below and click "Enter" to Submit

CRYPTID SIGHTING DATA

Date of Sighting:
11/21/2019

Creature Type:
Chupacabras Jersey Devil Loch Ness Monster Sasquatch Yeti Other

Distance from Creature (in ft.):

Creature Weight (in lbs.):

Creature Height (in ft.):

Color of Creature (use the color sliders to enter):
Color:rgb(127,78,164)

Red:127
Green:78
Blue:164

CRYPTID LOCATION DATA

Latitude of Sighting:
-15.311

Longitude of Sighting:
-104.95688

Submit

<http://ksamkeun.dothome.co.kr/wp/jquery/ch10/step5/sightings.html>

Check it off

- ~~1. 사용자가 신비생물을 본 날짜를 입력하는 날짜 선택기를 만든다.~~
- ~~2. 사용자가 본 신비생물이 어떤 타입인지 선택하는 좀 더 매력적인 라디오 버튼을 만든다.~~
- ~~3. 신비생물을 얼마나 멀리서 봤는지, 신비생물의 키와 몸무게는 얼마나 되는지, 신비생물이 있던 장소의 위도/경도 값은 얼마인지 입력할 숫자 필드를 슬라이더로 바꾼다.~~
- ~~4. 신비생물이 무슨 색이었는지 입력할 수 있도록 색상 믹서를 만든다.~~
5. **제출 버튼을 더 멋지게 만든다.**



Wait. Sure, we've used buttons since Chapter 1, but this is a jQuery UI button and an HTML form element.

OK, good point.

We've used jQuery's `click` method and jQuery UI's `button` method, but we haven't done much with selecting form elements like input submit buttons. Here's a quick guide on how to select those.



BULLET POINTS

- `$(":input")` = Select all input elements
- `$(":text")` = Select all elements of type text
- `$(":radio")` = Select all input elements of type radio
- `$(":checkbox")` = Select all elements of type checkbox
- `$(":submit")` = Select all input elements of type submit
- `$(":reset")` = Select all input elements of type reset
- `$(":checked")` = Select all inputs that have been checked
- `$(":selected")` = Select all inputs that have been selected
- `$(":enabled")` = Select all inputs that are enabled
- `$(":disabled")` = Select all inputs that are disabled
- `$(":password")` = Select all inputs that are intended for passwords

Exercise

Be sure to add this to your **my_scripts.js** file and test it out by opening **sightings.html** in your favorite browser.

```

$( "#button:submit" ).button();

$("#frmAddSighting").submit(function(){
    return false;
});

$('#btnSave').click(function() {

    var data = $("#frmAddSighting :input").serializeArray();

    $.post($("#frmAddSighting").attr('action'), data, function(json){

        if (json.status == "fail") {
            alert(json.message);
        }else if (json.status == "success") {
            alert(json.message);
        }else{alert("Nothing Happened");}
    }, "json");

});

```

```
my_scripts.js
```

sightings.html

```
<div class="ui-widget-header ui-corner-bottom form_pad">  
  <button type="submit" id="btnSave">Submit</button>  
  <input type="hidden" name="action" value="addSighting" id="action">  
</div>
```

실습과제 21-6 Test Drive

The screenshot shows a web browser window with the address bar displaying the URL: <http://ksamkeun.dothome.co.kr/wp/jquery/ch10/step6/sightings.html>. The page title is "Submit Your Cryptid Sighting". Below the title, there is a instruction: "Fill out the form below and click 'Enter' to Submit".

The form is divided into two main sections: "CRYPTID SIGHTING DATA" and "CRYPTID LOCATION DATA".

CRYPTID SIGHTING DATA

- Date of Sighting:** A text input field.
- Creature Type:** A row of buttons: "Chupacabras", "Jersey Devil", "Loch Ness Monster", "Sasquatch", "Yeti", and "Other".
- Distance from Creature (in ft.):** A slider control with a value of 0.
- Creature Weight (in lbs.):** A slider control with a value of 0.
- Creature Height (in ft.):** A slider control with a value of 0.
- Color of Creature (use the color sliders to enter):**
 - A color preview box showing a dark gray color.
 - Below the preview, the text "Color:rgb(127,127,127)" is displayed.
 - Three horizontal sliders for Red, Green, and Blue, each with a value of 127.

CRYPTID LOCATION DATA

- Latitude of Sighting:** A slider control.
- Longitude of Sighting:** A slider control.

At the bottom of the form, there is a "Submit" button.

<http://ksamkeun.dothome.co.kr/wp/jquery/ch10/step6/sightings.html>

실습과제 21-7 MySQL DB 이용

Your form looks great, but it doesn't actually **submit** any data at this point. However, you learned everything you need to do that back in Chapter 9, so take some time to think through how you could make this stylish form truly functional.

sightings.html, **service.php**, and **sightings.sql**.

You will have to do some work on your own to get it set up, but that's what being a web developer is all about, isn't it? We included the Ajax and JSON methods you learned in Chapter 9 so that the form you just built can submit data.

You need to do all of the setup on your own (i.e., run the **sightings.sql** script or create the database with the fields in that script yourself, and add the Ajax and JSON methods to your **my_scripts.js** file).

5마리 이상의 미확인 동물을 입력하시오.

20장 강의 service.php 복사
⇒ 커스터마이징!

sightings.sql

```
SET SQL_MODE="NO_AUTO_VALUE_ON_ZERO";
```

```
CREATE TABLE IF NOT EXISTS sightings (  
    sighting_id INT NOT NULL AUTO_INCREMENT,  
    sighting_date date NOT NULL,  
    creature_type varchar(35) NOT NULL,  
    creature_distance varchar(35) NOT NULL,  
    creature_weight varchar(35) NOT NULL,  
    creature_height varchar(35) NOT NULL,  
    creature_color varchar(35) NOT NULL,  
    sighting_latitude varchar(10) NOT NULL,  
    sighting_longitude varchar(10) NOT NULL,  
    PRIMARY KEY(sighting_id)  
) ENGINE=MyISAM DEFAULT CHARSET=utf8;
```

copy & Paste

YOUR_ID.dothome.co.kr/myadmin/

서버: localhost * 데이터베이스: ksamkeun * 테이블: sightings

보기 구조 SQL 검색 삽입 내보내기 가져오기 권한

테이블 ksamkeun.sightings에 SQL 질의를 실행: ?

```
1 SET SQL_MODE="NO_AUTO_VALUE_ON_ZERO";
2
3
4 CREATE TABLE IF NOT EXISTS sightings (
5 sighting_id INT NOT NULL AUTO_INCREMENT,
6 sighting_date date NOT NULL,
7 creature_type varchar(35) NOT NULL,
8 creature_distance varchar(35) NOT NULL,
9 creature_weight varchar(35) NOT NULL,
10 creature_height varchar(35) NOT NULL,
11 creature_color varchar(35) NOT NULL,
12 sighting_latitude varchar(10) NOT NULL,
13 sighting_longitude varchar(10) NOT NULL,
14 PRIMARY KEY(sighting_id)
15 ) ENGINE=MyISAM DEFAULT CHARSET=utf8;
```

SELECT * SELECT INSERT UPDATE DELETE 제거 형식

Get auto-saved query

or sliders to enter):

<http://ksamkeun.dothome.co.kr/wp/jquery/ch10/end/sightings2.html>

ksamkeun.dothome.co.kr / local x +

← → ↺ ⓘ 주의 요함 | ksamkeun.dothome.co.kr/myadmin/sql.php?server=1&db=ksamkeun&table=sightings&pos=0

☆ I EX 🔍 🌐 📄 📌 📧 👤 ⋮

phpMyAdmin

🏠 📄 📁 ⚙️ 🔄 📧

최근 즐겨찾기

🔍

information_schema

ksamkeun

New

dog

Ingredient

runners

sightings

술(칼럼)

인덱스

Taco

Taco_Ingredients

Taco_Order

Taco_Order_Tacos

🔍 서버: localhost ▶ 데이터베이스: ksamkeun ▶ 테이블: sightings

📄 보기

📄 구조

📄 SQL

🔍 검색

📄 삽입

📄 내보내기

📄 가져오기

📄 권한

🔧 테이블 작업

🔧 트리거

✅ 0 - 13행 표시 중 (합계 14, 질의 실행시간 0.0003 초.)

SELECT * FROM `sightings`

📄 프로파일링 [인라인 편집] [수정] [SQL 해석] [PHP 코드 보기] [다시 보기]

☐ 모두 보기

행 갯수: 25 ▼

행 필터링: 현재 테이블 검색

키로 정렬: 없음 ▼

+ 옵션

← T →

		sighting_id	sighting_date	creature_type	creature_distance	creature_weight	creature_height	creature_color	sighting_latitude	sighting_longitude		
<input type="checkbox"/>	✏ 수정	📄 복사	🚫 삭제	1	2019-07-03	Sasquatch	100	2125	6	rgb(85,143,169)	38.03475	128.05505
<input type="checkbox"/>	✏ 수정	📄 복사	🚫 삭제	2	2019-07-03	Chupacabras	100	680	2	rgb(196,129,70)	-50.57762	19.91426
<input type="checkbox"/>	✏ 수정	📄 복사	🚫 삭제	3	2019-07-03	Jersey Devil	260	680	2	rgb(17,129,70)	-26.10108	-27.50903
<input type="checkbox"/>	✏ 수정	📄 복사	🚫 삭제	4	2019-07-03	Loch Ness Monster	160	3055	6	rgb(17,129,183)	38.74549	-87.19088
<input type="checkbox"/>	✏ 수정	📄 복사	🚫 삭제	5	2019-07-03	Yeti	340	2065	8	rgb(184,129,20)	-70.93863	-55.66787
<input type="checkbox"/>	✏ 수정	📄 복사	🚫 삭제	6	2019-07-03	Other	10	185	6	rgb(250,231,249)	38.85379	127.31724
<input type="checkbox"/>	✏ 수정	📄 복사	🚫 삭제	7	1970-01-01	Sasquatch	110	1370	8	rgb(127,75,115)	42.9945	129.23651
<input type="checkbox"/>	✏ 수정	📄 복사	🚫 삭제	8	2019-07-04	Sasquatch	110	1370	8	rgb(127,75,115)	42.9945	129.23651
<input type="checkbox"/>	✏ 수정	📄 복사	🚫 삭제	9	2019-07-04	Loch Ness Monster	120	900	6	rgb(87,44,127)	-10.21356	41.31972
<input type="checkbox"/>	✏ 수정	📄 복사	🚫 삭제	10	1970-01-01	Other	0	0	0	rgb(127,197,127)	38.03475	41.31972
<input type="checkbox"/>	✏ 수정	📄 복사	🚫 삭제	11	1970-01-01	Yeti	140	450	2	rgb(127,127,127)		
<input type="checkbox"/>	✏ 수정	📄 복사	🚫 삭제	12	2019-09-04	Yeti	140	450	2	rgb(127,127,127)	0.16436	-0.07987
<input type="checkbox"/>	✏ 수정	📄 복사	🚫 삭제	13	1970-01-01	Loch Ness Monster	50	335	2	rgb(127,127,127)		
<input type="checkbox"/>	✏ 수정	📄 복사	🚫 삭제	14	2019-11-21	Yeti	70	1550	12	rgb(27,99,197)	-13.00708	81.91038

⬆️ ☐ 모두 체크

선택한 것을: ✏ 수정 📄 복사 🚫 삭제 📄 내보내기

☐ 모두 보기

행 갯수: 25 ▼

행 필터링: 현재 테이블 검색

키로 정렬: 없음 ▼

📄 쿼리 결과 처리방법

📄 인쇄

📄 클립보드에 복사하기

📄 내보내기

📄 차트 표시

📄 뷰 생성

🖥 콘솔

Windows 전풍 이즈

Q & A

