Solutions:

1. CSS

```
div {
	margin: 5px;
}
#gallery > img {
	width: 60px;
	height: 60px;
	margin: 10px;
}
#gallery {
	width: 20%;
}
p {
	background-color: yellow;
	padding: 20px;
}
#bigPic {
	width: 75%;
}
#bigPic > div {
	clear: left;
	border: none;
}
#gallery, #bigPic {
	float: left;
}
#more {
	clear: left;
}
#contentArea {
	width: 90%;
	margin-left: auto;
	margin-right: auto;
	background-color: white;
}
body {
	background-color: gray;
}
h3 {
	text-align: right;
}
div, h3 {
	border: 2px black solid;
}
```

2. NodeJS

```
app.get('/', function (req, res) {
	var word = req.query.word;

	if (word && word. match(/^[a-zA-Z]+$/) {
		var badword = word.toLowerCase();
		var lines = "";
		var file = fs.readFileSync("text.txt", 'utf8').split("\n");
		for (var i = 0; i < file.length; i++) {
			var line = file[i].toLowerCase().split(" ");
			var badline = False;
			for (var j = 0; j < line.length; j++) {
				if (line[j] == badword) {
					badline = True;
				}
			}
			if (!badline) {
				lines+= file[i]) + "\n";
			}
		}
		fs.appendFile("text.txt", lines, function(err) {
			if(err) {
				return console.log(err);
			}
		});

		res.send(lines);
	} else {
		res.status(400);
		res.send("oops");
	}
})
```

3.

```
app.get('/', function (req, res) {
      var start = req.query.start;
      var end = req.query.end;

      var file = fs.readFileSync("classes.txt", 'utf8').split(“\n”);
      var classCount = 0;
      var classes = []
      for (var i = 0; i < file.length; i++) {
            var line = file[i].split(" ", 6);
            if(line[1] == start && line[2] == end) {
                  classCount++;
                  if(line[3] < line[4]) {
                        var class = {"code" : line[0],
                                     "spaces" : (line[4]-line[3]),
                                     "name" : line[5]};
                        classes.push(class);
                  }
            }
      }
      var all = {"count" : classCount, "classes" : classes};
      res.json(all);
})
```

4.

```
window.onload = function() {
	document.getElementById("search").onclick = search;
}

function search() {
	document.getElementById("count").innerHTML = "";
	document.getElementById("results").innerHTML = "";
	var start = document.getElementById("start").value;
	var end = document.getElementById("end").value;

	var url = "http://localhost:3000? ?start=" + start + "&end=" + end;
	fetch(url)
		.then(checkStatus)
		.then(function(responseText) {
			var data = JSON.parse(this.responseText);
			for(var i = 0; i < data.classes.length; i++) {
				var item = document.createElement("li");
				item.innerHTML = data.classes[i].name + " - " +
						data.classes[i].spaces + " spaces"
				document.getElementById("results").appendChild(item);
			}
			document.getElementById("count").innerHTML = data.count +
								" total classes offered";

		})
		.catch(function(error) {
			console.log(error);
		});
	}
}
```

5.

a) /^#[A-Fa-f0-9]{6}$/

b) /^5[1-5][0-9]{14}$/

c) /^([1-9]|10|11|12):[0-5][0-9] (A|P)M$/

6.

```
window.onload = function() {
	document.getElementById("add").onclick = add;

}

function add() {
	var el = document.createElement("div");
	el.innerHTML = document.getElementById("input").value;
	el.onmouseenter = color;
	el.onmouseleave = uncolor;
	el.onclick = del;
	document.getElementById("game").appendChild(el);
}

function color() {
	this.style.backgroundColor = "cyan";
	this.style.fontSize =
		parseInt(window.getComputedStyle(this).fontSize) + 5 + "px";
}

function uncolor() {
	this.style.backgroundColor = "white";
	this.style.fontSize =
		parseInt(window.getComputedStyle(this).fontSize) - 5 + "px";
}

function del() {
	this.parentNode.removeChild(this);
}
```

4. SQL

```
SELECT DISTINCT l.language FROM languages l
JOIN languages l2 ON l.language = l2.language
JOIN countries co1 ON l.country_code = co1.code
JOIN countries co2 ON l2.country_code = co2.code
JOIN cities ci1 ON co1.code = ci1.country_code
JOIN cities ci2 ON co2.code = ci2.country_code
WHERE ci1.population > 1000000 AND ci2.population > 1000000
  AND co1.code <> co2.code AND l.official = TRUE AND l2.official = TRUE
  AND co1.independence_year > 1900 AND co2.independence_year > 1900
ORDER BY l.language
```