

```

+-----+
| MovableObject | <<abstract>>
+-----+
| - x: number |
| - y: number |
+-----+
| + update(): void |
| + draw(): void |
+-----+

      ↑

+-----+ +-----+
| snowflake | | Bird |
+-----+ +-----+
| - radius: number | | - speedx: number |
| - speed: number | | - speedy: number |
| | - color: string |
+-----+ +-----+
| + update(): void | | + update(): void |
| + draw(): void | | + draw(): void |
+-----+ +-----+

```

MovableObject:

- Abstrakte Superklasse, die grundlegende Eigenschaften (x, y) und Methoden (update, draw) enthält.
- Wird von snowflake und Bird geerbt.

snowflake:

- Hat spezifische Eigenschaften (radius, speed) und implementiert update und draw

Bird:

- Besitzt zusätzliche Eigenschaften wie speedx, speedy, color und implementiert update sowie draw für die Flugbewegung und das Zeichnen.

```

+-----+
| start |
+-----+
| v |
+-----+
| Initialisiere canvas |
| und Kontext |
+-----+
| v |
+-----+
| Erstelle bewegliche Objekte (z. B. |
| Schneeflocken und Vogel) |
+-----+
| v |
+-----+
| Zeichne den statischen Hintergrund |
| (Himmel, Sonne, Baum, Vogelhaus) |
+-----+
| v |
+-----+
| Aktualisiere jedes Objekt |
| (z. B. Bewegung, Grenzen) |
+-----+
| v |
+-----+
| Zeichne jedes Objekt |
| (z. B. Schneeflocken, Vogel) |
+-----+
| v |
+-----+
| Fordere nächsten Frame an |
| (requestAnimationFrame) |
+-----+
| v |
+-----+
| Wiederhole |
+-----+

```

Die Fünf Fragen der Objektorientierten Modellierung

1. Was sind die Objekte?

- Schneeflocken (snowflake) und Vogel (Bird) sind die beweglichen Objekte.
- Die statischen Objekte wie Baum, Schneemann und Vogelhaus werden nicht animiert und sind daher keine Klassen.

2. Welche Beziehungen gibt es zwischen den Objekten?

- snowflake und Bird erben von der abstrakten Superklasse MovableObject.

3. Welche Eigenschaften und Methoden sind generalisiert?

- Generalisiert in MovableObject:
 - Eigenschaften: x, y.
 - Methoden: update(), draw().

4. Welche Eigenschaften und Methoden sind spezialisiert?

- snowflake:
 - Eigenschaften: radius, speed.
 - Methoden: Eigene Logik in update() und draw().
- Bird:
 - Eigenschaften: speedx, speedy, color.
 - Methoden: Eigene Logik in update() und draw().

5. Welche Objekte können gemeinsam verwaltet werden?

- Alle beweglichen Objekte (snowflake, Bird) werden in einem einzigen Array movableObjects verwaltet.

Die zwei Regeln der objektorientierten Modellierung

1. Vermeide Wiederholungen (DRY-Prinzip):

- Die Eigenschaften `x` und `y` sowie die Methoden `update` und `draw` wurden in der Superklasse `MovableObject` zentralisiert, sodass keine redundante Logik in den Unterklassen existiert.

2. Teile Gemeinsamkeiten, spezialisiere Unterschiede:

- Gemeinsamkeiten (`x`, `y`, `update`, `draw`) wurden in der Superklasse `MovableObject` definiert.
- Unterschiede wie spezifische Bewegungs- und Zeichenlogik wurden in den spezialisierten Klassen (`Snowflake`, `Bird`) implementiert.