JSON Statham

CS 337 JSON

Benjamin Dicken

JSON

- JavaScript Object Notation
- Basically, a text representation of Js objects / arrays / key+value pairs
- Easily human-readable, easily machine-readable
- A data format, useful for transferring data to/from Js clients and servers
  - (as opposed to CSV, XML, etc)

An Example

people.csv

```
name,hgt,lbs
Joe,75,200
Jack,60,170
John,70,175
```

people.json

```
{ "people": [
  ["Joe", 75, 200],
  ["Jack", 60, 170],
  ["John", 70, 175]
]}
```

An Example

people.csv

```
name,hgt,lbs
Joe,75,200
Jack,60,170
John,70,175
```

people.json

```
{ "people": [
  {"name":"Joe", "hgt":75, "wgt":200},
  {"name":"Jack", "hgt":60, "wgt":170},
  {"name":"John", "hgt":70, "wgt":175}
]}
```

Represent a responsibilities item with JSON

```
{"name": "complete PA 8",
 "stat": "INPROGRESS",
 "difficulty": 7,
 "list": "school" }
```

Represent multiple lists item with JSON

```
{ "item": [
  { "name": "complete PA 8",
    "stat": "INPROGRESS",
    "difficulty": 7,
    "list": "school" },
  { "name": "find final project group",
    "stat": "INPROGRESS",
    "difficulty": 5,
    "list": "school" },
  . . .
]}
```

Represent multiple lists item with JSON

```json
{
 "school": [
  { "name": "complete PA 8", "stat": "INPROGRESS",
    "difficulty": 7, "list": "school" },
  { "name": "find final project group", "stat": "INPROGRESS",
    "difficulty": 5, "list": "school" } ],
 "TA": [
  { "name": "grade PA 7", "stat": "INPROGRESS",
    "difficulty": 9, "list": "TA" },
  { "name": "grade exams", "stat": "TOBESTARTED",
    "difficulty": 9, "list": "TA" } ],
 . . . .
}
```

JSON

- **`JSON.parse()`** Converts a JSON string to a Javascript object
- **`JSON.stringify()`** Converts a Javascript object to a string

- Can call **stringify** before sending data, and **parse** after receiving data

Change the responsibilities application to send/receive JSON

Change the app to have list Categories as well