| abstract class **AbstractElement**  string ElementId, string[] NodeIds, Node[] Nodes, int ElementDOF, int NodesPerElement, int[] SystemIndicesOfElement, string ElementMaterialId , string ElementCrossSectionId, AbstractMaterial ElementMaterial, int Type, double[] ElementState , double[] ElementDeformations, double Determinant,  abstract double[,] ComputeMatrix(); abstract double[] ComputeDiagonalMatrix(); abstract void SetSystemIndicesOfElement(); abstract double[] ComputeVerformungsvektor(); abstract Point CenterOfGravity();  void SetReferences(FEModel modell); |
| --- |

| abstract class **Abstract2D** :  **AbstractElement**  CrossSection ElementCrossSection  void SetCrossSectionReferences(FEModel modell)  abstract double[] ComputeElementState(double z0, double z1) | abstract class **Abstract3D** : **AbstractElement**  abstract double[] ComputeElementState(double z0, double z1, double z2); |
| --- | --- |

| abstract class **AbstractLinear2D2** :  **Abstract2D**  double[] Sx  double ComputeLength()  void ComputeGeometry()  double[] ComputeSx()  override void SetSystemIndicesOfElement() | abstract class **AbstractLinear2D3** : **Abstract2D**  double[] Sx  void ComputeGeometry()  double[] ComputeSx()  private class AlgebraicException : Exception | abstract class **AbstractLinear2D4** : **Abstract2D**  double[] Sx  void ComputeGeometry(double z0, double z1)  double[,] ComputeSx(double z0, double z1)  double[] ComputeS(double z0, double z1) | abstract class **AbstractLinear2D4** : **Abstract3D**  public double[,] Sx  void ComputeGeometry(double z0, double z1, double z2)  double[,] ComputeSx(double z0, double z1, double z2) |
| --- | --- | --- | --- |

| abstract class **AbstractBeam : AbstractLinear2D2**  abstract double[] ComputeElementState(); |
| --- |

| partial class **Biegebalken** : **AbstractBeam**  public Biegebalken(string[] eNodeIds, string eCrossSectionID, string eMaterialID, FEModel feModell)  override double[,] ComputeMatrix()  override double[] ComputeDiagonalMatrix()  double[] ComputeLoadVector(AbstractLoad ael, bool inElementCoordinateSystem)  override double[] ComputeElementState()  override double[] ComputeElementState(double \_z0, double \_z1)  override double[] ComputeVerformungsvektor()  override void SetSystemIndicesOfElement()  override Point CenterOfGravity() | partial class **BiegebalkenGelenk** : **AbstractBeam**  BiegebalkenGelenk(string[] eNodeIds, string eMaterialID, string eCrossSectionID, FEModel feModell, int \_type)  override double[,] ComputeMatrix()  override double[] ComputeDiagonalMatrix()  double[] ComputeLoadVector(AbstractLoad ael, bool inElementCoordinateSystem)  override double[] ComputeElementState()  override double[] ComputeElementState(double \_z0, double \_z1)  override double[] ComputeVerformungsvektor()  override void SetSystemIndicesOfElement()  override Point CenterOfGravity() | class **Fachwerk** : **AbstractBeam**  Fachwerk(String[] eNodes, String crossSectionId, String materialId, FEModel feModell)  override double[,] ComputeMatrix()  override double[] ComputeDiagonalMatrix()  static double[] ComputeLoadVector(AbstractElementLoad ael, bool inElementCoordinateSystem)  override double[] ComputeElementState()  override double[] ComputeVerformungsvektor()  override void SetSystemIndicesOfElement()  override Point CenterOfGravity() | class **FederElement** : **AbstractElement**  FederElement(String[] springNode, string eMaterialId, FEModel eModell)  override double[,] ComputeMatrix()  override double[] ComputeDiagonalMatrix()  override double[] ComputeVerformungsvektor()  override void SetSystemIndicesOfElement()  override Point CenterOfGravity() |
| --- | --- | --- | --- |