2016 json-schema json-ld primers

```
// primer-json-schema.js

// FISHCO1LBC
// TCAACYAATCAYAAAGATATYGGCAC
//
// FISHCO1HBC
// ACTTCYGGGTGRCCRAARAATCA

{
  "id": 1,
  "name": "FISHCO1LBC",
  "description": "Fish CO1 barcoding primer",
  "sequence": "TCAACYAATCAYAAAGATATYGGCAC",
  "orientation": [
    "forward",
    "reverse"
  ]
}

{
    "type": "object",
    "properties": {
        "id": {
            "type": "string",
            "description": "unique id",
            "minLength": 0,
            "maxLength": 255
        },
        "name": {
            "type": "string",
            "description": "short name",
            "minLength": 4,
            "maxLength": 32
        },
        "description": {
            "type": "string"
        },
        "sequence": {
            "type": "string",
            "description": "sequence"
        },
        "orientation": {
            "type": "string",
            "default": "forward",
            "enum": [
                [
                    "forward",
                    "reverse"
                ]
            ]
        }
    },
    "required": [
        "id",
        "name",
        "sequence",
```

```
        "orientation"
      ]
}


{
  "$id": "http://example.com/example.json",
  "type": "object",
  "definitions": {},
  "$schema": "http://json-schema.org/draft-06/schema#",
  "properties": {
    "id": {
      "$id": "/properties/id",
      "type": "integer",
      "title": "id",
      "description": "id string",
      "default": ""
    },
    "name": {
      "$id": "/properties/name",
      "type": "string",
      "title": "The Name Schema",
      "description": "An explanation about the purpose of this instance.",
      "default": "",
      "examples": [
        "FISHCO1LBC"
      ]
    },
    "description": {
      "$id": "/properties/description",
      "type": "string",
      "title": "The Description Schema",
      "description": "An explanation about the purpose of this instance.",
      "default": "",
      "examples": [
        "Fish CO1 barcoding primer"
      ]
    },
    "sequence": {
      "$id": "/properties/sequence",
      "type": "string",
      "title": "The Sequene Schema",
      "description": "An explanation about the purpose of this instance.",
      "default": "",
      "examples": [
        "TCAACYAATCAYAAAGATATYGGCAC"
      ]
    },
    "orientation": {
      "$id": "/properties/orientation",
      "type": "array",
      "uniqueItems": false,
      "items": {
        "$id": "/properties/orientation/items",
        "type": "string",
        "title": "orientation",
        "description": "forward or reverse strand",
        "default": "forward",
        "examples": [
          "forward"
        ],
```

```
        "enum": [
          "forward, reverse"
        ]
      }
    }
  },
  "required": [
    "id",
    "name",
    "sequence"
  ]
}

const mobx = require("mobx")
const { types } = require('mobx-state-tree');
const jsonSchemaToTypes = require('jsonschema-to-mobx-state-tree')(types);

const eventSchema = {
  type: "object",
  title: "Event",
  properties: {
    title: {
      type: "string"
    },
    public: {
      type: "boolean",
      default: false
    },
    time: {
      type: "object",
      properties: {
        start: {
          type: "string",
          format: "datetime"
        },
        end: {
          type: "string",
          format: "datetime"
        }
      },
      required: ["start"]
    }
  },
  required: ["title", "public"]
};

const primer = {
  "type": "object",
  "properties": {
    "id": {
      "type": "string",
      "description": "unique id",
    },
    "name": {
      "type": "string",
      "description": "short name",
      "minLength": 4,
      "maxLength": 32
    },
    "description": {
      "type": "string"
```

```
    },
    "sequence": {
      "type": "string",
      "description": "sequence"
    },
    "orientation": {
      "type": "string",
      "default": "forward",
      "enum": [
        [
          "forward",
          "reverse"
        ]
      ]
    }
  },
  "required": [
    "id",
    "name",
    "sequence",
    "orientation"
  ]
}

const eventModel = jsonSchemaToTypes(primer);
console.log(eventModel);

// https://npm.runkit.com/jsonschema-to-mobx-state-tree
```

json-ld_simple_protocol_flow.json

```
<script type="application/ld+json">
[{
  "@context":
  {
    "schema": "http://schema.org",
    "name": {
      "@id": "http://schema.org/name",
      "@type": "@id"
    },
    "input": {
      "@id": "http://schema.org/url",
      "@type": "@id"
    },
    "content": {
      "@id": "http://schema.org/code",
      "@type": "@id"
    }
  },
  "name": "calculate transformation efficiency",
  "input": "http://fab.bio/100ideas/pcr/1/1sjf2h2",
  "content": "content",
  "dateCreated": "2013-02-14T13:15:03-08:00"
}]
</script>


/*######################*/
```

```
28
29
30 <script type="application/ld+json">
31 {
32    "@context": "http://schema.org",
33    "@type": "Recipe",
34    "name": "calculate transformation efficiency",
35    "url": "https://www.example-petstore.com/",
36    "dateCreated": "2013-02-14T13:15:03-08:00"
37 }
38 </script>
39
40
41 /*########################*/
42
43 <script type="application/ld+json">
44 [{
45   "@context":
46   {
47     "name": {
48       "@id": "http://schema.org/name",
49       "@type": "@id"
50     },
51    "operation": "http://schema.org/Recipe",
52    "input": {
53       "@id": "http://schema.org/url",
54       "@type": "@id"
55     },
56     "content": {
57       "@id": "http://schema.org/code",
58       "@type": "@id"
59     }
60   },
61   "@type": "operation",
62   "name": "calculate transformation efficiency",
63   "input": "http://fab.bio/100ideas/pcr/1/1sjf2h2",
64   "content": "content",
65   "dateCreated": "2013-02-14T13:15:03-08:00"
66 }]
67 </script>
68
69
70 /*########################*/
71
72
73 {
74   "@context": {
75     "name": "http://fab.bio/schema/0.1/name",
76     "homepage": {
77       "@id": "http://xmlns.com/foaf/0.1/workplaceHomepage",
78       "@type": "@id"
79     },
80     "Person": "http://xmlns.com/foaf/0.1/Person"
81   },
82   "@id": "http://me.example.com",
83   "@type": "Person",
84   "name": "John Smith",
85   "homepage": "http://www.example.com/"
86 }
87
88 /*########################*/
```