



ZEROSEVEN

**DIMEKO**

gGmbH

**DIMEKO**

„DIGITALE  
MEDIENKOMPETENZ“  
FÜR JUNG UND ALT

# 3D- Druck

---

- Kinder und Jugendliche werden mit Spaß an das Thema 3D Drucker herangeführt.
- Dazu benutzen wir einen Lebensmittel- Drucker.
- Es können vereschieden Formen gedruckt werden.
- Der Benutzer muss dass zu druckende Objekt, jedoch erst in eine 3D Druck- fähige Datei wandeln.



# 3D- Drucker

---



# Digital AG

---

- Moderne Fortbildungsmöglichkeit für Schüler ab 6. Klasse
- Gewährt Einblicke in mehrere Berufsfelder ohne vorkenntnisse
- Die Zeiten der AG sind an die entsprechenden Partnerschulen angepasst
- Die AG ist in drei Bereiche unterteilt:
  - “Robotik”
  - “digitale Musikproduktion”
  - “Programmierung in dreidimensionalen Welten”



# Digital AG – Robotik

---

- In dem Bereich der Robotik werden den Teilnehmern Grundlagen der Robotik und des Programmieren vermittelt
- Die Schüler können eigene Handlungsweisen und Algorithmen für den Roboter programmieren



# Digital AG – digitale Musik

---

- Hier werden den Teilnehmern Grundfähigkeiten der analogen und digitalen Musikproduktion näher gebracht
- Es wird klassische analoge Musik produziert
- Diese wird im Anschluss digitalisiert und an einem PC abgemischt



# Digital AG – Programmierung

---

- Fokus auf die Interaktion von Objekten in einem 3D Raum
- Dabei werden wichtige physikalische und mathematische Grundlagen (z.B. Vektorenrechnung oder Funktionen) in Koordinatensystemen erklärt
- Das entsprechende Objekt können die Schüler im Anschluss in virtueller Realität begutachten
- Sie erlangen tiefe Einblicke in den Alltag eines Programmierers

```
def __init__(self):
    self.file = None
    self.fingerprints = set()
    self.logdupes = True
    self.debug = debug
    self.logger = logging.getLogger(__name__)
    if path:
        self.file = open(os.path.join(path, 'fingerprint.log'), 'a')
        self.file.seek(0)
        self.fingerprints.update(self.fingerprints)

    @classmethod
    def from_settings(cls, settings):
        debug = settings.getbool('debug')
        return cls(job_dir(settings), debug)

    def request_seen(self, request):
        fp = self.request_fingerprint(request)
        if fp in self.fingerprints:
            return True
        self.fingerprints.add(fp)
        if self.file:
            self.file.write(fp + os.linesep)

    def request_fingerprint(self, request):
        return request_fingerprint(request)
```

Vielen Dank für Ihre  
Aufmerksamkeit!

---



ZEROSEVEN

**DIMEKO**

gGmbH