

# Darija Dobrotschajewa

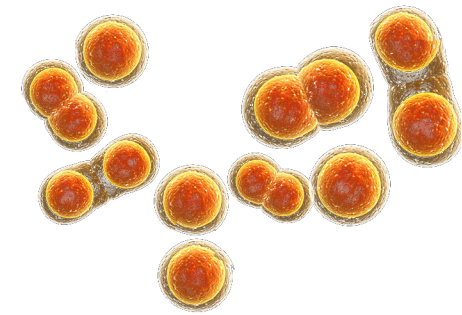


Daria Dobrochaeva ist eine ukrainische Bakteriologin und Immunologin, die für ihre Arbeit auf dem Gebiet der Mikrobiologie bekannt ist. Sie leistete bedeutende Beiträge zur Erforschung von Bakteriophagen und deren Einsatz zur Bekämpfung von Infektionskrankheiten. Ihre wissenschaftlichen Leistungen trugen zur Verbesserung der Behandlung bakterieller Infektionen bei und trugen zur Entwicklung der immunologischen Forschung in der Ukraine bei.

## Fassade für Daria Dobrochaeva (Bakteriologin)

Details: Fassadenelemente, die dem Aufbau von Mikroorganismen unter dem Mikroskop ähneln: stilisierte Zellen, Moleküle und Bakterien. Sowohl die Treppe als auch die Seitenteile des Elements ähneln der Struktur einer DNA. Fassadenelemente, die auf der Fassade abgerufen werden können sind als Panels programmiert. Die Fassade verfügt über ein eigenes Raster und bei jedem Start ändern die Paneele ihre Größe und Position entsprechend dem neuen Raster. Die umliegende Lage spiegelt die Stadt Kiew wider, wo Daria einen bedeutenden Teil ihres Lebens und ihrer beruflichen Tätigkeit verbrachte.

```
for i, panel in enumerate(divPanels[0:]):
    if random.uniform(0, 100) < 40:
        MikrobePanel(panel)
    elif 40 < random.uniform(0, 100) < 95:
        MikrobePanel2(panel)
    else:
        kugel(panel)
```



DM2\_WS2024

Gr: Gruber

Svitlana Kisteniuik

```
import DM_lib as dm
.....
frontPanels = [ [gC(L,0,H), gC(L,0,H+1), gC(L+1,0,H+1), gC(L+1,0,H)]
for L in range(7) for H in range(7) ]

.....
def MikrobePanel2 (panel):
    p0,p1,p2,p3 = panel
    coords = [p0,p1,p2,p3,p0]
    center=dm.pntInbetween(p0,p2)
    vec1 = rs.VectorCreate(p0,p3)
    negvec1 = rs.VectorRotate(vec1,180,[0,0,1])
    vech = rs.VectorCreate(p0,p1)
    negvech = [vech[0],vech[1],-vech[2]]

    #Q1 h+, l+
    q1 = rs.VectorAdd(center,
rs.VectorScale(vec1,random.uniform(0.1,0.5)))
    q1 = rs.VectorAdd(q1,
rs.VectorScale(vech,random.uniform(0.1,0.5)))
    ...
    center = dm.pntInbetween(q1,q3)
    crvMikrobe2 = rs.AddCurve([q1, q2,q3, q4,q1])
    rs.ObjectColor(crvMikrobe2, [random.randint(50, 200),
random.randint(50, 200), 100])

    scal = rs.VectorScale(dVec,-1*rs.Distance(p0,p3)*.2)
    cntCrv = rs.VectorAdd(center,scal)
    points = random.randint(25,40)
    pntCrv = rs.DivideCurve(crvMikrobe2,points)
    for i in range(points):
        pntIn = dm.pntInbetween(pntCrv[i],center)
        pntIn = rs.VectorAdd(pntIn,scal)
        rs.AddCurve([pntCrv[i],pntIn,cntCrv])
```

