

Typescript spiekbriefje



```
const item = Scene.getItem("myItem");  
item.transform.rotateBy(0, 180, 0, 1); // Rotate by 180 degrees in 1 second
```

```
const item = Scene.getItem("myItem");  
item.input.onClick(() => {  
  // Action to perform when clicked  
});
```

```
const music = Sound.load("soundID");  
music.play(); // Play the sound
```

```
GUI.HUD.showChoicePanel({  
  question: 'This is your question.',  
  answer1: 'First answer.',  
  answer2: 'Second answer.',  
  onAnswer: answer => {  
    if (answer == 1) {  
      // Action for first answer  
    }  
    else if (answer == 2) {  
      // Action for second answer  
    }  
  }  
});
```

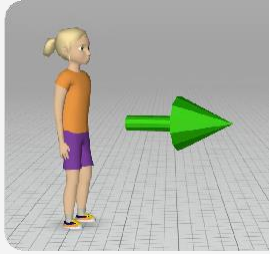
Laatste update: september 2024

**Korte beschrijvingen van
TypeScript code voor beginners**

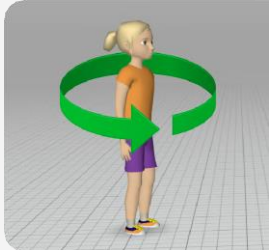


Transformeer

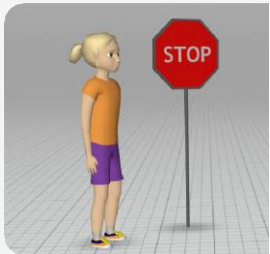
Actie om uit te voeren



Een object of personage in de loop van de tijd **verplaatsen**.



Een object in de loop van de tijd om zichzelf heen laten **draaien**.



De transitie van een object of een personage **stoppen**.

TypeScript-code om te gebruiken

```
const item = Scene.getItem("myItem");
item.transition.moveBy(new Vector3(1, 0, 0), 1);
```

Verplaatst het object met 1 meter langs de x-as in 1 seconde.

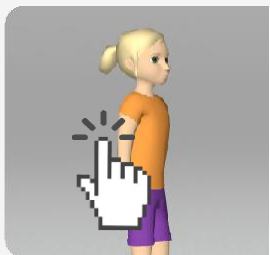
```
const item = Scene.getItem("myItem");
item.transition.rotateLocal(new Vector3(0, 0, 1), 3, 1);
```

Draait het object lokaal met 3 graden om de z-as in 1 seconde.

```
item.transition.stop(); // Stop de huidige transitie van het object
```



Gebeurtenissen

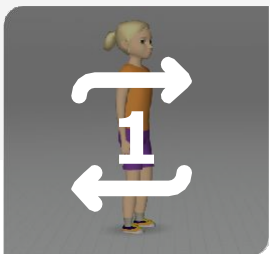


Voert een actie uit wanneer op het object wordt **geklikt**.

```
const item = Scene.getItem("mijnItem");
item.input.onClick(() => {
  // Actie uit te voeren bij klikken
});
```



Controle



Voert één of meer acties **herhaaldelijk** uit.

```
for (let i = 0; i < X; i++) {
  // Herhaalde actie
}
```

Herhaalt de actie x keer



Laat een actie een bepaalde tijd **wachten**.

```
Time.schedule(() => {
  // Actie na 1 seconde
}, 1);
```

Voert de actie uit na 1 seconde.



Gaat naar een specifieke **scène**.

```
Space.goToScene(1);
```

Schakelt over naar de scène met ID 1.



Acties

Actie om uit te voeren



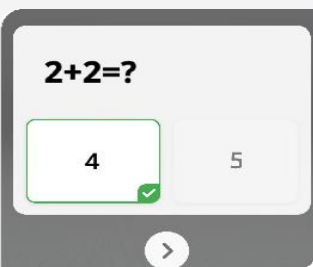
Laat een personage iets **zeggen** of **denken**.



Wijzigt de **transparantie** van een item.



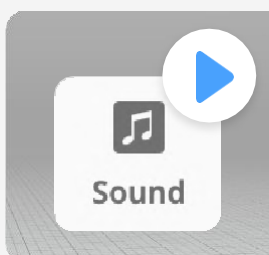
Toont een **infopaneel** met een titel, tekst en een optionele afbeelding.



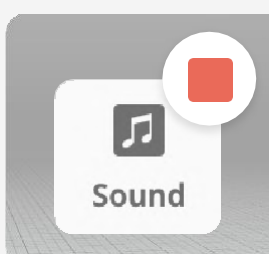
Toont een **quizvenster** met twee of meer antwoorden en voert acties uit op basis van correct of fout antwoord.



Toont een **keuzepaneel** met een vraag en twee of meer antwoorden, en voert acties uit op basis van het gekozen antwoord.



Speelt een geluid af.



Stopt met het afspelen van het geluid.

TypeScript-code om te gebruiken

```
const character = Scene.getItem("mijnKarakter");
character.speech = "Hoi!"; // Karakter zegt "Hoi!"
character.thought = "Hmm.."; // Karakter denkt "Hmm.."
```

```
const item = Scene.getItem("mijnItem");
item.opacity = 0.5; // Stel de transparantie in op 50%
```

Hiermee stel je de transparantie (doorzichtigheid) van het object in op.

```
GUI.HUD.showInfoPanel({
  title: "Titel hier",
  text: "Tekst hier",
  image: "afbeeldingURL" // Optioneel: Voeg een afbeelding toe of laat dit weg
});
```

```
GUI.HUD.showQuizPanel({
  question: 'Hier komt je quizvraag.',
  answer1: 'Antwoord 1',
  answer2: 'Antwoord 2',
  answer3: 'Antwoord 3',
  answer4: 'Antwoord 4',
  correctAnswer: 1, // Geef het juiste antwoordnummer op
  onCorrect: () => {
    // Actie voor juist antwoord
  },
  onWrong: () => {
    // Actie voor fout antwoord
  }
});
```

```
GUI.HUD.showChoicePanel({
  question: 'Hier komt je vraag.',
  answer1: 'Eerste antwoord.',
  answer2: 'Tweede antwoord.',
  onAnswer: answer => {
    if (answer === 1) {
      // Actie voor eerste antwoord
    }
    else if (answer === 2) {
      // Actie voor tweede antwoord
    }
  }
});
```

```
const muziek = Sound.load("geluidID");
muziek.play(); // Speel het geluid af
```

Laadt een geluid en speelt het af.

```
muziek.stop(); // Stop het geluid
```