

Flip your card

Introduction

What you will make:

In this project you will make a card that flips with content on the front and back.



Flip cards help people to interact with a webpage. Flip cards also allow you to provide extra information in a small space or hide information that shouldn't be immediately visible. Some common examples of animated flip cards are revision cards, business cards, social media contact cards, and product cards.

You will:

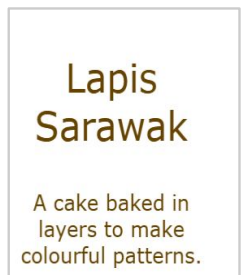
- Use CSS animations to create a **card that flips over** when it is interacted with
- Apply a **gradient** background using CSS
- Add and use a font from **Google Fonts**

When you will hover your mouse on the front of the flip card it will automatically turn and show the back of flip card.

Front of
flip card



Back of
flip card



1. Create your card

In this step you will create an infocard for a tasty treat with an image, heading, and text.

Step 1:

Open the starter file to start the project. /rpf.io/webcard

Step 2:

The `<title>` tag within your code is used to display text in the title bar of your web browser. This text should be relevant to your webpage.

You can find the `<title>` in the `<head>` element at the top of your `index.html` file.

Step 3:

Change your title to `Treat flip cards`.

index.html

```
1 <!DOCTYPE html>
2 <html lang="en">
3
4 <!-- This part is for extra information the browser needs to l
5 <head>
6
7   <meta charset="utf-8">
8
9   <!-- Don't shrink the page on mobile -->
10  <meta name="viewport" content="width=device-width, initial-s
11
12  <!-- Appears on the web browser tab and search results -->
13  <title>Treat flip cards</title>
```

Step 4:

Click on the small triangle next to the `<head>` tag to collapse the `<head>`.

Step 5:

Add a `<section>` to the `<main>` part of your page. Give the `<section>` a `wrap` class to make your webpage responsive.

index.html

```
26 | <body>
27 |
28 |     <main>
29 |         <section class="wrap">
30 |
31 |     </section>
```

Graphic designers are people who create engaging physical or digital visual content. Some examples of graphic design work include company logos, product packaging, and website banner design. Our graphic designer Kate has made some images of treats from around the world for you to use in your project.

The `<div>` element is used to group other elements. The `<div>` element is used when there isn't a more appropriate tag, such as `<section>` and `<blockquote>`.

Step 6:

Click a `<div>` that contains an image. The first image is `LapisSarawak02.png`, a sweet treat from Malaysia.



```
29 | <section class="wrap">
30 |     <div>
31 |         
32 |     </div>
33 | </section>
```

Step 7:

Add another `<div>` with heading and short paragraph about Lapis Sarawak:

- Heading: `Lapis Sarawak`
- Paragraph: `A cake baked in layers to make colourful patterns.`

```
<section class="wrap">
  <div>
    
  </div>
  <div>
    <h2>Lapis Sarawak</h2>
    <p>A cake baked in layers to make colourful patterns.</p>
  </div>
</section>
```

Step 8:

Test: Click the **Run** button.

View your webpage. You should be able to see an image of a Lapis Sarawak and your text that describes the image.

2. Flip your card

In this step you will make a card that flips with content on the front and back.

Use `<div></div>` tags to structure your code and separate the main parts of the card:

- The flip card
 - The viewable content
 - The front card face
 - The back card face

The `card` class has been created for you in `style.css`. The `card` class sets the width, height, and border of the card.

Step 1:

Add `<div></div>` tags around your current Lapis Sarawak code with the CSS class `card`.

```
<main>
  <section class="wrap">
    <div class="card">
      <div>
        
      </div>
      <div>
        <h2>Lapis Sarawak</h2>
        <p>A cake baked in layers to make
        </p>
      </div>
    </div>
  </section>
</main>
```

The `card-content` class has been created for you in `style.css`. The `card-content` class controls the timing, perspective, and 3D effect of the flip.

Step 2:

Add another set of `<div></div>` tags inside your card with the CSS class `card-content`.

```
<main>
  <section class="wrap">
    <div class="card">
      <div class="card-content">
        <div>
          
        </div>
        <div>
          <h2>Lapis Sarawak</h2>
          <p>A cake baked in layers to make
          </p>
        </div>
      </div>
    </div>
  </section>
</main>
```

Step 3:

Test: Click the **Run** button. Hover over (or tap) the card on your webpage. The content should flip over.

Step 4:

Add the `card-face` class to the `<div>` tag around the image. The `card-face` class tells the elements within it to hide when they are flipped to the back.

```
<main>
  <section class="wrap">
    <div class="card">
      <div class="card-content">
        <div class="card-face">
          
        </div>
        <div>
          <h2>Lapis Sarawak</h2>
          <p>A cake baked in layers to make
          </p>
        </div>
      </div>
    </div>
  </section>
</main>
```

Step 5:

Test: Click the **Run** button.

Hover over (or tap) the card on your webpage. The image and text appear on the front of the card and are both hidden when the card is flipped.

You need the text to appear on the back card face and not on the front card face. Earlier when the text did appear on the back, it was flipped so the text could only be read backwards from right to left. You can solve both of these things with one piece of code.

Step 6:

Add the **flipme** class to the **<div>** tag around the heading and paragraph. This rotates the text 180 degrees so it is flipped to the back face and now can be read correctly.

```
<main>
  <section class="wrap">
    <div class="card">
      <div class="card-content">
        <div class="card-face">
          
        <div class="card-face flipme">
          <h2>Lapis Sarawak</h2>
          <p>A cake baked in layers to make c
        </div>
      </div>
    </div>
  </section>
```

Step 8:

Choose: Go to the **style.css** file and change the setting in the **card-content** class to use the timing you like best.

Change the seconds in the **transition: transform 1s;** line of code.

Step 7:

Test: Click the **Run** button.

Hover over (or tap) the card on your webpage to see your flip card.

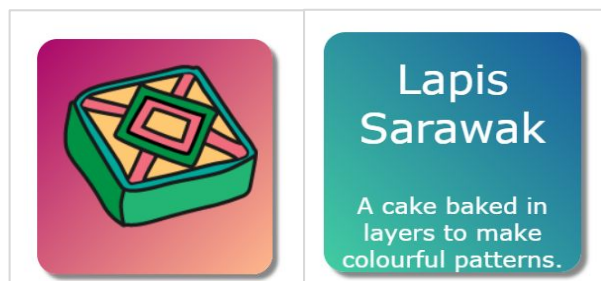


style.css

```
.card-content {
  position: relative;
  width: 100%;
  height: 100%;
  text-align: center;
  transition: transform 1s;
  transform-style: preserve-3d;
  perspective: 60rem;
}
```

3. Style your card

In this step you will add colour gradients and rounded corners to your card. You will also align the text so that it appears in the centre of the card.



Colour gradients are a gradual change from one colour to another. To process how to display a smooth colour gradient, computers work out the positions and colour shades needed to go from the starting colour through to the end colour.

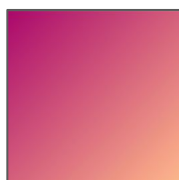


Gradients are used to make your card attract the viewer's attention.

Step 1:

Add the **gradient1** class to the **<div>** that displays the image.

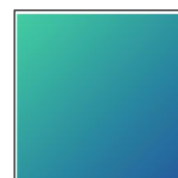
The **gradient1** class uses a linear gradient that transitions diagonally from one colour at the top to another colour at the bottom.



Step 2:

Add the **gradient2** class to the **<div>** that displays the heading and paragraph.

The **gradient2** class uses a linear gradient that transitions diagonally from one colour at the top to another colour at the bottom right.



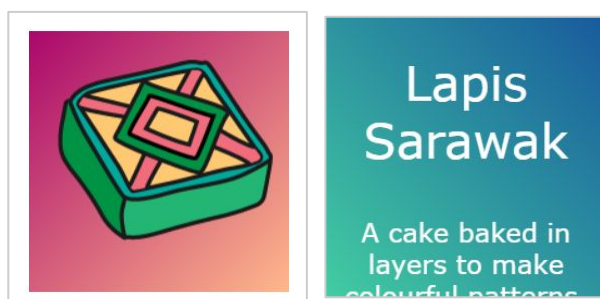
```
<div class="card">
  <div class="card-content">
    <div class="card-face gradient1">
      
    </div>
```

```
</div>
<div class="card-face flipme gradient2">
  <h2>Lapis Sarawak</h2>
  <p>A cake baked in layers to make col</p>
</div>
```

Step 3:

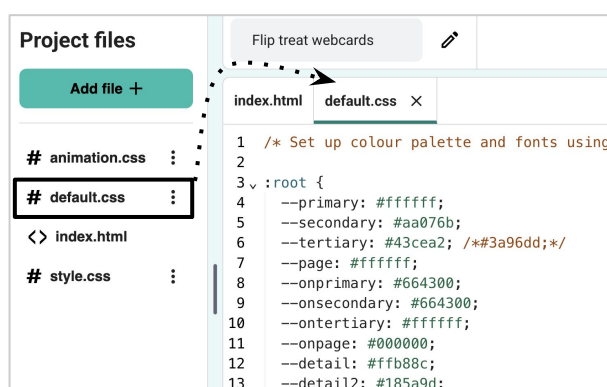
Test: Click the **Run** button.

Hover over (or tap) the card in your webpage to see that both gradients have been added.



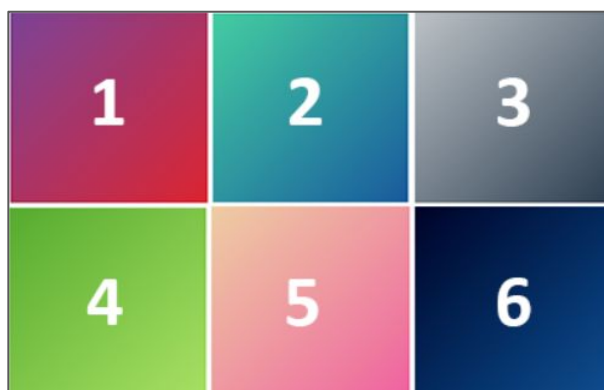
Step 4:

Click on the **default.css** file and find the colour palette near the top of the page.



Choose: Colour gradients use variables from the colour palette. You can choose the gradients you like the best.

1. #7b4397 → #dc2430
2. #43cea2 → #185a9d
3. #bdc3c7 → #2c3e50
4. #56ab2f → #a8e063
5. #eecda3 → #ef629f
6. #000428 → #004e92



Step 5:

Update the variables to see the front and back card faces in your new gradient:

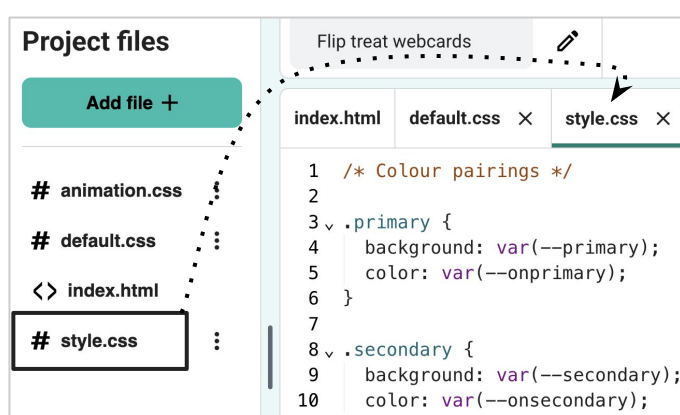
- The front card face uses **secondary** and **detail** variables
- The back card face uses **tertiary** and **detail2** variables

```
/* Set up colour palette and fonts using CSS variables

:root {
  --primary: #ffffff;
  --secondary: #aa076b;
  --tertiary: #43cea2;
  --page: #ffffff;
  --onprimary: #664300;
  --onsecondary: #664300;
  --ontertiary: #ffffff;
  --onpage: #000000;
  --detail: #ffb88c;
  --detail2: #185a9d;
}
```

Step 6:

Click on the **style.css** file and find the **gradient1** and **gradient2** classes.



Step 7:

Choose: Update the gradient to the direction you like best. Try **top**, **right**, or **bottom left**.

```
.gradient1 {
  background-image: linear-gradient(
    to bottom right,
    var(--secondary),
    var(--detail)
  );
  color: var(--onsecondary);
}

.gradient2 {
  background-image: linear-gradient(
    to bottom right,
    var(--tertiary),
    var(--detail2)
  );
  color: var(--ontertiary);
}
```

Step 7:

Click on `index.html`. Add the `rounded` class to both of your card faces to add rounded corners to your card.

```
<main>
  <section class="wrap">
    <div class="card">
      <div class="card-content">
        <div class="card-face gradient1 rounded">
          
        </div>
        <div class="card-face flipme gradient2 rounded">
          <h2>Lapis Sarawak</h2>
          <p>A cake baked in layers to make colourful patterns.</p>
        </div>
      </div>
    </div>
  </section>
</main>
```

The text on the card is not aligned correctly.



Step 8:

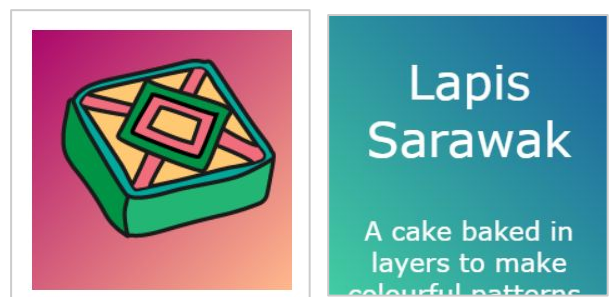
Add the `ycenter` class to align the text in the centre along the `y` axis.

```
<div class="card-face gradient1 rounded">
  
</div>
<div class="card-face flipme gradient2 rounded ycenter">
  <h2>Lapis Sarawak</h2>
  <p>A cake baked in layers to make colourful patterns.</p>
</div>
</div>
```

Step 9:

Test: Click the **Run** button.

Hover over (or tap) the card in your webpage to check that the text on the second face is aligned in the centre of the `y` axis.



Step 10:

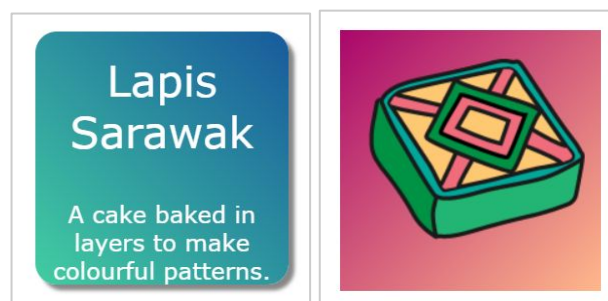
Use the shadow class to add a shadow effect to both sides of your card.

```
<div class="card-content">
  <div class="card-face gradient1 rounded shadow">
    
  </div>
  <div class="card-face flipme gradient2 rounded ycenter shadow">
    <h2>Lapis Sarawak</h2>
    <p>A cake baked in layers to make colourful patterns.</p>
  </div>
</div>
```

Step 11:

Test: Click the **Run** button.

Hover over (or tap) the card in your webpage to check that the card now has a shadow effect.



4. Choose your font

You will now use Google's online font library to import a decorative font into your webpage.

Web designers need to consider how a website looks and how it reflects the brand of the company, product, or person that they design the website for. Standard web safe fonts are recommended for the majority of text on a webpage. When a designer wants to create an impact, a more unique font can be used. Designers can import **web fonts** from an online font library to make their website attract people's attention.

Web designers carefully consider the font styles for their website. The **three** most common categories of font are:

- Library fonts
- Web safe fonts
- Fallback fonts

Library fonts are typically imported from a third-party library such as [Google Fonts](https://fonts.google.com/) (rpf.io/fonts). Companies sometimes pay a fee to use a font as part of their website branding.

Web safe fonts are standard fonts that should be available through any web browser. However, you can never be 100% sure that this is the case. Here is a list of web safe fonts:

- Arial
- Verdana
- Helvetica
- Tahoma
- Trebuchet MS
- Times New Roman
- Georgia
- Garamond
- Courier New
- Brush Script MT

Fallback fonts are generic font families that are used to match the styling that the web designer would like to use. The main font families are:

- Serif: a font style typically used in print publishing, letters have tiny decorative edges called ‘serifs’
- Sans-serif: a clean screen-readable font without the decorative edges
- Monospace: a font where each character uses the same width of space
- Cursive: a handwriting font
- Fantasy: a decorative font typically used for big headings

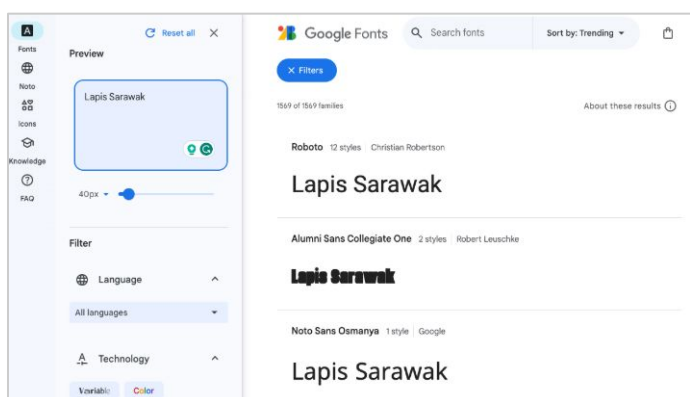
If a fallback font isn’t listed, then the web browser uses the browser’s default font, which is typically Times New Roman.

Step 1:

Type rpf.io/fonts in a new browser to open google fonts page.

Step 2:

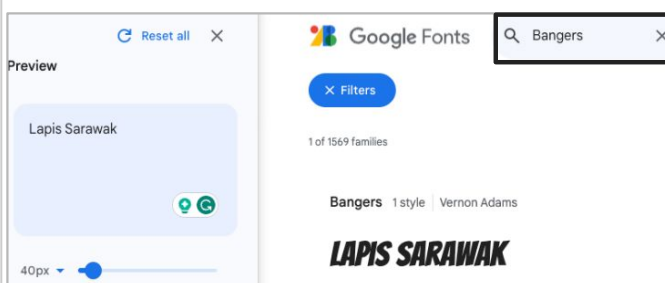
Type some sample text in the **Preview** box. The example uses the treat name in this project, **Lapis Sarawak**.



Step 3:

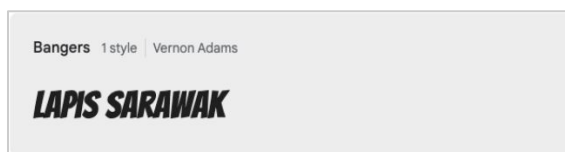
There are many different search filters to use. You can search by language, or different font properties.

Enter the font name **Bangers** into the ‘search fonts’ box.



Step 4:

Click on the Bangers card that has appeared in the search results.



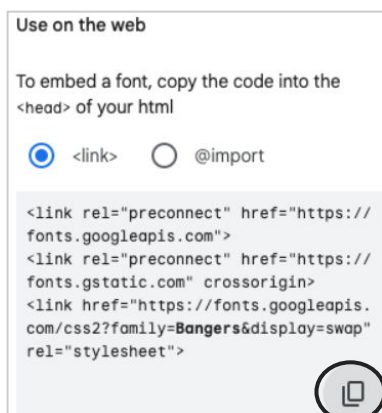
Step 5:

Click on the ‘Select’ link (our example shows ‘Select Regular 400’).



Step 5:

Click the copy icon to copy the HTML code.



Step 6:

Expand the **<head>** section of **index.html**, if it is collapsed.

Find the comment in your **index.html** document that says **<!-- Import fonts from Google -->**.

Step 7:

Paste the HTML code that you have just copied below the comment.

```
15 <!-- Import fonts from Google -->
16 <link rel="preconnect" href="https://fonts.googleapis.com">
17 <link rel="preconnect" href="https://fonts.gstatic.com" crossorigin>
18 <link href="https://fonts.googleapis.com/css2?family=Bangers&display=swap" rel="stylesheet">
```

You have now imported the fonts that you need from the Google Fonts library. Next, add in the correct CSS so that the web browser knows when to use this font.

Step 8:

Go to your **default.css** file and find the **header-font** and **title-font** variables. Replace the current font **Verdana** with **Bangers**.

```
--body-font: 1.1rem Verdana, sans-serif;
--header-font: lighter 3rem Bangers, cursive;
--title-font: lighter 2rem Bangers, cursive;
```

Step 9:

Test: Click the **Run** button.

Hover over (or tap) your flip card in the preview pane. You should see your new font applied to the heading of the back of the card.



Choose: To choose your own font, go back to rpf.io/fonts

Remove the Bangers style in Google Fonts before you select your new font.

Tip Don't forget to update your **default.css** file to include the new font name.