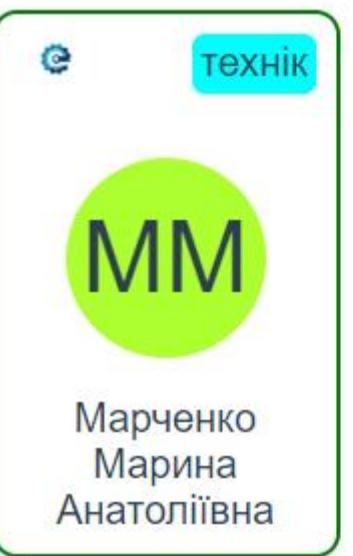
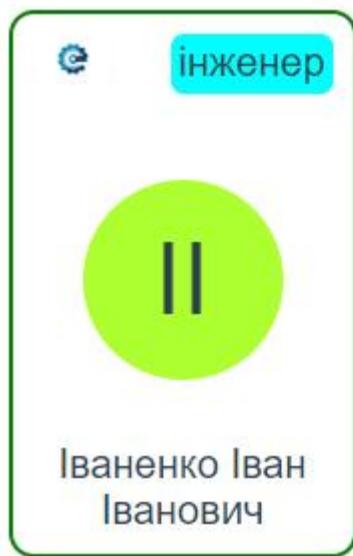


Vue. Slot

<https://vuejs.org/guide/components/slots.html#slots>

Для чого потрібні слоти? Чим відрізняються картки?

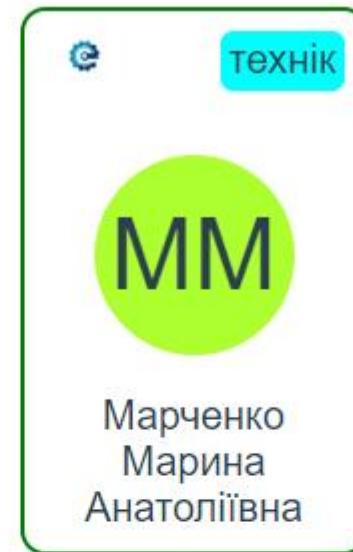


```
employees: [  
  {  
    id: 1,  
    name: 'Іваненко Іван Іванович',  
    position: 'інженер',  
    logo: 'https://s.timgcdn.com/sablon-vektor-tehniceskogo-obslu'...  
  },  
  ...  
]
```

```
products: [  
  {  
    id: 1,  
    title: 'Смартфон XYZ',  
    action: 'знижка',  
    imgSrc: 'https://cdn.tmo...  
    logo: 'https://img.freep...  
    price: 100,  
  },  
  ...  
]
```

Для чого потрібні слоти? Чим відрізняються картки?

Відображаємо ініціали

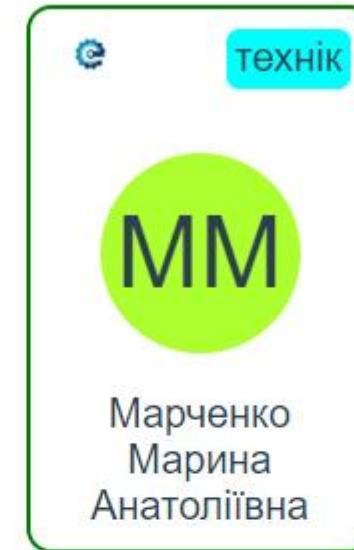
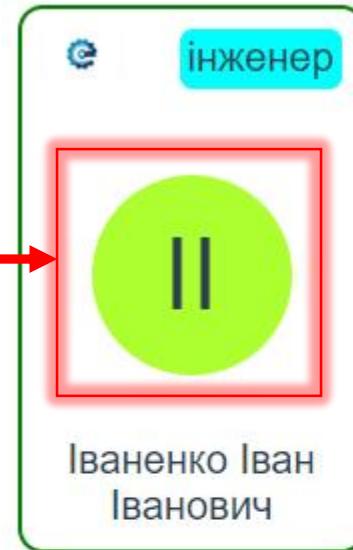


Відображаємо фото



Для чого потрібні слоти? Чим відрізняються картки?

Відображаємо ініціали



Секції, які відрізняються
описуємо з використанням
слотів

Відображаємо фото



Схема використання слотів

Дочірній компонент	Батьківський компонент
<pre><template> фіксована частина <slot> ... змінювана частина ...</slot> фіксована частина </template></pre>	<pre><template> <дочірній-компонент> ... контент для вставки1 ... </дочірній-компонент> </template></pre>

Дочірній компонент	Батьківський компонент
<pre><template> фіксована частина ... <slot> ... змінювана частина фіксована частина ... </template></pre>	<pre><template> <!-- дочірній-компонент --> ... контент для вставки ... </дочірній-компонент> </template></pre>

Дочірній компонент

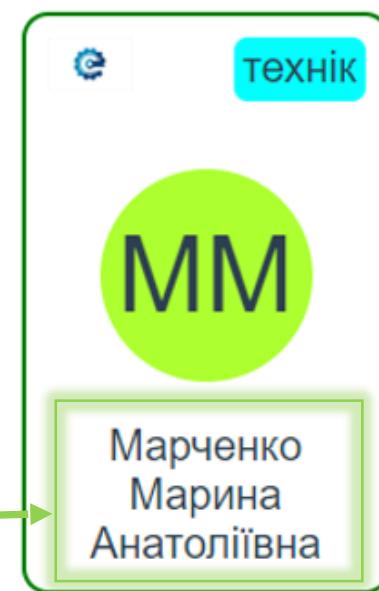
▼ DataCard.vue components u x

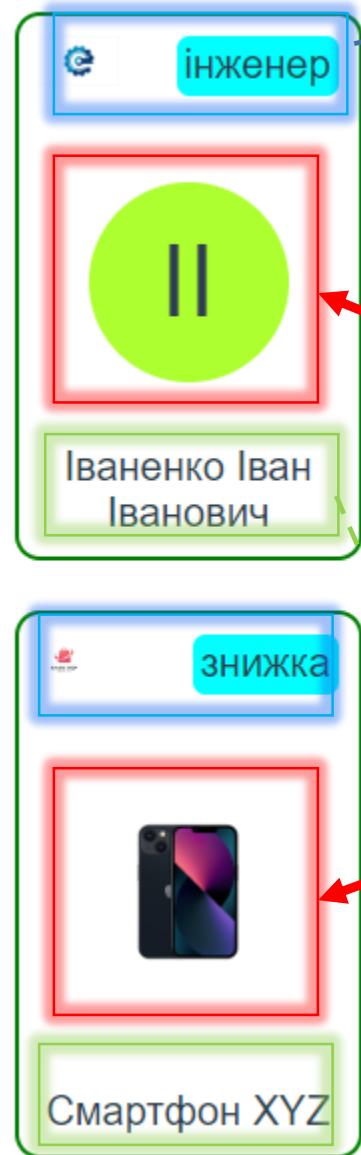
```
src > components > ▼ DataCard.vue > {} "DataCard.vue" > script > default
1  <template>
2    <div class="card">
3      <div class="header">...
4      <div class="main">
5        <slot>
6          
7        </slot>
8      </div>
9      <div class="footer">...
10     </div>
11   </div>
12 > <div class="container">
13   <data-card
14     v-for="employee in getEmployeesList"
15     :key="employee.id"
16     :info="{
17       logo: employee.logo,
18       label: employee.position,
19       message: employee.name,
20     }"
21   </data-card>
22 > </div>
23 > </template>
```

Батьківський компонент

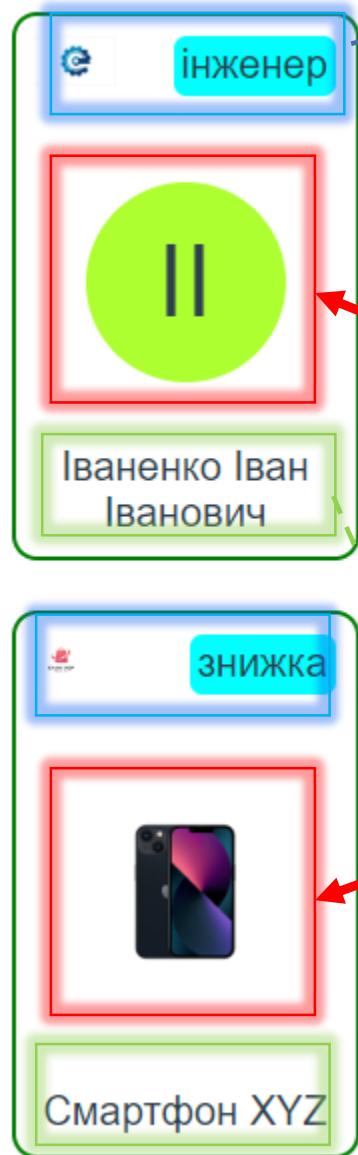
▼ PersonList.vue components u x

```
src > components > ▼ PersonList.vue > {} "PersonList.vue" > template
1  <template>
2    <div class="container">
3      <data-card
4        v-for="employee in getEmployeesList"
5        :key="employee.id"
6        :info="{
7          logo: employee.logo,
8          label: employee.position,
9          message: employee.name,
10         }"
11      <div class="initials">
12        <div>
13          {{ getInitials(employee) }}
14        </div>
15      </div>
16    </data-card>
17  </div>
18 > </template>
```





```
<template>
  <div class="card">
    <div class="header">
      
      <div>
        {{ info.label }}
      </div>
    </div>
    <div class="main">
      <slot>
        ...
      </slot>
    </div>
    <div class="footer">
      {{ info.message }}
    </div>
  </div>
</template>
```



```
<template>
  <div class="card">
    <div class="header">
      
      <div>
        {{ info.label }}
      </div>
    </div>
    <div class="main">
      <slot>
        Сюди будемо вставляти потрібний контен
      </slot>
    </div>
    <div class="footer">
      {{ info.message }}
    </div>
  </div>
</template>
```



```
<template>
  <div class="card">
    <div class="header">
      
      <div>
        {{ info.label }}
      </div>
    </div>
    <div class="main">
      <slot>
        
      </slot>
    </div>
    <div class="footer">
      {{ info.message }}
    </div>
  </div>
</template>
```

Значення за замовчуванням – це значення, яке найчастіше використовується

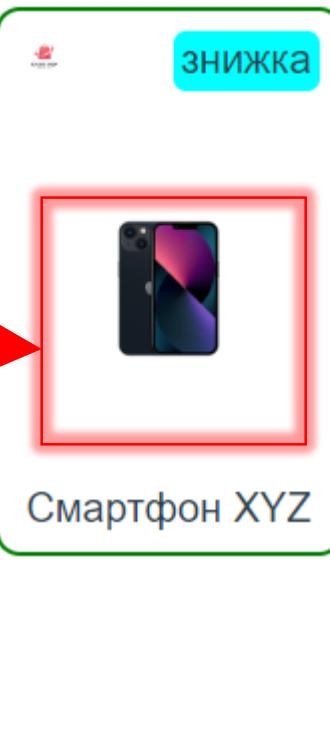
ProductsList.vue

```
<template>
  <div class="container">
    <data-card
      v-for="product in getProductsList"
      :key="product.id"
      :info="{
        logo: product.logo,
        label: product.action,
        image: product.imgSrc,
        message: product.title,
      }"
    >
      </data-card>
    </div>
  </template>
```

Використовуємо
значення слота
за замовчуванням

DataCard.vue

```
<template>
  <div class="card">
    <div class="header">
      
    </div>
    <div>
      {{ info.label }}
    </div>
    <div class="main">
      <slot>
        
      </slot>
    </div>
    <div class="footer">
      {{ info.message }}
    </div>
  </template>
```



EmployeesList.vue

```
<template>
  <div class="container">
    <data-card
      v-for="employee in getEmployeesList"
      :key="employee.id"
      :info="{
        logo: employee.logo,
        label: employee.position,
        message: employee.name,
      }"
    >
      <div class="initials">
        <div>
          {{ getInitials(employee) }}
        </div>
      </div>
    </data-card>
  </div>
</template>
```

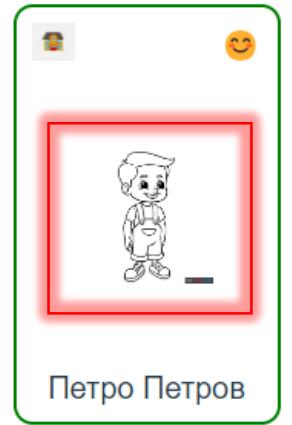
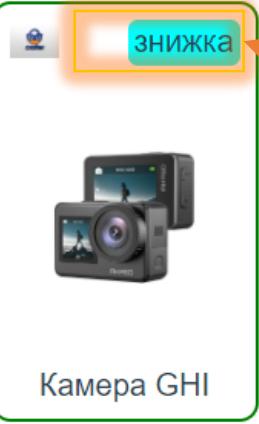
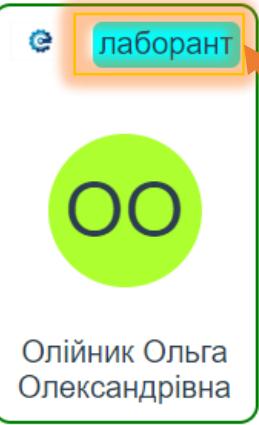
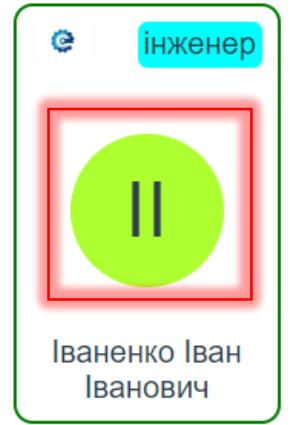
DataCard.vue

```
<template>
  <div class="card">
    <div class="header">
      
    </div>
    <div>
      {{ info.label }}
    </div>
    <div class="main">
      <slot>
        <img alt="info.image || defaultImageSrc" />
      </slot>
    </div>
    <div class="footer">
      {{ info.message }}
    </div>
  </div>
</template>
```

Значення за
замовчуванням
заміняємо іншим
вмістом



Для чого потрібні слоти? Чим відрізняються картки?



Відрізняються також
ци елементи

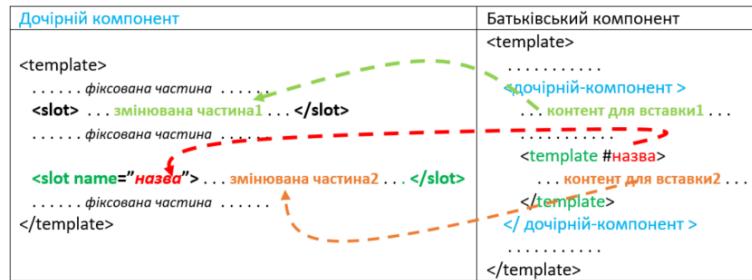
Якщо є декілька змінюваних частин, які хочемо змінювати ззовні, то потрібно також використовувати іменовані слоти

Схема використання іменованих слотів `<slot name="назва"> ... </slot>`

Дочірній компонент	Батьківський компонент
<pre><template> фіксована частина ... <slot> ... змінювана частина1 ... </slot> фіксована частина ... <slot name="назва"> ... змінювана частина2 ... </slot> фіксована частина ... </template></pre>	<pre><template> <дочірній-компонент> <slot> ... контент для вставки1 ... </slot> <slot name="#назва"> ... контент для вставки2 ... </slot> </template> </дочірній-компонент> </template></pre>

Якщо є декілька змінюваних частин, які хочемо змінювати ззовні, то потрібно також використовувати іменовані слоти

Схема використання іменованих слотів <slot name="назва"> ... </slot>



Дочірній компонент

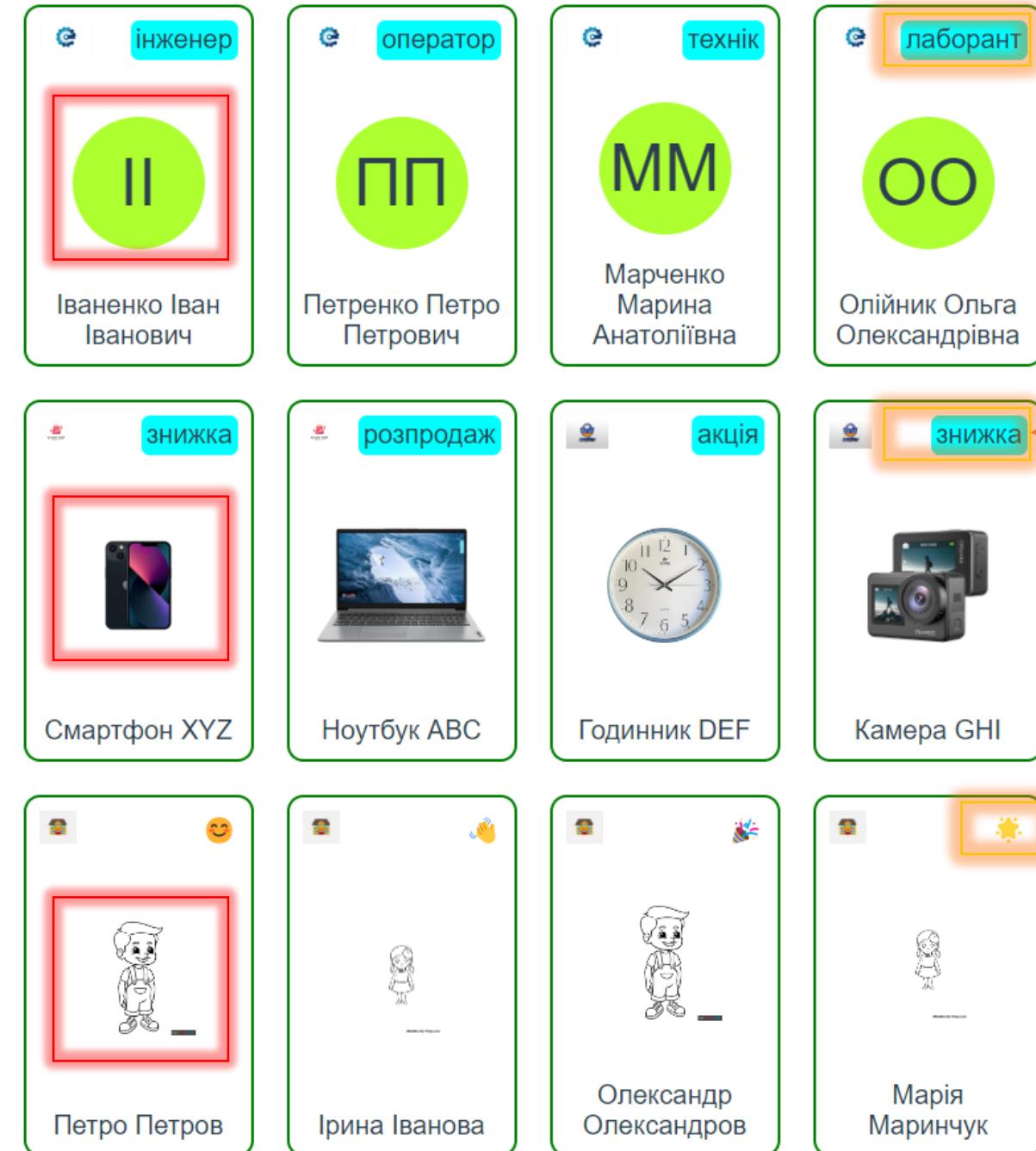
```
▼ DataCard.vue components U X
src > components > ▼ DataCard.vue > {} "DataCard.vue" > ⚡ script
1  <template>
2    <div class="card">
3      <div class="header">
4        
5        <slot name="right-header">
6          <div>
7            {{ info.label }}
8          </div>
9        </slot>
10       </div>
11       <div class="main">
12         <slot>
13           
14         </slot>
15       </div>
16     <div class="footer"> ...
17   </div>
18   </div>
19 </template>
```

Батьківський компонент

```
▼ FriendList.vue components U X
src > components > ▼ FriendList.vue > {} "FriendList.vue" > ⚡ template
1   <template>
2     <div class="container">
3       <data-card
4         v-for="friend in getFriendsList"
5           :key="friend.id"
6           :info="{
7             logo: friend.logo,
8             label: friend.position,
9             message: friend.name,
10            image: friend.imgSrc,
11          }"
12        >
13          <template #right-header>
14            {{ friend.status }}
15          </template>
16        </data-card>
17     </div>
18   </template>
```



Олександр
Олександров



```
File: DataCard.vue components U X
c > components > ▼ DataCard.vue > {} "DataCard.vue" > ⚡
1   <template>
2     <div class="card">
3       <div class="header">
4         
5
6         <slot name="right-header">
7           <div>
8             {{ info.label }} <!-->
9           </div>
10        </slot>
11      </div>
12      <div class="main">
13        <slot>
14          
16        </slot>
17      </div>
18      <div class="footer">
19        {{ info.message }} <!-->
20      </div>
21    </template>
```

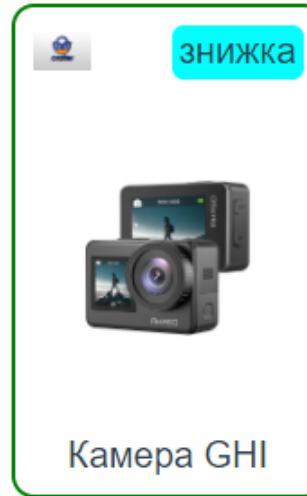
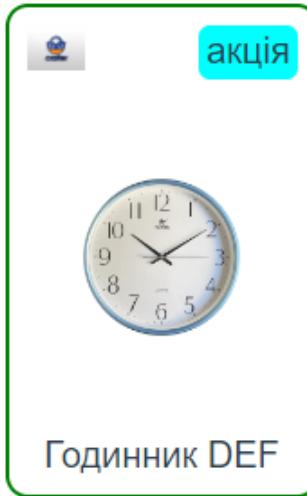
```
<template>
  <div class="container">
    <data-card
      v-for="friend in getFriendsList"
      :key="friend.id"
      :info="{
        logo: friend.logo,
        label: friend.position,
        message: friend.name,
        image: friend.imgSrc,
      }"
    >
      <template #right-header>
        {{ friend.status }}
      </template>
    </data-card>
  </div>
</template>
```

DataCard.vue components U X

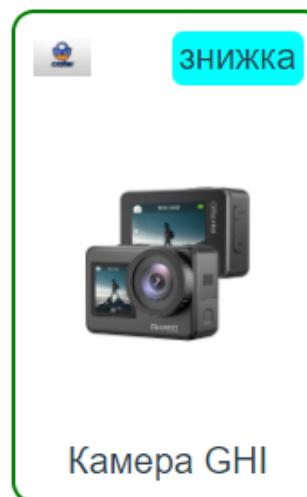
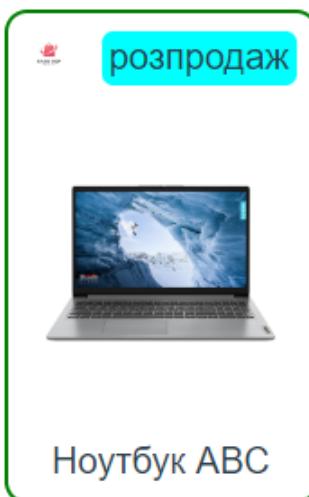
c > components > DataCard.vue > {} "DataCard.vue" > ↗

```
1   <template>
2     <div class="card">
3       <div class="header">
4         
5
6         <slot name="right-header">
7           <div>
8             {{ info.label }}
9           </div>
10          </slot>
11
12        <div class="main">
13          <slot>
14            
16          </slot>
17        </div>
18        <div class="footer">
19          {{ info.message }}
20        </div>
21      </template>
```

Що робити, якщо батьківський хоче отримати доступ до даних, які у дочірньому? Можливо для виконання додаткових обчислень.



Total : 8645



Min price : 100
Max price : 5000

Схема передачі даних через слоти `<slot name="назва" :властивість = "значення"> ... </slot>`

Дочірній компонент	Батьківський компонент
<pre><template> фіксована частина <slot :властивість = "значення1"> ... змінювана частина1 ... </slot> фіксована частина <slot name="назва" :властивість = "значення2"> ... змінювана частина2 ... </slot> фіксована частина </template></pre>	<pre><template> <дочірній-компонент > <template #default="{значення1}"> ... контент з використанням значення1... </template> <template #назва="значення2"> ... контент з використанням значення2... </template> </дочірній-компонент > </template></pre>

Дочірній компонент

```
ProductList.vue components U X
src > components > ProductList.vue > {} "ProductList.vue" > template >
1   <template>
2     <div class="container">
3       <data-card
4         v-for="product in getProductsList"
5         :key="product.id"
6         :info="{
7           logo: product.logo,
8           label: product.action,
9           image: product.imgSrc,
10          message: product.title,
11        }"
12      >
13        </data-card>
14      </div>
15      <div>
16        <slot :product-list="getProductsList"></slot>
17      </div>
18    </template>
```

Батьківський компонент

```
HomeView.vue views 2, M
src > views > HomeView.vue > {} "HomeView.vue" > template > ...
1   <template>
2     <div class="home">
3       <product-list>
4         <template #default="{ productList }">
5           <hr />
6           <div>
7             Total : {{ getTotalPrice
8               (productList) }}
9           </div>
10          </template>
11        </product-list>
12        <product-list>
13          <template #default="{ productList }">
14            <hr />
15            <div>
16              Min price : {{ getMinPrice
17                (productList) }}
18            </div>
19            <div>
20              Max price : {{ getMaxPrice
21                (productList) }}
22            </div>
23          </template>
24        </product-list>
25      </div>
26    </template>
```

Що робити, якщо деякі сторінки мають інший шаблон?

Шаблон для усіх сторінок окрім авторизації

Головна Продукти Менеджери Про компанію Контакти

Контент сторінки

Власник : Баба Галя Тел. 0500000001
Адреса: Усі знають де живе баба Галя

Шаблон сторінки авторизації, та зміни пароля

На головну

Контент сторінки

Ви знаходитесь на сторінці адміністрування.
Увійдіть, щоб користуватись сервісами

Що робити, якщо деякі сторінки мають інший шаблон?

Шаблон для усіх сторінок окрім авторизації

Головна Продукти Менеджери Про компанію Контакти

Контент сторінки

Власник : Баба Галя Тел. 0500000001
Адреса: Усі знають де живе баба Галя

Шаблон сторінки авторизації, та зміни пароля

На головну

Контент сторінки

Ви знаходитесь на сторінці адміністрування.
Увійдіть, щоб користуватись сервісами

Потрібно розробити masterpage-шаблони з слотами для **контенту** і розробляти сторінки на основі цих шаблонів

masterpage-шаблони :

- це звичайні компоненти, які використовуються як основа для групи інших компонентів
- можуть містити спільні для відповідної групи компонентів елементи (наприклад, навігаційне меню, footer, ...)
- містять слоти, для вставки специфічного для кожного компонента **контенту**