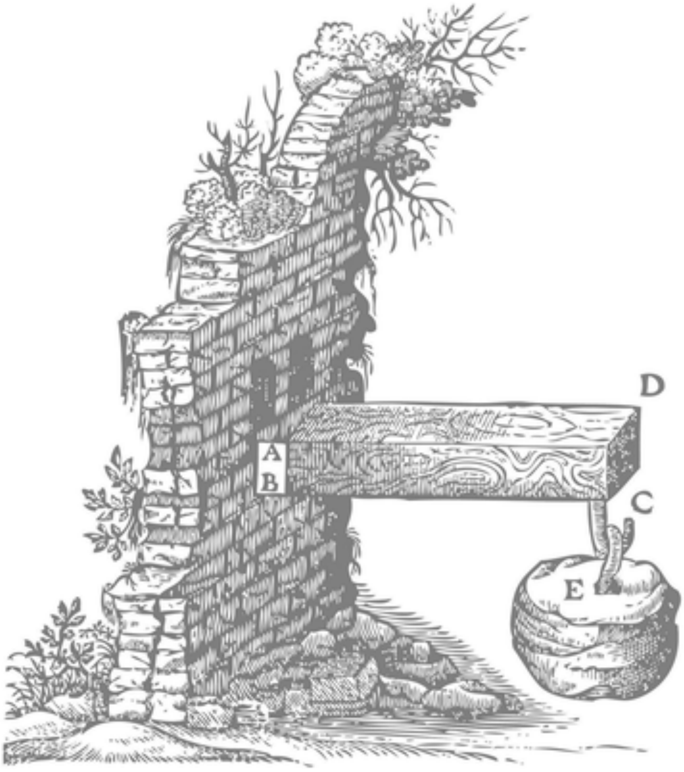
Engineering Mechanics

Beam Bending

In[154]:=
```
Remove["Global`*"]
entferne
SetOptions[SelectedNotebook[],
setze Optionen ausgewähltes Notebook
 PrintingStyleEnvironment → "Printout", ShowSyntaxStyles → True]
 Stilumgebung des Druckformats                                  wahr
(*"
  course: Engineering Mechanics (three semesters)
  cohort: ss 2021 - ws 2021/2022 - ss 2022
  course section: Engineering Mechanics 1
  seminar no: 14
  topic: beam theory
  subtopic: a simple scenario to indicate
  the limitations of linear theory in engineering mechanics
  author: Dr. Marcus Aßmus
  university: Otto von Guericke University
  institute: Institute of Mechanics
  "*)

 (*"
problem description:
  - straight beam
  - clamped single-sided (cantilever beam)
  - load acts at free end of the beam, we neglect weight force
  - rectangular cross-section
  - x (index 1) is longitudinal beam axis, y
  (index 2) is horizontal axis, z (index 3) is vertical axis
  - origin of coos is in the clamping
  - length is significantly greater
  than the cross-sectional dimensions ( L1 >> L2,L3 )
  - isotropic, homogeneous, elastic material
  - let us have a look at the follwing picture as an example...
  "*)

beam = ImageResize[
       passe Größe des Bildes an
   Import["https://raw.githubusercontent.com/marcusassmus/marcusassmus.github.io/
   importiere
      cd82beb521c844a572f796bc064e168bfeeb3cad/work/14.jpg",
    "Graphics"
   ], 400];
text := "source: Galileo Galilei, Discorsi e dimostrazioni matematiche
    intorno a due nuove scienze, Lodewijk Elzevir, Leiden, 1638";
Show[beam]
zeige an
Text[Style[text, Black, Small]]
Text Stil          schwarz  klein
```

Out[158]=



Out[159]= *source*: Galileo Galilei, Discorsi e dimostrazioni matematiche intorno a due nuove scienze, Lodewijk Elzevir, Leiden, 1638

In[160]:=
```
(* geometry [m] *)
L1 = 2;
L2 = L1 / 20;
L3 = L2;

(* second moment of area [m⁴] *)
Iy = L2 * L3^3 / 12;

(* material properties [N/m²] *)
Y = 210 * 10^9; (* Young's modulus *)

(* free end load  [N] *)
F = 1 * 10^5; (* chosen so that a difference in the results is apparent *)

(* boundary conditions *)
bc = {w[0] == 0,
   w'[0] == 0,
   w''[L1] == 0,
   w'''[L1] == -F / (Y * Iy)};

(* key figures *)
Print["SB = ", N[Y * Iy ] "Nm²"] (* bending stiffness *)
gib aus          numerischer Wert
Print["M = ", N[F * L1] "Nm"] (* applied bending moment *)
gib aus          numerischer Wert
Print["LL = ", N[F * L1^2] "Nm²"] (* load level, ie.,
gib aus          numerischer Wert
applied bending moment scaled by the lever arm length L1 *)
Print["R = ", N[(F * L1^2) / (Y * Iy)]]
gib aus          numerischer Wert
(* dimensionless ratio of applied bending moment to bending stiffness,
R<<1 indicates stiff response (small deflections),
while R~1 signals significant deformation relative to length *)
```

SB = 1.75×10^6 Nm²

M = 200 000. Nm

LL = 400 000. Nm²

R = 0.228571

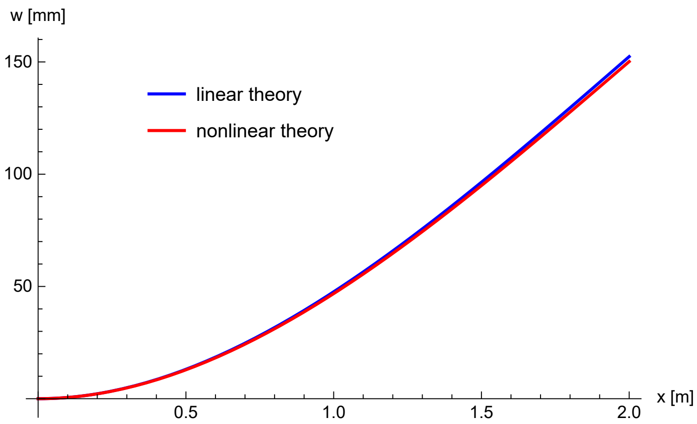
In[171]:=
```
(* ordinary differential equations of boundary value problem *)
eqlin = {Y * Iy * D[w[x], {x, 4}] == 0}; (* geometrically linear theory *)
                 leite ab
eqNL = {Y * Iy * D[D[w[x], {x, 2}] / (1 + (D[w[x], x])^2)^(3/2), {x, 2}] == 0};
                 ·· leite ab                  leite ab
(* geometrically nonlinear theory *)
```

In[172]:=
```
(* solve ODE's *)
solLin = DSolve[{eqlin, bc}, w, x]; (* linear *)
         löse Differentialgleichung

solNL = NDSolve[{eqNL, bc}, w, {x, 0, L1} ]; (* nonlinear *)
        löse Differentialgleichung numerisch

(* determine some values for the point of maximum deflection x=L1 *)
wLin[x_] = w[x] /. solLin[[1]];
wNL[x_] = w[x] /. solNL[[1]];
Print["wLin(L1)= ", N[wLin[L1]] "m"]
gib aus             numerischer Wert
Print["wNL(L1)= ", N[wNL[L1]] "m"]
gib aus            numerischer Wert
Print["Δw(L1)= ", N[wLin[L1] - wNL[L1]] "m"]
gib aus           numerischer Wert
Print["wLin(L1)/L3= ", N[wLin[L1] / L3] ]
gib aus                numerischer Wert
(* maximum deflection to thickness ratio *)
```

```
wLin(L1)= 0.152381 m

wNL(L1)= 0.150165 m

Δw(L1)= 0.00221559 m

wLin(L1)/L3= 1.52381
```

In[180]:=
```
(* plot both results and compare w(x) [mm] *)
Plot[Evaluate[{1000 * w[x] /. solLin[[1]], 1000 * w[x] /. solNL[[1]] }],
stell··· werte aus
 {x, 0, L1}, PlotStyle → {Blue, Red},
             Darstellungsstil blau rot
 PlotLegends → Placed[{"linear theory", "nonlinear theory"}, {0.35, 0.8}],
 Legenden der G··· plaziert
 AxesLabel → {"x [m]", "w [mm]"}]
 Achsenbeschriftung
```

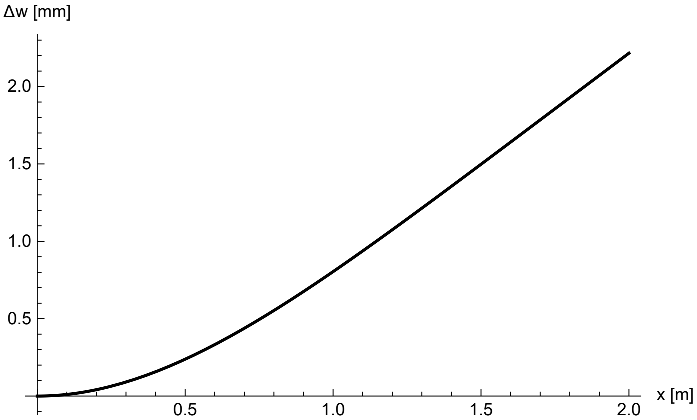
Out[180]=



In[181]:=

```
wdiff[x_] = (wLin[x] - wNL[x]);

(* plot the difference of the solutions gained *)
Plot[1000 * wdiff[x], {x, 0, L1}, PlotStyle → Black,
 AxesLabel → {"x [m]", "Δw [mm]"}, AxesOrigin → {0, 0}]
```

Out[182]=

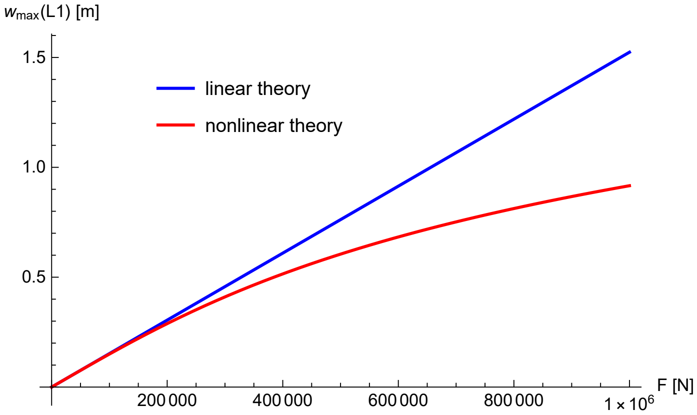


In[183]:=

```
(* parametrize the load 'F' - up to truly large deflections *)
Remove[F]
Fmin = 0;
Fmax = 10^6; (* !!! this maximum value is one
   order of magnitude higher than the previously used !!! *)
bc = {w[0] == 0,
   w'[0] == 0,
   w''[L1] == 0,
   w'''[L1] == -F / (Y * Iy)};

(* solve once again *)
wLinMax[F_] := Module[{sol}, sol = DSolve[{eqlin, bc}, w, x];
  w[L1] /. sol[[1]]]

wNLMax[F_?NumericQ] := Module[{sol}, sol = NDSolve[{eqNL, bc}, w, {x, 0, L1}];
  w[L1] /. sol[[1]]]
```

In[189]:=
```
(* plot maximum deflection in dependence of load *)
(* this shows the deviation of both approaches and determines
 where geometrical nonlinear theory must replace the linear model *)
Plot[{wLinMax[F], wNLMax[F]}, {F, Fmin, Fmax},
 AxesLabel → {"F [N]", "wmax(L1) [m]"}, PlotStyle → {Blue, Red},
 PlotLegends → Placed[{"linear theory", "nonlinear theory"}, {0.35, 0.8}]]
```
stelle Funktion graphisch dar
Achsenbeschriftung | Darstellungsstil | blau | rot
Legenden der G··· | plaziert

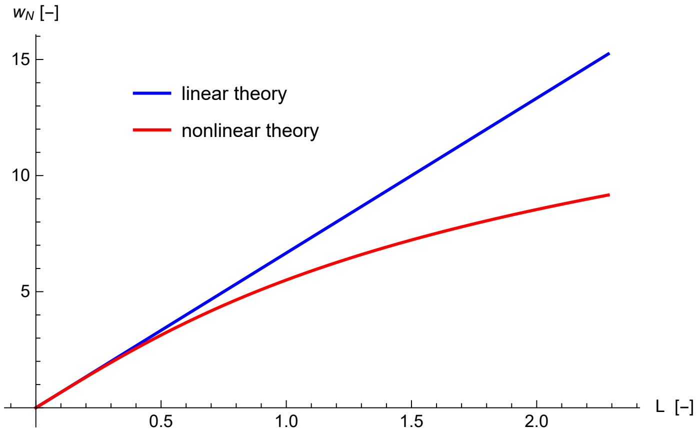
Out[189]=



In[190]:=
```
(*"
sometimes, above plot is normalized by,
e.g., wmax/L3 over (F*L1^2)/(Y*Iy),
resulting in a kind of generalized rendition for comparison purposes,
which, however, gives a dimensionless representation;
wmax/L3 measures curvature/rotation severity relative to section depth;
(F*L1^2)/(Y*Iy) quantifies the load level relative to the bending stiffness;
in linear theory, the slope is a direct measure of the
  slenderness ratio (L1/L3) scaled by a constant (say c, here c=1/3)
"*)

L[F_] = (F * L1^2) / (Y * Iy) ; (* normalized load L, where L(Fmax) = R holds *)
wLinMaxN[F_] = wLinMax[F] / L3; (* dimensionless deflection - linear theory *)
wNLMaxN[F_] = wNLMax[F] / L3; (* dimensionless deflection - nonlinear theory *)

ParametricPlot[{{L[F], wLinMaxN[F]}, {L[F], wNLMaxN[F]}},
 {F, 0, Fmax}, AxesLabel → {"L  [-]", "wN  [-]"}, PlotStyle → {Blue, Red},
 PlotLegends → Placed[{"linear theory", "nonlinear theory"}, {0.35, 0.8}],
 AspectRatio → 1 / GoldenRatio]
```

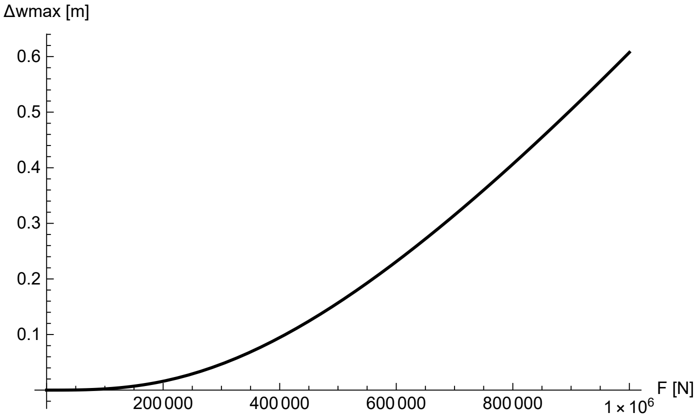
parametrische Darstellung · Achsenbeschriftung · Darstellungsstil · blau · rot · Legenden der G··· · plaziert · Seitenverhältnis · Goldener Schnitt

Out[193]=



In[194]:=
```
(* let us have a look at the differences between both theories more closely *)
wMaxDiff[F_] = (wLinMax[F] - wNLMax[F]); (* absolute difference *)
wMaxDiffRel[F_] = (wLinMax[F] - wNLMax[F]) / wNLMax[F] * 100;
(* relative difference *)

Plot[wMaxDiff[F], {F, Fmin, Fmax}, PlotStyle → Black,
 AxesLabel → {"F  [N]", "Δwmax  [m]"}, AxesOrigin → {0, 0}]
Plot[wMaxDiffRel[F], {F, Fmin, Fmax}, PlotStyle → Black,
 AxesLabel → {"F  [N]", "Δwmax  [%]"}, AxesOrigin → {0, 0},
 Prolog → {Opacity[0.2, Green], Rectangle[{Fmin, 0}, {Fmax, 5}]}
 (* 5% difference regime *)
]
```
stelle Funktion graphisch dar | Darstellungsstil | schwarz | Achsenbeschriftung | Achsenursprung | stelle Funktion graphisch dar | Darstellungsstil | schwarz | Achsenbeschriftung | Achsenursprung | Prolog | Deckkraft | grün | Rechteck

Out[196]=



Out[197]=