

Завдання:

До результату виконання роботи, де було створено веб-сторінку для музичної групи (CSS_background) створити інтерактивну карту з відмітками, в яких містах музична група планує проводити концерт. При наведенні курсора на точку міста буде з'являтися виноска з датою заходу і кнопкою «Купити квитки»:



Завантаження файлів

Вам знадобляться певні файли для виконання завдання. Завантажте архів «css3-transitions-animations-practice.rar» на комп'ютер і ознайомтеся з його вмістом. Щоб не витратити час на написання коду, який не відноситься безпосередньо до теми практичної роботи, в архіві є HTML-розмітка і базові стилі CSS. Вам знадобиться додати тільки ті стилі, які будуть стосуватися анімованих об'єктів.

План роботи

1. Для чотирьох точок на карті (міста, в яких будуть проходити концерти) створити пульсуючу анімацію для залучення уваги.

2. При наведенні курсора на точку міста анімація повинна ставати на паузу. Коли курсор прибралий, анімація поновлюється.
3. Також при наведенні курсору на точку міста повинна плавно з'являтися невелика виноска з місцем та датою концерту, кнопкою для покупки квитків. Коли курсор прибралий, інформаційна виноска зникає.
4. Колір кнопки в виносці повинен змінюватися при наведенні курсору.
5. Додаткове завдання.

1. Анімація для відміток на карті

Відкрийте в браузері веб-сторінку з завантаженого архіву. Перед вами блок з картою, на якій нанесено чотири позначки з назвами міст. Завдання - створити ефект пульсуючих хвиль навколо кожної точки. Зробимо дві хвилі, в ролі яких виступатимуть псевдоелементи **:before** та **:after**:



Псевдоелементи зручно використовувати, коли потрібно додати якесь оформлення до основного елементу. Це позбавляє від необхідності додавати зайвий HTML-код, який не несе особливого сенсу.

Відкрийте в редакторі коду документ `style.css` з папки `css`. Потрібно дотримуватися порядку в таблиці стилів, додаючи нові правила не просто в кінець документа, а розміщуючи їх там, де це буде логічно. Тому знайдіть селектор **.pin .popover** і додайте над ним наступні стилі для псевдоелементів елемента **.pin**:

```
.pin:before,  
.pin:after {  
    content: '';  
    position: absolute;  
    left: 50%;  
    top: 50%;  
    display: block;  
    border-radius: 50%;  
    border: 1px solid #f3244a;  
    width: 0;  
    height: 0;  
    margin-left: -2px;  
    margin-top: -2px;  
}
```

Не додається ніякий контент до цих псевдоелементів, а тільки створити порожній блок з рамкою і округленими кутами, щоб в результаті отримати кільце.

У початковому вигляді ширина і висота псевдоелементів дорівнює нулю. І за допомогою анімації ми збільшимо їх розміри, щоб створити ефект пульсуючих хвиль. Для цього потрібно буде зробити ключові кадри для кожної хвилі.

Знайдіть на початку таблиці стилів коментар `/* Keyframes */` і запишіть під ним два наступних правила `@keyframes`:

```
@keyframes pinBeforeWave {
  from {
    width: 0;
    height: 0;
    margin-left: -2px;
    margin-top: -2px;
  }
  to {
    width: 40px;
    height: 40px;
    margin-left: -21px;
    margin-top: -21px;
    opacity: 0;
  }
}

@keyframes pinAfterWave {
  from {
    width: 0;
    height: 0;
    margin-left: -2px;
    margin-top: -2px;
  }
  to {
    width: 66px;
    height: 66px;
    margin-left: -34px;
    margin-top: -34px;
    opacity: 0;
  }
}
```

Обидві анімації мають однаковий початок, але відрізняються закінченням. Анімацію **pinBeforeWave** ми застосуємо до псевдоелементу **:before**, і в процесі її виконання він збільшиться до розмірів 40×40 пікселів (маленька хвиля). Анімація **pinAfterWave** призначена для псевдоелемента **:after**, до кінця якої він прийме розміри 66×66 пікселів (велика хвиля).

Крім цього, в обох анімаціях передбачено зміщення елементів на певну кількість пікселів вліво і вгору - це потрібно для того, щоб при збільшенні кільця відмітка міста візуально залишалася в його центрі. І, нарешті, властивість **opacity** в останньому ключовому кадрі означає те, що обидві хвилі будуть плавно зникати в процесі свого розширення.

Застосуйте створені анімації до псевдоелементів. Для цього додайте наступний код, розмістивши його під стилем для селектора **.pin:before**, **.pin:after**:

```
.pin:before {  
    animation: pinBeforeWave 1s ease-in infinite;  
}  
.pin:after {  
    animation: pinAfterWave 1s ease-in infinite;  
}
```

Обидві анімації триватимуть одну секунду. Розподіл швидкості анімації протягом цієї секунди відбуватиметься по функції **ease-in**. Анімація буде повторюватися нескінченно.

Збережіть зміни в таблиці стилів і оновіть сторінку в браузері. Анімація навколо відміток вже працює, і нам час переходити до наступного пункту плану.

2. Пауза анімації при наведенні курсору

Коли користувач наводить курсор на елемент **.pin**, анімація псевдоелементів повинна ставати на паузу. CSS дозволяє застосовувати стилі до дочірніх елементів, коли на їх батьків наведено курсор. Те ж саме працює і з псевдоелементами. Додайте цей стиль під попереднім:

```
.pin:hover:before,  
.pin:hover:after {  
    animation-play-state: paused;  
}
```

Даний код каже: коли на елемент **.pin** наведений курсор (стан **:hover**), потрібно поставити на паузу анімацію псевдоелементів **:before** та **:after**.

3. Поява виноски

При наведенні курсора на позначку має відбуватися ще одна дія, а саме поява виноска з інформацією про концерт. У початковому стані всі виноски приховані від очей користувача за допомогою властивостей **visibility: hidden** і **opacity: 0**.

Знайдіть у файлі CSS селектор **.pin .popover:before** і додайте після нього наступний стиль:

```
.pin:hover .popover {  
    visibility: visible;  
    opacity: 1;  
}
```

У вас могло виникнути питання, навіщо ми приховали виноску відразу двома способами. Справа в тому, що використання лише однієї з цих властивостей буде недостатньо. Якщо прибрати властивість `visibility` і залишити тільки `opacity: 0`, то елемент не буде по-справжньому прихований: так, його не буде видно, але його побачить скрінрідер, що іноді може заважати.

Якщо ж прибрати властивість прозорості і залишити тільки `visibility: hidden`, то ми не зможемо добитися плавності при появі елемента, оскільки властивість `transition` не діє на властивість `visibility`. Саме тому потрібно використовувати відразу дві властивості для приховування виноски.

Якщо зараз оновити сторінку і навести курсор на точку, то побачите, що виноска з'являється, але робить це різко. Настав час звернутися до властивості `transition`. Знайдіть селектор **.pin .popover** і додайте до нього такий рядок:

transition: all 0.2s ease-in-out;

Тепер виноска з'являється плавно: властивість `opacity` переходить від значення 0 до значення 1 за 200 мілісекунд, а швидкість руху анімації відповідає функції `ease-in-out`.

Застосуйте ефект для виноски, щоб вона зникла з легкою затримкою. Для цього додайте ще одне значення до властивості `transition` для селектора **.pin .popover** - 0.5s перед крапкою з комою:

transition: all 0.2s ease-in-out 0.5s;

Зараз затримка появи виноски спрацьовує в обох напрямках - коли курсор наводиться і коли відводиться. Щоб залишити затримку тільки для другого випадку, допишіть наступний рядок до селектору **.pin:hover .popover**:

transition-delay: 0s;

Збережіть зміни і оновіть сторінку в браузері. Тепер все працює правильно: виноска з'являється плавно і без затримки, а зникає так само плавно, але з затримкою в півсекунди.

Використаємо властивість `transform` і її функцію обертання `rotate`, щоб додати ще один візуальний ефект при появі виноски. Вона буде розгортатися в процесі появи, а потім так само зникати.

У стані невидимості виноска буде розгорнута на 90 градусів по осі Y. Допишіть наступний стиль до селектору **.pin .popover**:

```
transform: rotateY (90deg);
```

При наведенні курсора на точку міста виноска повинна повертатися в своє нормальне положення. Для цього додайте рядок нижче до стилю для селектора **.pin:hover .popover**:

```
transform: rotateY (0deg);
```

Збережіть код і оновіть веб-сторінку. Поспостерігайте тепер за поведінкою виносок: вони з'являються плавно і з ефектом розвороту, нагадуючи картки.

4. Колір кнопки у виносці

Останній пункт плану - плавна зміна кольору кнопки в виносці при наведенні курсору на неї. Для початку створимо стиль для кнопки в стані **:hover**. Знайдіть селектор **.pin .popover .button** і запишіть під ним наступний код:

```
.pin .popover .button:hover {  
    background-color: #e10087;  
}
```

І, нарешті, щоб перехід від початкового кольору до кінцевого відбувався плавно, додайте цей рядок до стилю селектора **.pin .popover .button**:

```
transition: all 0.2s ease-in-out;
```

Знову збережіть зміни і перевірте результат.

Додаткове завдання:

До створеної сторінки додайте форму замовлення квитків, яка буде з'являтися при кліку на кнопку "BUY TICKETS". Рекомендовані елементи форми:

- Name
- Phone number
- E-mail
- Submit

Ці елементи є обов'язковими. Додайте інші елементи, які вважаєте потрібними для цієї форми замовлення. Не забувайте правильно вказувати типи полів.