

Міністерство Освіти України
Державний університет
Телекомунікацій

Лабораторна робота 7.
Валютний інформер

Підготував: студент групи
ПД-21 Гапей Максим
Юрійович Перевірів:
викладач
Яскевич В.О.

Київ
2021

Мета роботи:

- Поглибити та закріпити знання з асинхронних функцій.
- Придбати практичні навички роботи з Fetch API.
- Навчитись отримувати дані з зовнішніх джерел

Крок 1.

Адреса запиту курсу на поточну дату:

<https://bank.gov.ua/NBUStatService/v1/statdirectory/exchange?json> Структура

відповіді:

```
[
  {
    "r030":840,"txt":"Долар
    США","rate":5.05,"cc":"USD","exchangedate":"27.05.2007"
  }
]
```

Масив об'єктів.Поля:

- txt: назва валюти
- cc: код валюти
- exchangedate: дата.

Запит:

```
> fetch(`https://bank.gov.ua/NBUStatService/v1/statdirectory/exchange?valcode=USD&date=20210530&json`)
  .then(result => result.json())
  .then(obj => console.log(obj));
< ▶ Promise {<pending>}

▼ [{...}] ⓘ
  ▶ 0: {r030: 840, txt: "Долар США", rate: 27.526, cc: "USD", exchangedate: "30.05.2021"}
    length: 1
  ▶ __proto__: Array(0)
```

Крок 2.

Для того, щоб отримати поточну дату можна використовувати вбудований об'єкт Date:

```
let now = new Date();  
  
alert( now ); // покаже поточну дату
```

Крок 3.

Було прийнято рішення розмістити свій код у асинхронній функції, так можна не чекати обробки всіх запитів, відображати сторінку далі.

Крок 4.

```
function renderRate(rate) {  
  const li = document.createElement('li');  
  li.innerHTML = `<span class="" onclick="getRateHistory('${rate.cc}')">${rate.txt} - ${rate.rate}</span>`;   
  return li;  
}
```

Було обрано варіант підключення обробника подій через код HTML. addEventListener не використовувався, тому що потрібно було б звертатися до event.target, щоб можна було отримати дані. А так одразу викликається функція і передаються аргументи. Але інші методи теж непрограють цьому.

Крок 5.

<https://bank.gov.ua/NBUStatService/v1/statdirectory/exchange?valcode=USD&date=20070527&json>

Структура відповіді аналогічна структурі з [кроку 1](#).

Курс Євро на позавчора:

```
> let dateNow1 = new Date();  
    dateNow1.setDate(dateNow1.getDate() - 2);  
    dateNow1 = dateNow1.toLocaleDateString();  
    dateNow1 = dateNow1.split("/").reverse().join("");  
  
    fetch(`https://bank.gov.ua/NBUStatService/v1/statdirectory/exchange?valcode=EUR&date=${dateNow1}&json`)  
      .then(result => result.json())  
      .then(obj => console.log(obj));  
  
◀ ▶ Promise {<pending>}  
  
▼ [{...}] ⓘ  
  ▶ 0: {r030: 978, txt: "Євро", rate: 33.5693, cc: "EUR", exchangedate: "28.05.2021"}  
    length: 1  
    __proto__: Array(0)
```

Крок 6.

```
async function getRate(url) {  
  const res = await fetch(url);  
  const json = await res.json();  
  return json;  
}
```

Оголошення **async function** визначає асинхронну функцію, яка повертає об'єкт **AsyncFunction**. Ви також можете визначити асинхронні функції, використовуючи вираз **async function**. Оператор **await** використовується для очікування закінчення **Promise**. Може бути використаний лише всередині **async function**.

Крок 7.

```
function getRateHistory(cc) {
  rateHistoryView.innerHTML = "";
  rateHistoryView.innerHTML += "Курс валют за останні 7 днів : "
  const dates = getDateList();
  Promise.all(
    dates.map(date => getRate(`https://bank.gov.ua/NBUStatService/v1/statdirectory/exchange?valcode=${cc}&date=${date}&json`))
  )
  .then(
    date => date.map( day => renderRateHistory(day[0]) )
  );
}

function renderRateHistory(rate) {
  rateHistoryView.innerHTML += `<li>${rate.exchangedate} - ${rate.txt} - ${rate.cc} - ${rate.rate}</li>`;
}
```

Крок 8.

```
function getRateHistory(cc) {
  rateHistoryView.innerHTML = "";
  rateHistoryView.innerHTML += "Курс валют за останні 7 днів : "
  const dates = getDateList();
  Promise.all(
    dates.map(date => getRate(`https://bank.gov.ua/NBUStatService/v1/statdirectory/exchange?valcode=${cc}&date=${date}&json`))
  )
  .then(
    date => date.map( day => renderRateHistory(day[0]) )
  );
}

function renderRateHistory(rate) {
  rateHistoryView.innerHTML += `<li>${rate.exchangedate} - ${rate.txt} - ${rate.cc} - ${rate.rate}</li>`;
}

async function getRate(url) {
  const res = await fetch(url);
  const json = await res.json();
  return json;
}

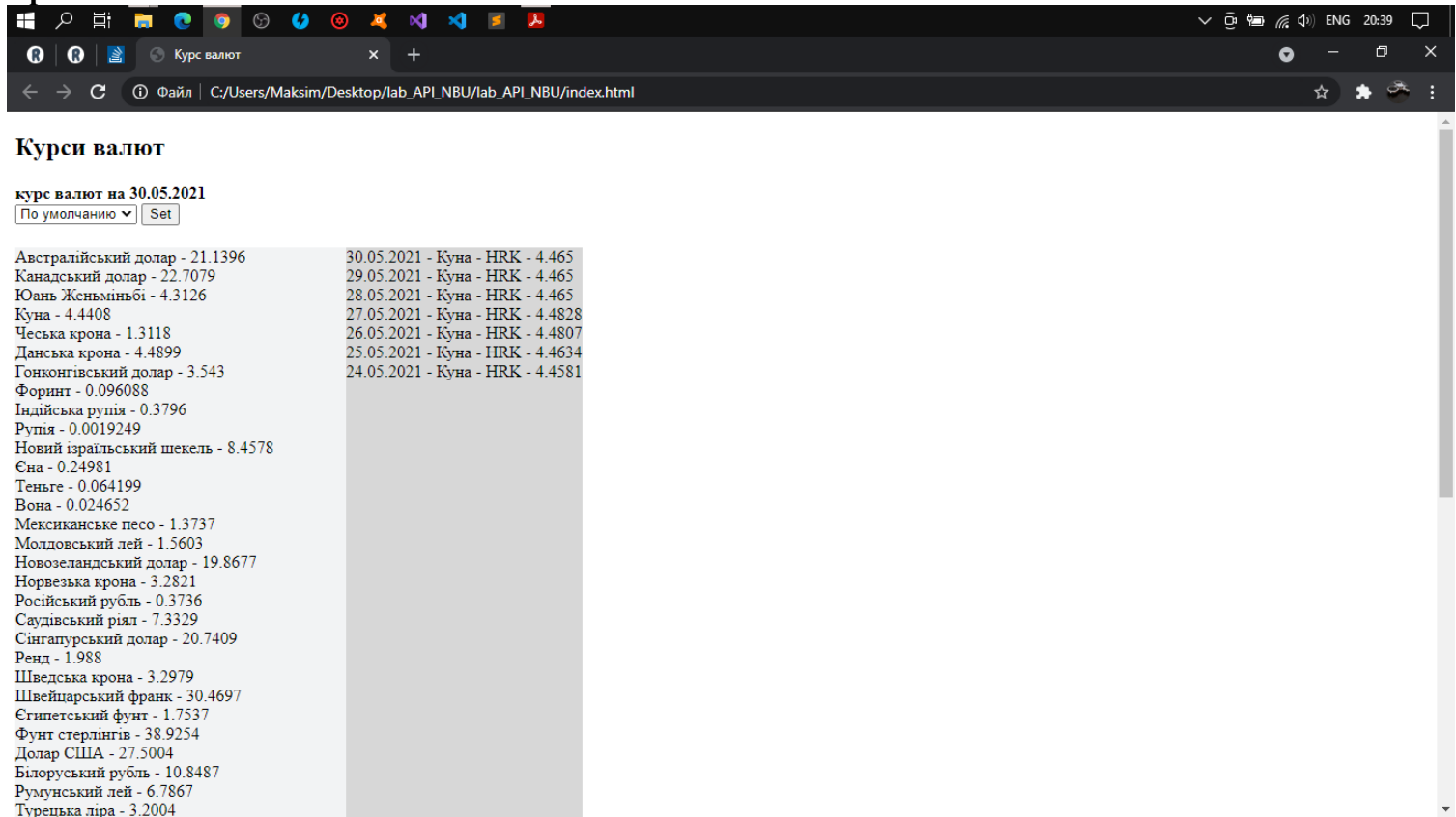
function getDateList() {
  const dayMilliseconds = 24 * 60 * 60 * 1000;
  let currentDate = new Date();

  let dateList = [];

  for (let i = 0; i < within; ++i) {
    dateList.push(currentDate.toLocaleDateString().split('.').reverse().join(''));
    currentDate.setTime(currentDate.getTime() - dayMilliseconds);
  }

  return dateList;
}
```

Крок 9 .



Валюта	Курс
Австралійський долар	21.1396
Канадський долар	22.7079
Юань Женьміньбі	4.3126
Куна	4.4408
Чеська крона	1.3118
Данська крона	4.4899
Гонконгівський долар	3.543
Форинт	0.096088
Індійська рупія	0.3796
Рупія	0.0019249
Новий ізраїльський шекель	8.4578
Єна	0.24981
Тенге	0.064199
Вона	0.024652
Мексиканське песо	1.3737
Молдовський лей	1.5603
Новозеландський долар	19.8677
Норвезька крона	3.2821
Російський рубль	0.3736
Саудівський ріял	7.3329
Сінгапурський долар	20.7409
Ренд	1.988
Шведська крона	3.2979
Швейцарський франк	30.4697
Єгипетський фунт	1.7537
Фунт стерлінгів	38.9254
Долар США	27.5004
Білоруський рубль	10.8487
Румунський лей	6.7867
Турецька ліра	3.2004

Крок 10-11.

```
function getDateList() {
  const dayMilliseconds = 24 * 60 * 60 * 1000;
  let currentDate = new Date();

  let dateList = [];

  for (let i = 0; i < within; ++i) {
    dateList.push(currentDate.toLocaleDateString().split('.').reverse().join(''));
    currentDate.setTime(currentDate.getTime() - dayMilliseconds);
  }

  return dateList;
}
```

Висновок: були поглиблені та закріплені знання з асинхронних функцій. Були придбані практичні навички роботи з Fetch API. Навчилися отримувати дані з зовнішніх джерел.