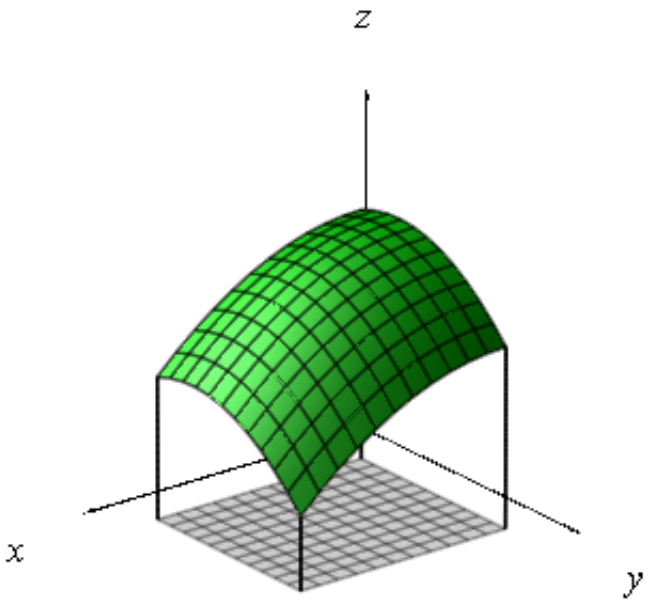
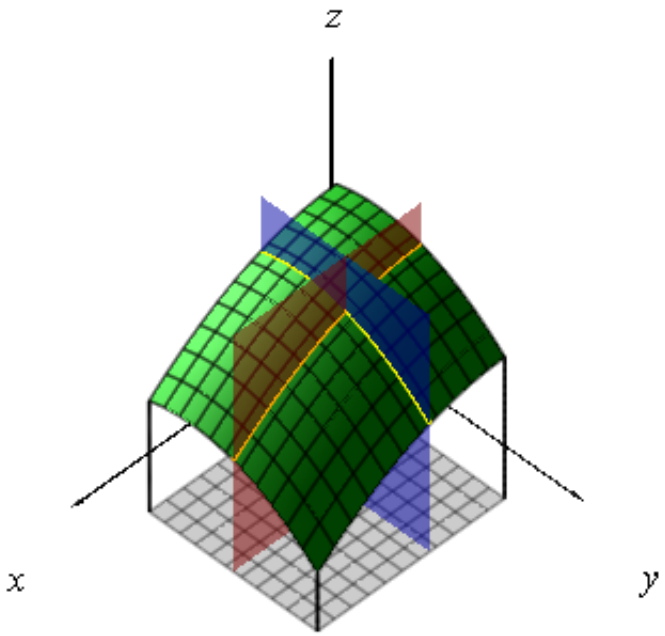
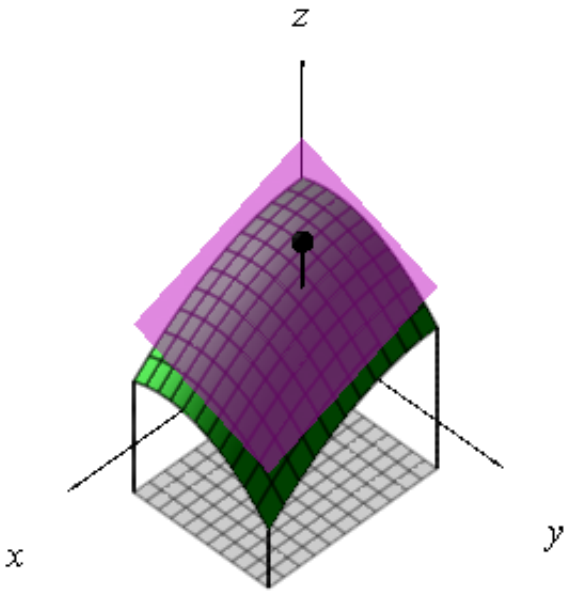
```
> with(plots):
> f := (x, y) → 6 - ((x - 1)^2 + 2·(y - 2)^2)/10 :
> f1 := D[1](f):
> f2 := D[2](f):
> Flate := plot3d([x, y, f(x, y)], x = 1..6, y = 1..5, grid = [12, 12], color = "Green"):
> zLinje := proc(x, y, z0, z1) if z0 < z1 then spacecurve([x, y, t], t = z0..z1, color = "Black",
      linestyle = dash, thickness = 1) else if z1 < z0 then spacecurve([x, y, t], t = z1..z0, color
      = "Black", linestyle = dash, thickness = 1) end if; end if; end proc:
> xPlan := x → plot3d([x, y, z], y = 1..5, z = 0..7, color = "Blue", style = patchnogrid, transparency
      = 0.5):
> yPlan := y → plot3d([x, y, z], x = 1..6, z = 0..7, color = "Red", style = patchnogrid, transparency
      = 0.5):
> xKurve := x → spacecurve([x, y, f(x, y)], y = 1..5, thickness = 2, color = "Yellow"):
> yKurve := y → spacecurve([x, y, f(x, y)], x = 1..6, thickness = 2, color = "Orange"):
> xTangent := spacecurve([3, 3 + t, f(3, 3) + f2(3, 3) * t], t = -2..2, color = "Cyan", thickness
      = 2):
> yTangent := spacecurve([3 + s, 3, f(3, 3) + f1(3, 3) * s], s = -2..3, color = "Red", thickness
      = 2):
> Tangentplan := plot3d([3 + s, 3 + t, f(3, 3) + s*f1(3, 3) + t*f2(3, 3)], s = -2..3, t = -2..2,
      style = patchnogrid, transparency = 0.5, color = "Magenta"):
> Normalvektor := arrow([3, 3, f(3, 3)], [-f1(3, 3), -f2(3, 3), 1], length = 4, color = "Black",
      width = 0.07, head_length = 0.5, head_width = 0.5):
> xyProjeksjon := plot3d([x, y, 0], x = 1..6, y = 1..5, color = "Grey", transparency = .5, grid
      = [12, 12]):
> Bakgrunnsbilde := display(Flate, xyProjeksjon, zLinje(1, 1, 0, f(1, 1)), zLinje(1, 5, 0, f(1, 5)),
      zLinje(6, 1, 0, f(6, 1)), zLinje(6, 5, 0, f(6, 5)),
  textplot3d([0.5, 0.5, 10,'z'], font = [helvetica, 14], color = black),
  textplot3d([9, 0.5, 0,'x'], font = [helvetica, 14], color = black),
  textplot3d([0.5, 8, 0,'y'], font = [helvetica, 14], color = black),
  arrow([0, 0, 0], [1, 0, 0], length = 7, color = black, width = 0.05, head_length = 0.3, head_width
      = 0.1),
  arrow([0, 0, 0], [0, 1, 0], length = 6, color = black, width = 0.05, head_length = 0.3, head_width
      = 0.1),
  arrow([0, 0, 0], [0, 0, 1], length = 8, color = black, width = 0.05, head_length = 0.3, head_width
      = 0.1), projection = 0.85 ):
> display(Bakgrunnsbilde, projection = 0.9, orientation = [52, 68]);
```




```
> A1 := display(Bakgrunnsbilde, yPlan(3), yKurve(3)) :
> A2 := display(Bakgrunnsbilde, xPlan(3), xKurve(3)) :
> A3 := display(Bakgrunnsbilde, xPlan(3), xKurve(3), yPlan(3), yKurve(3), zLinje(3, 3, 0, 7)) :
> display(Bakgrunnsbilde, A1, A2, A3, insequence = true);
```




```
> B1 := display(Bakgrunnsbilde, yPlan(3), yKurve(3), yTangent) :
> B2 := display(Bakgrunnsbilde, xPlan(3), xKurve(3), xTangent) :
> B3 := display(Bakgrunnsbilde, xPlan(3), xKurve(3), xTangent, yTangent, yPlan(3),
      yKurve(3), zLinje(3, 3, 0, 7)) :
> B4 := display(Bakgrunnsbilde, xPlan(3), xKurve(3), xTangent, yTangent, yPlan(3), yKurve(3),
      Tangentplan) :
> B5 := display(Bakgrunnsbilde, Tangentplan, Normalvektor) :
> display(Bakgrunnsbilde, B1, B2, B3, B4, B5, insequence = true, scaling = constrained);
```


z
x
y