

1.

flexboxfroggy.com

☆

# FLEXBOX FROGGY

Level 1 of 24

Welcome to Flexbox Froggy, a game where you help Froggy and friends by writing CSS code! Guide this frog to the lilypad on the right by using the `justify-content` property, which aligns items horizontally and accepts the following values:

- `flex-start`: Items align to the left side of the container.
- `flex-end`: Items align to the right side of the container.
- `center`: Items align at the center of the container.
- `space-between`: Items display with equal spacing between them.
- `space-around`: Items display with equal spacing around them.

For example, `justify-content: flex-end;` will move the frog to the right.

```
1 #pond {
2   display: flex;
3   justify-content: flex-end;
4 }
5
6
7
8
9
10
```

Next

Flexbox Froggy is created by [Codepip](#) • [YouTube](#) • [Twitter](#) • [GitHub](#) • [Settings](#)

Want to learn CSS grid? Play [Grid Garden](#).

2.

←

→

↺

flexboxfroggy.com

☆

🔖

🎵

👤

⋮

# FLEXBOX FROGGY

◀ Level 2 of 24 ▶

Use `justify-content` again to help these frogs get to their lilypads. Remember that this CSS property aligns items horizontally and accepts the following values:



- `flex-start`: Items align to the left side of the container.
- `flex-end`: Items align to the right side of the container.
- `center`: Items align at the center of the container.
- `space-between`: Items display with equal spacing between them.
- `space-around`: Items display with equal spacing around them.

```
1 #pond {
2   display: flex;
3   justify-content:center;
4 }
5
6
7
8
9
10
```

Next

Flexbox Froggy is created by [Codepip](#) • [YouTube](#) • [Twitter](#) • [GitHub](#) • [Settings](#)

Want to learn CSS grid? Play [Grid Garden](#).



3.

flexboxfroggy.com

☆ | 📁 | 🎵 | 👤 | ⋮

# FLEXBOX FROGGY

◀ Level 3 of 24 ▶

Help all three frogs find their lilypads just by using `justify-content`. This time, the lilypads have lots of space all around them.




If you find yourself forgetting the possible values for a property, you can click on the property name to view them. Try clicking on `justify-content`.

```
1 #pond {  
2   display: flex;  
3   justify-content :space-around;  
4 }  
5  
6  
7  
8  
9  
10
```

Next

Flexbox Froggy is created by [Codepip](#) • [YouTube](#) • [Twitter](#) • [GitHub](#) • [Settings](#)

Want to learn CSS grid? Play [Grid Garden](#).



4.

flexboxfroggy.com

☆

# FLEXBOX FROGGY

◀

Level 4 of 24

▶


Now the lilypads on the edges have drifted to the shore, increasing the space between them. Use `justify-content`. This time, the lilypads have equal spacing between them.


```
1 #pond {  
2   display: flex;  
3   justify-content : space-between;  
4 }  
5  
6  
7  
8  
9  
10
```


Next

Flexbox Froggy is created by [Codepip](#) • [YouTube](#) • [Twitter](#) • [GitHub](#) • [Settings](#)

Want to learn CSS grid? Play [Grid Garden](#).







5.

flexboxfroggy.com

☆

# FLEXBOX FROGGY

◀ Level 5 of 24 ▶

Now use **align-items** to help the frogs get to the bottom of the pond. This CSS property aligns items vertically and accepts the following values:


- **flex-start**: Items align to the top of the container.
- **flex-end**: Items align to the bottom of the container.
- **center**: Items align at the vertical center of the container.
- **baseline**: Items display at the baseline of the container.
- **stretch**: Items are stretched to fit the container.

```
1 #pond {  
2   display: flex;  
3   align-items :flex-end;  
4 }  
5  
6  
7  
8  
9  
10
```

Next

Flexbox Froggy is created by [CodePip](#) • [YouTube](#) • [Twitter](#) • [GitHub](#) • [Settings](#)

Want to learn CSS grid? Play [Grid Garden](#).



6.

flexboxfroggy.com/#hi

☆ | 📁 | 🎵 | 🌐 | ⋮

# FLEXBOX FROGGY

◀ स्तर 6 / 24 ▶


`justify-content` एवं `align-items` का प्रयोग करते हुए मेंढक को तालाब के केंद्र तक पहुंचाइये।

```
1 #pond {
2   display: flex;
3   justify-content : center;
4   align-items : center;
5 }
6
7
8
9
10
```

अगला

Flexbox Froggy के निर्माणकर्ता [Codepip](#) • [YouTube](#) • [Twitter](#) • [GitHub](#) • [Settings](#)

Want to learn CSS grid? Play [Grid Garden](#).



7.

flexboxfroggy.com/#hi

☆

# FLEXBOX FROGGY

◀ स्तर 7 / 24 ▶


मेंढकों को पुनः तालाब पार करना है, इस समय लिली के पत्तों के मध्य काफी जगह मौजूद है।  
justify-content एवं align-items का प्रयोग कीजिये।


```
1 #pond {  
2   display: flex;  
3   align-items : flex-end;  
4   justify-content : space-around;  
5 }  
6  
7  
8  
9  
10
```


अगला

Flexbox Froggy के निर्माणकर्ता [Codepip](#) • [YouTube](#) • [Twitter](#) • [GitHub](#) • [Settings](#)

Want to learn CSS grid? Play [Grid Garden](#).









8.

←

→

↺

flexboxfroggy.com/#hi

☆

🔖

🎵

👤

⋮

# FLEXBOX FROGGY

◀ स्तर 8 / 24 ▶

**flex-direction** की मदद से मेंढकों को लिली के पत्तों को समान क्रम में लाइए। यह CSS प्रॉपर्टी कंटेनर के अन्दर किसी वस्तु की दिशा निर्धारित करती है, एवं निम्न मान स्वीकार करती है:




- **row**: वस्तुएं पाठ दिशा के अनुरूप हो जाती है।
- **row-reverse**: वस्तुएं पाठ दिशा के अनुरूप हो जाती हैं।
- **column**: वस्तुएं लम्ब रूप में उपर से नीचे की ओर हो जाती हैं।
- **column-reverse**: वस्तुएं लम्ब रूप में नीचे से उपर की ओर हो जाती हैं।

```
1 #pond {
2   display: flex;
3   flex-direction :row-reverse;
4 }
```

अगला

Flexbox Froggy के निर्माणकर्ता [Codepi](#) • [YouTube](#) • [Twitter](#) • [GitHub](#) • [Settings](#)

Want to learn CSS grid? [Play Grid Garden.](#)





9.

flexboxfroggy.com/#hi

☆ | 🗂️ | 🎵 | 👤 | ⋮

# FLEXBOX FROGGY

◀ स्तर 9 / 24 ▶

**flex-direction** की मदद से मेंढकों को उनके कॉलम तक पहुंचाइये। यह CSS प्रॉपर्टी कंटेनर के अन्दर किसी वस्तु की दिशा निर्धारित करती है, एवं निम्न मान स्वीकार करती है:




- **row**: वस्तुएं पाठ दिशा के अनुरूप हो जाती है।
- **row-reverse**: वस्तुएं पाठ दिशा के अनुरूप हो जाती हैं।
- **column**: वस्तुएं लम्ब रूप में उपर से नीचे की ओर हो जाती हैं।
- **column-reverse**: वस्तुएं लम्ब रूप में नीचे से उपर की ओर हो जाती हैं।

```
1 #pond {  
2   display: flex;  
3   flex-direction: column;  
4 }  
5  
6  
7  
8  
9  
10
```

अगला

Flexbox Froggy के निर्माणकर्ता [Codepip](#) • [YouTube](#) • [Twitter](#) • [GitHub](#) • [Settings](#)

Want to learn CSS grid? Play [Grid Garden](#).



10.

flexboxfroggy.com/#hi

☆

# FLEXBOX FROGGY

◀ स्तर 10 / 24 ▶

मेंढकों को उनके अपने लिली के पत्ते तक पहुंचाइये। हालांकि यह एक जैसे दिखते हैं, फिर भी इन्हें पहुँचाने के लिए हमें **flex-direction** एवं **justify-content** की मदद लेनी होगी।




ध्यान रखिये यदि आप पंक्ति एवं स्तम्भ की दिशा बदलते हैं, तो उनके शुरुआत एवं अंत भी बदल जाते हैं

```
1 #pond {
2   display: flex;
3   flex-direction: row-reverse;
4   justify-content: flex-end;
5 }
6
7
8
9
10
```

अगला

Flexbox Froggy के निर्माणकर्ता [Codepip](#) • [YouTube](#) • [Twitter](#) • [GitHub](#) • [Settings](#)

Want to learn CSS grid? Play [Grid Garden](#).



11.

flexboxfroggy.com/#hi

☆ | 📁 | 📖 | 👤

# FLEXBOX FROGGY

◀ स्तर 11 / 24 ▶

**flex-direction** एवं **justify-content** से मेंढकों को उनके लिली के पत्ते ढुंढने में मदद कीजिये।




ध्यान दीजिये, जब भी flex direction स्तम्भ की जाती है, **justify-content** ऊर्ध्वाधर में एवं **align-items** छैतिज़ में बदल जाता है।

```
1 #pond {
2   display: flex;
3   flex-direction: column;
4   justify-content: flex-end;
5 }
6
7
8
9
10
```

अगला

Flexbox Froggy के निर्माणकर्ता [Codepip](#) • [YouTube](#) • [Twitter](#) • [GitHub](#) • [Settings](#)

Want to learn CSS grid? Play [Grid Garden](#).



12.

←

→

↻

flexboxfroggy.com/#hi

☆

🔖

🎵

👤

⋮

FLEXBOX FROGGY

◀ स्तर 12 / 24 ▶

flex-direction एवं justify-content की मदद से मेंढकों को उनके लिली के पत्ते तक पहुंचाइये।

1 #pond {

2   display: flex;

3   flex-direction: column-reverse;

4   justify-content: space-between;

5 }

6

7

8




9

10

अगला

Flexbox Froggy के निर्माणकर्ता [Codepip](#) • [YouTube](#) • [Twitter](#) • [GitHub](#) • [Settings](#)

Want to learn CSS grid? Play [Grid Garden](#).



13.

## FLEXBOX FROGGY

◀ स्तर 13 / 24 ▶

`flex-direction`, `justify-content` एवं `align-items` की मदद से मेंढकों को उनके लिली के पत्ते तक पहुंचाइये।

```
1 #pond {  
2   display: flex;  
3   align-items: flex-end;  
4   flex-direction: row-reverse;  
5   justify-content: center;  
6 }  
7  
8  
9  
10
```

अगला

Flexbox Froggy के निर्माणकर्ता [Codepip](#) • [YouTube](#) • [Twitter](#) • [GitHub](#) • [Settings](#)Want to learn CSS grid? Play [Grid Garden](#).

←

→

↺

flexboxfroggy.com/#hi

☆

🔖

☰

👤

⋮

# FLEXBOX FROGGY

◀ स्तर 14 / 24 ▶

कभी-कभी पंक्ति एवं स्तम्भ का क्रम बदलना भी काफी नहीं नहीं होता। इस स्थिति में हम **order** प्रॉपर्टी का इस्तेमाल कर सकते हैं। मूलतः वस्तुओं का मान शून्य होता है। लेकिन इस प्रॉपर्टी के लिए सकारात्मक अथवा नकारात्मक पूर्णांक मान के लिए निर्धारित कर सकते हैं।




**order** प्रॉपर्टी का इस्तेमाल कर मेंढकों को पुनः व्यवस्थित करें।

```
1 #pond {
2   display: flex;
3 }
4
5 .yellow {
6   order :2;
7 }
8
9
10
```

अगला

Flexbox Froggy के निर्माणकर्ता [Codepip](#) • [YouTube](#) • [Twitter](#) • [GitHub](#) • [Settings](#)

Want to learn CSS grid? Play [Grid Garden](#).



15.

←

→

↺

flexboxfroggy.com/#hi

☆

🔖

🔊

👤

⋮

# FLEXBOX FROGGY

◀ स्तर 15 / 24 ▶






**order** प्रॉपर्टी का इस्तेमाल कर लाल मेंढकों को उसके लिली के पत्ते तक पहुंचाइये।

```
1 #pond {
2   display: flex;
3 }
4
5 .red {
6   order:-1;
7 }
8
9
10
```

अगला

Flexbox Froggy के निर्माणकर्ता [Codepip](#) • [YouTube](#) • [Twitter](#) • [GitHub](#) • [Settings](#)

Want to learn CSS grid? Play [Grid Garden](#).





16.

←

→

↻

flexboxfroggy.com/#hi

☆

🔖

🎵

👤

⋮

# FLEXBOX FROGGY

◀ स्तर 16 / 24 ▶





एक अन्य प्रॉपर्टी **align-self** जिसे आप प्रत्येक वस्तु पर अलग-अलग लगा सकते हैं। यह प्रॉपर्टी **align-items** के मान को ही स्वीकार करती है परन्तु किसी विशिष्ट वस्तु के लिए।


```
1 #pond {
2   display: flex;
3   align-items: flex-start;
4 }
5
6 .yellow {
7   align-self: flex-end;
8 }
9
10
```

अगला

Flexbox Froggy के निर्माणकर्ता [Codepip](#) • [YouTube](#) • [Twitter](#) • [GitHub](#) • [Settings](#)

Want to learn CSS grid? Play [Grid Garden](#).





17.

flexboxfroggy.com/#hi

☆

# FLEXBOX FROGGY

◀ स्तर 17 / 24 ▶




**order** एवं **align-self** का इस्तेमाल करते हुए मेंढकों को उनके गंतव्य तक पहुंचाइये।



```
1 #pond {
2   display: flex;
3   align-items: flex-start;
4 }
5
6 .yellow {
7   order: 1;
8   align-self: flex-end;
9 }
10
```

अगला

Flexbox Froggy के निर्माणकर्ता [Codepip](#) • [YouTube](#) • [Twitter](#) • [GitHub](#) • [Settings](#)

Want to learn CSS grid? Play [Grid Garden](#).





flexboxfroggy.com/#hi

☆ | 📁 | 🎵 | 👤 | ⋮

# FLEXBOX FROGGY

◀ स्तर 18 / 24 ▶

अरे नहीं! सभी मेंढक एक ही पंक्ति में चिपक गए हैं। **flex-wrap** प्रॉपर्टी द्वारा उन्हें फैलाइये, यह निम्न मान स्वीकार करता है:






- **nowrap**: प्रत्येक वस्तु एक ही पंक्ति में समायोजित करता है।
- **wrap**: वस्तुएं अतिरिक्त पंक्तियों में समायोजित हो जाती हैं।
- **wrap-reverse**: वस्तुएं अतिरिक्त पंक्तियों में समायोजित होती हैं।



```
1 #pond {  
2   display: flex;  
3   flex-wrap :wrap;  
4 }  
5  
6  
7  
8  
9  
10
```

अगला

Flexbox Froggy के निर्माणकर्ता [Codepip](#) • [YouTube](#) • [Twitter](#) • [GitHub](#) • [Settings](#)

Want to learn CSS grid? [Play Grid Garden.](#)





# FLEXBOX FROGGY

◀ स्तर 19 / 24 ▶

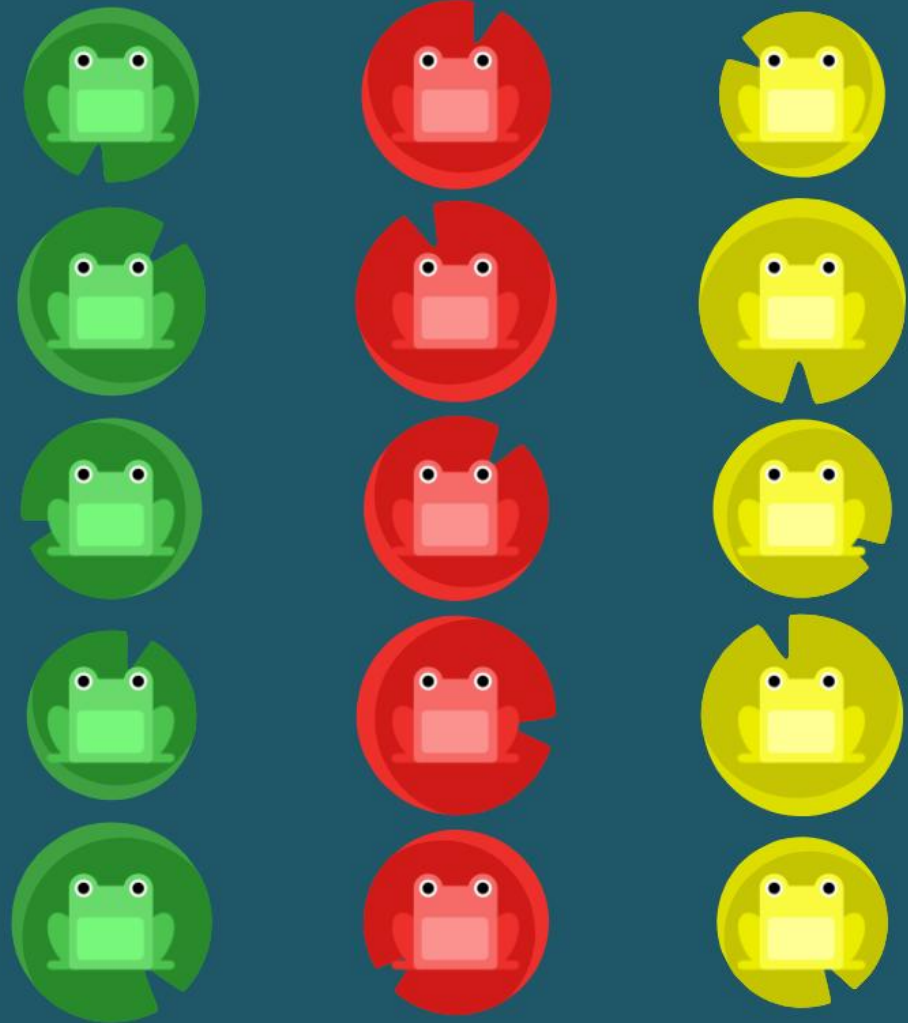
मेंढकों की सेना को `flex-direction` एवं `flex-wrap` के इस्तेमाल से तीन स्तम्भ बनाने में मदद कीजिये।

```
1 #pond {  
2   display: flex;  
3   flex-wrap: wrap;  
4   flex-direction: column;  
5 }  
6  
7  
8  
9  
10
```

अगला

Flexbox Froggy के निर्माणकर्ता [Codepip](#) • [YouTube](#) • [Twitter](#) • [GitHub](#) • [Settings](#)

Want to learn CSS grid? Play [Grid Garden](#).



# FLEXBOX FROGGY

◀ स्तर 20 / 24 ▶

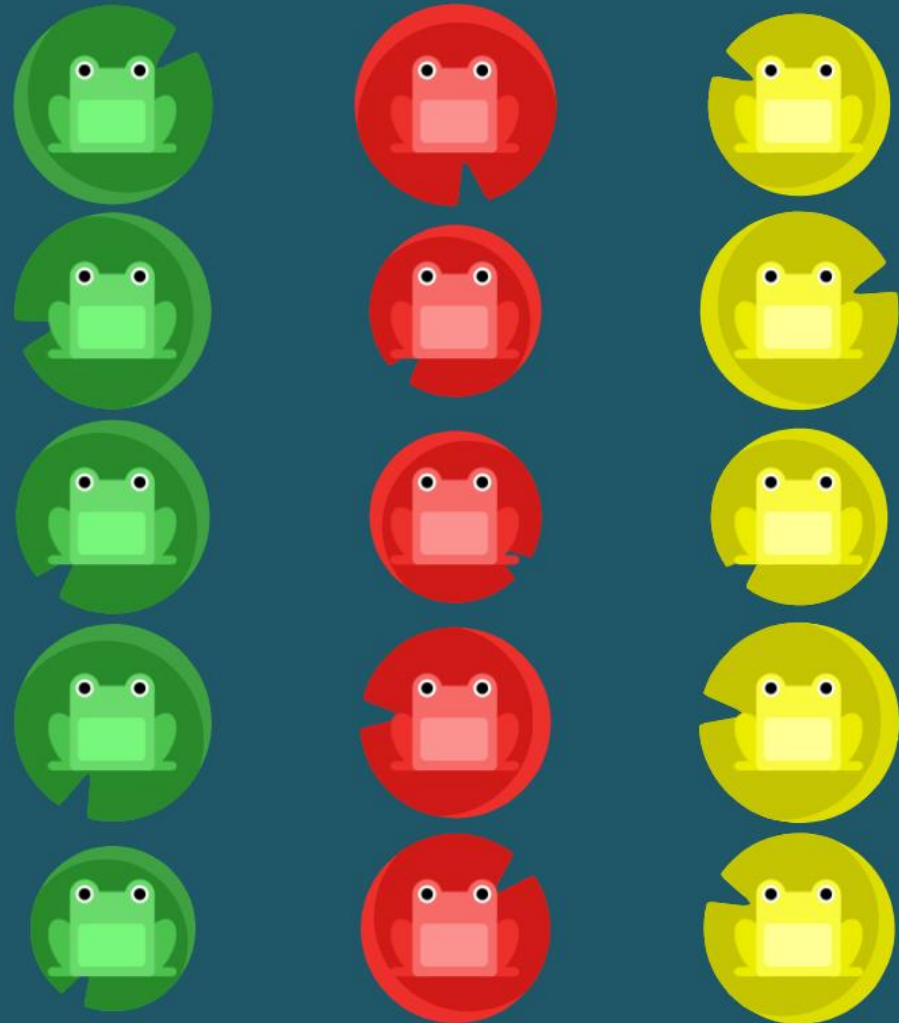
`flex-direction` एवं `flex-wrap` अक्सर इस्तेमाल किये जाते हैं इसलिए `flex-flow` नयी प्रॉपर्टी इनको मिला कर बनाई गयी। यह प्रॉपर्टी इन दोनों में से किसी प्रॉपर्टी के मानों को ग्रहण करती है जो की सामान अंतर में लिखी गयी हों।

उदाहरण के लिए, हम `flex-flow: row wrap` का इस्तेमाल हम वस्तुओं को अनेक पंक्तियों में लाने के लिए कर सकते हैं।

`flex-flow` का इस्तेमाल पिछले स्तर को पार करने में कीजिये।

```
1 #pond {  
2   display: flex;  
3   flex-flow: column wrap;  
4 }  
5  
6  
7  
8  
9  
10
```

अगला

Flexbox Froggy के निर्माणकर्ता [Codepip](#) • [YouTube](#) • [Twitter](#) • [GitHub](#) • [Settings](#)Want to learn CSS grid? Play [Grid Garden](#).



## FLEXBOX FROGGY

◀ स्तर 21 / 24 ▶

मेंढक पूरे तालाब के ऊपर फैले हुए हैं, परन्तु लिली के पत्ते तालाब के ऊपर की तरफ हैं। आप `align-content` की मदद से पंक्तियों के मध्य के अंतर को निश्चित कर सकते हैं। यह प्रॉपर्टी निम्न मान स्वीकार करती है:

- **flex-start**: पंक्तियों को कंटेनर के शीर्ष में रखा जाता है।
- **flex-end**: पंक्तियों को कंटेनर के अंत में रखा जाता है।
- **center**: पंक्तियों को कंटेनर के उर्ध्वाधर केंद्र में रखा जाता है।
- **space-between**: पंक्तियों के मध्य बराबर दूरी रखी जाती है।
- **space-around**: पंक्तियों के चारों ओर दूरी बराबर रखी जाती है।
- **stretch**: पंक्तियों को खींच कर कंटेनर के बराबर किया जाता है।

यह भ्रामक प्रतीत होता है, परन्तु `align-content` पंक्तियों के अंतर को दर्शाता है जबकि `align-items` बताता है की किस तरह कोई वस्तु कंटेनर में संरेखित है। जब केवल एक ही पंक्ति मौजूद हो तो `align-content` का कोई प्रभाव नहीं होगा।

```
1 #pond {  
2   display: flex;  
3   flex-wrap: wrap;  
4   align-content: flex-start;  
5 }  
6  
7  
8  
9  
10
```

अगला



22.

[flexboxfroggy.com/#hi](#)

# FLEXBOX FROGGY

◀ स्तर 22 / 24 ▶

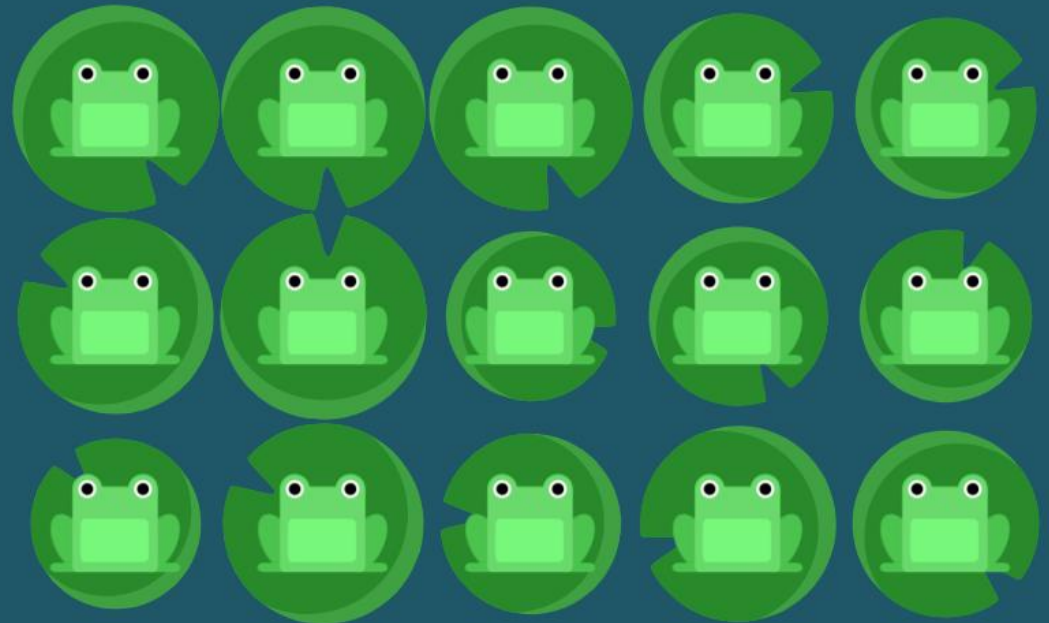
अब लिली के पत्ते तालाब के अंत में आ गये हैं। `align-content` का इस्तेमाल कर मेंढकों को पत्तों तक पहुंचाइये।

```
1 #pond {  
2   display: flex;  
3   flex-wrap: wrap;  
4   align-content: flex-end;  
5 }  
6  
7  
8  
9  
10
```

अगला

Flexbox Froggy के निर्माणकर्ता [Codepip](#) • [YouTube](#) • [Twitter](#) • [GitHub](#) • [Settings](#)

Want to learn CSS grid? Play [Grid Garden](#).





23.

flexboxfroggy.com/#hi

☆ | 📁 | 🎵 | 👤 | ⋮

# FLEXBOX FROGGY

◀ स्तर 23 / 24 ▶

मेंढकों के सामूहिक भोज के बाद अब वक्त है घर जाने का, **flex-direction** एवं **align-content** के इस्तेमाल से इन सभी को घर पहुंचाइये।

```
1 #pond {
2   display: flex;
3   flex-wrap: wrap;
4   flex-direction: column-reverse;
5   align-content: center;
6 }
7
8
9
10
```

अगला

Flexbox Froggy के निर्माणकर्ता [Codepip](#) • [YouTube](#) • [Twitter](#) • [GitHub](#) • [Settings](#)

Want to learn CSS grid? Play [Grid Garden](#).

# FLEXBOX FROGGY

◀ स्तर 24 / 24 ▶

आपने अभी तक जितना सीखा है उसकी मदद से एक आखिरी बार सभी मेंढकों को पत्तों तक पहुंचाइये। सभी प्रॉपर्टीज की सूची :

- `justify-content`
- `align-items`
- `flex-direction`
- `order`
- `align-self`
- `flex-wrap`
- `flex-flow`
- `align-content`

```
1 #pond {  
2   display: flex;  
3   justify-content: center;  
4   flex-flow: column wrap-reverse;  
5   align-content: space-between;  
6   flex-direction: column-reverse;  
7 }  
8  
9  
10
```

अगला

Flexbox Froggy के निर्माणकर्ता [Codepip](#) • [YouTube](#) • [Twitter](#) • [GitHub](#) • [Settings](#)Want to learn CSS grid? Play [Grid Garden](#).