

Міністерство освіти і науки України
Харківський національний університет радіоелектроніки

Кафедра штучного інтелекту

Дисципліна «Web-технології та web-дизайн»

Практичне заняття №5

«Формати представлення та обміну даними: XML та JSON»

Виконав:

ст. гр. ІТШІ-24-3

Корнієнко Олександр

Леонідович

Прийняла:

ст. викл. кафедри ІІІ

Гриньова О.Є.

Мета: Навчитись моделювати ієрархічні, слабоструктуровані дані у форматі JSON. Опанувати синтаксис та основні концепції JSON Schema для опису правил та обмежень. Навчитись валідувати JSON документи за допомогою зовнішньої схеми. Розвинути навички пошуку та виправлення помилок у структурі даних .

2. Технічне завдання

Необхідно розробити модель даних для сторінки блогу у форматі JSON та створити для неї зовнішню схему валідації. Оберіть будь-яку предметну область (ПрО) та опишіть її у звіті. Треба розробити модель обраної предметної області у вигляді окремого JSON/XML - файлу, де будуть зберігатися інформація про об'єкти ПрО. Заповніть даними JSON/XML -файл. Далі всі завдання для практичної роботи приведено на прикладі предметної області «БЛОГ»! Для тих, хто обрав свою власну ПрО, виконайте аналогічні завдання! Наприклад, для ПрО «БЛОГ» створити окремий файл сторінки блогу, в якому повинні бути описані записи в блозі (блогер у себе на сторінці пише пости на різні теми, ці пости коментують інші користувачі, можливо, блогер відповідає на коментарі). Кожен запис повинен містити не тільки інформативну частину, але також метадані, таку як дата публікації, тема, ключові слова, прикріплені файли, автор коментаря, список коментарів. Навести опис, фрагменти коду та скріншот результату для кожного пункту.

Код JSON-документа (blog.json)

```
{
  "siteOwner": {
    "name": "Олександр",
    "profession": "Frontend Developer",
    "description": "Я цікавлюся програмуванням, дизайном та створенням сайтів."
  },
  "skills": [
    "HTML / CSS",
    "JavaScript",
    "React",
    "C++ (базовий рівень)",
    "Робота в команді"
  ],
  "hobbies": ["Програмування", "Ігри", "Аніме", "Спорт або музика"],
  "projects": [
    {
      "projectId": 1,
      "name": "HTML/CSS Portfolio",
      "technology": "HTML/CSS",
      "duration": "1 рік",
      "githubUrl": "https://github.com/Terracrit/homework"
    },
    {
      "projectId": 2,
      "name": "JavaScript Practice",
      "technology": "JavaScript",
      "duration": "6 місяців",
      "githubUrl": "https://github.com/Terracrit/homework_JS"
    },
    {
      "projectId": 3,
      "name": "React Learning",
      "technology": "React",
      "duration": "Кілька місяців",
      "githubUrl": "https://github.com/Terracrit/my-react"
    }
  ],
  "gallery": [
    {
      "imageId": 1,
      "src": "./images/skrin_1.png",
      "description": "Скріншот першої роботи"
    }
  ]
}
```

```

    {
      "imageId": 2,
      "src": "./images/skrin_2.png",
      "description": "Скріншот другої роботи"
    }
  ],

  "contacts": {
    "phone": "+380960000007",
    "email": "beautyzone@gmail.com",
    "address": "Україна, м. Київ, вул. Тараса Шевченка 5"
  }
}

```

Код зовнішньої JSON Schema (blog.schema.json)

```

{
  "$schema": "https://json-schema.org/draft/2020-12/schema",
  "type": "object",
  "required": [
    "siteOwner",
    "skills",
    "hobbies",
    "projects",
    "gallery",
    "contacts"
  ],
  "additionalProperties": false,

  "properties": {
    "siteOwner": {
      "type": "object",
      "required": ["name", "profession", "description"],
      "additionalProperties": false,
      "properties": {
        "name": { "type": "string", "minLength": 2 },
        "profession": { "type": "string" },
        "description": { "type": "string", "minLength": 10 }
      }
    },
    "skills": {
      "type": "array",
      "items": { "type": "string" },
      "minItems": 1
    },
    "hobbies": {
      "type": "array",

```

```

    "items": { "type": "string" },
    "minItems": 1
  },

  "projects": {
    "type": "array",
    "items": {
      "type": "object",
      "required": [
        "projectId",
        "name",
        "technology",
        "duration",
        "githubUrl"
      ],
      "additionalProperties": false,
      "properties": {
        "projectId": {
          "type": "integer",
          "minimum": 1
        },
        "name": {
          "type": "string",
          "minLength": 3
        },
        "technology": {
          "type": "string"
        },
        "duration": {
          "type": "string"
        },
        "githubUrl": {
          "type": "string",
          "format": "uri"
        }
      }
    }
  },
},

"gallery": {
  "type": "array",
  "items": {
    "type": "object",
    "required": ["imageId", "src", "description"],
    "additionalProperties": false,
    "properties": {
      "imageId": {
        "type": "integer",
        "minimum": 1
      },

```

```

        "src": {
            "type": "string"
        },
        "description": {
            "type": "string"
        }
    }
}
},
"contacts": {
    "type": "object",
    "required": ["phone", "email", "address"],
    "additionalProperties": false,
    "properties": {
        "phone": {
            "type": "string",
            "pattern": "^\\+380\\d{9}$"
        },
        "email": {
            "type": "string",
            "format": "email"
        },
        "address": {
            "type": "string"
        }
    }
}
}
}
}

```

Опис ваших вимог до структури даних:

Кореневий елемент JSON-документа є об'єктом.

Документ повинен містити інформацію про власника сайту, навички, хобі, проєкти, галерею та контактні дані.

Усі розділи є обов'язковими.

Кожен проєкт має унікальний ідентифікатор типу integer більше 0 та посилання на GitHub.

Контактний номер телефону повинен відповідати формату +380XXXXXXXXXX.

Заборонено використання неописаних у схемі властивостей.

An online, interactive JSON Schema validator. Supports JSON Schema Draft 3, Draft 4, Draft 6

Select schema: Custom

```

87     "description": {
88       "type": "string"
89     }
90   },
91 },
92 },
93 },
94 "contacts": {
95   "type": "object",
96   "required": ["phone", "email", "address"],
97   "additionalProperties": false,
98   "properties": {
99     "phone": {
100       "type": "string",
101       "pattern": "^\\+380\\d{9}$"
102     },
103     "email": {
104       "type": "string",
105       "format": "email"
106     },
107     "address": {
108       "type": "string"
109     }
110   }
111 },
112 },
113 },
114 }

```

✓ No errors found. JSON validates against the schema

Input JSON:

```

1 {
2   "siteOwner": {
3     "name": "Олександр",
4     "profession": "Frontend Developer",
5     "description": "Я цікавлюся програмуванням, дизайном та створенням сайтів."
6   },
7
8   "skills": [
9     "HTML / CSS",
10    "JavaScript",
11    "React",
12    "C++ (базовий рівень)",
13    "Робота в команді"
14  ],
15
16  "hobbies": ["Програмування", "Ігри", "Аніме", "Спорт або музика"],
17
18  "projects": [
19    {
20      "projectId": 1,
21      "name": "HTML/CSS Portfolio",
22      "technology": "HTML/CSS",
23      "duration": "1 рік",
24      "githubUrl": "https://github.com/Terracrit/homework"
25    },
26    {
27      "projectId": 2,
28      "name": "JavaScript Practice"

```

Проходження валідації

✖ Found 1 error(s)

Message: **Required properties are missing from object: skills.**
 Schema path: **#/required**

Після видалення skills

```
33 {
34   "proje
35   "name"
```

✖ Found 1 error(s)

Message: **Invalid type. Expected Integer but got Number.**
 Schema path: **#/properties/projects/items/properties/projectId/type**

Після вводу не цілого числа

```
57   "email": "beautyzone@gmail.com",
58   "address": "Україна,
59   }
60 }
61 }
```

✖ Found 1 error(s)

Message: **String 'beautyzone@gmail.' does not validate against format 'email'.**
 Schema path: **#/properties/contacts/properties/email/format**

Products

Documentation

Community

Після вводу не правильного емейлу

```
25   "duration": "1 рік",
26   "githubUrl": "https://git
27   },
28 }
```

✖ Error parsing JSON

Message: **After parsing a value an unexpected character was encountered: "."**

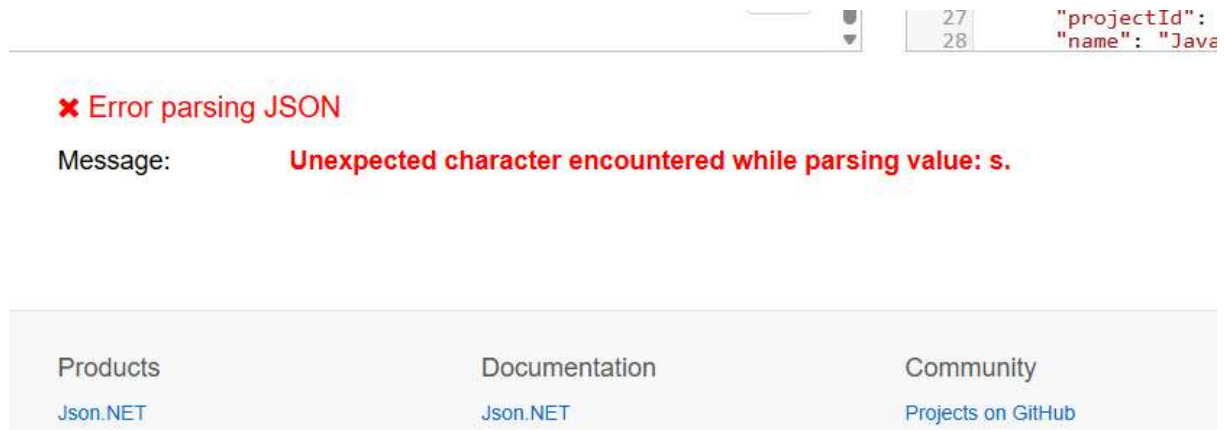
Products

Documentation

Community

Folk

Після вводу чогось не прописаного у схемі



Якщо ввести неправильну структуру об'єкта / масива

Висновок

У ході виконання роботи я два json файли `blog.json` і `blog.schema.json`. Створено JSON-документ із ієрархічною структурою та розроблено зовнішню схему валідації, у якій визначено типи даних, обов'язкові поля, обмеження значень і заборону використання неописаних властивостей. За допомогою онлайн-валідатора було перевірено коректність JSON-документа та змодельовано типові помилки, що дозволило на практиці переконатися в ефективності JSON Schema для виявлення структурних і типових помилок у даних.