

Übungsblatt 3- EIPNF WS

Aufgabe 1:

Hier sehen wir den in der Vorlesung bereits besprochenen Code. Aufgabe ist es nun vier verschiedene drawAnimation Methoden zu schreiben, vier Objekte der Klasse Animation zu erzeugen, mit unterschiedlichen Charactern zu initialisieren und so auf vier Tasten vier verschiedene Animationen abrufen zu können.

```
class Animation {
  int animationCounter;
  char animationKey;

  public Animation (char c) {
    animationCounter = 0;
    animationKey = c;
  }

  void handleInput () {
    if (keyPressed && animationKey == key) {
      animationCounter++;
    } else {
      animationCounter = 0;
    }
  }

  void drawAnimation1 () {
    if (animationCounter>0) {
      // hier steht deine Animation
    }
  }

  void drawAnimation2 () {...}
  void drawAnimation3 () {...}
  void drawAnimation4 () {...}
}

// -----

Animation a;
// deklariere weitere Objekte

void setup () {
  size(600, 400);
  a = new Animation('1');
  // initialisiere alle Objekte
}

void draw () {
  background(0);
  a.handleInput();
  a.drawAnimation1();
  // führe für jedes Objekt die entsprechenden Animationen aus
}
```

Aufgabe 2:

Schreibe eine Funktion `blinkingCircle()`, die einen blinkenden Kreis an einer Stelle `x,y` mit Radius gleich `r` zeichnet. Und keinen Wert zurückgibt. Das Blinken ist abhängig von einem `counter`. `x,y,r` und `counter` werden der Funktion als Parameter übergeben.

Verwende die Funktion `isEven()` um zu überprüfen, ob `counter` gerade ist, ist dies der Fall soll der Kreis gezeichnet werden, ansonsten soll kein Kreis gezeichnet werden.

```
int counter;

void setup () {
  size(100,100);
  counter = 0;
}

void draw () {
  counter++;
  //rufe blinkingCircle auf
  //übergebe beliebige werte für x,y,r, übergebe counter
}

//definiere blinkingCircle
//...

bool isEven (int value) {
  if (value%2==0) {
    return true;
  }
  else {
    return false;
  }
}
```