**Package Lagerverwaltung**

**Beschreibung des Packages**

Das Package *lagerverwaltung* umfasst alle Aspekte der Produktverwaltung und enthält Klassen zur optischen Aufbereitung von Lagerbeständen und enthaltenden Produkten.

**Diagramme**

**TO DO**

**Schnittstellen**

Das Package *lagerverwaltung* hat die Schnittstellen zu *verkaufsverwaltung* und zu *kassenfunktion*.

**Aufteilung der Klassen in das Drei-Schichten-Modell:**

GUI-Schicht: LagerVerwaltungsFrame, ProduktBearbeitenFrame, ProduktHinzufügenFrame

Logik-Schicht: LagerTableModel, MengeCellRenderer, PreisCellRenderer, Produkt, FoFormats

Persistenz-Schicht:

**Operationen**

ProduktHinzufuegenFrame:

* Konstruktor: package ProduktHinzufuegenFrame(JFrame, List<Produkt>,List<Produkt>)
* innere Klasse: MyOKHandler
  + public void actionPerformed(ActionEvent)
* innere Klasse: MyAbbHandler
  + public void actionPerformed(ActionEvent)

ProduktBearbeitenFrame:

* Konstruktor: package ProduktBearbeitenFrame(JFrame, Produkt)
* innere Klasse: MyOKHandler
  + public void actionPerformed(ActionEvent)
* innere Klasse: MyAbbHandler
  + public void actionPerformed(ActionEvent)

Produkt:

* Konstruktor: public Produkt(String, double, int, int, boolean, int)
* public int getVorratsmenge()
* public void setVorratsmenge(int)
* public int getUntergrenze()
* public void setUntergrenze(int)
* public boolean isAbfüllmaterial()
* public void setAbfüllmaterial(Boolean)

PreisCellRenderer:

* Konstruktor: public PreisCellRenderer()
* Public Component getTableCellRendererComponent(JTable, Object, boolean, boolean, int, int)

MengeCellRenderer:

* Konstruktor: public MengeCellRenderer()
* public Component getTableCellRendererComponent(JTable, Object, boolean, boolean, int, int)

LagerVerwaltungFrame:

* Konstruktor: public LagerVerwaltungFrame(List<Produkt>, List<Produkt>)
* innere Klasse: ActionListenerActionListener()
* public void actionPerformed(ActionEvent)
* innere Klasse: SelectionListener(ListSelectionListener())
  + public void valueChanged(ListSelectionEvent)
* innere Klasse: addMouseListener(MouseAdapter())
  + public void MousePressed(MouseEvent)
* private void addProdukt()
* private void deleteProdukt()
* private void bearbeiteProdukt()
* private void printListe()

LagerTableModel:

* Konstruktor: package LagerTableModel(List<Produkt>, List<Produkt>)
* public int getColumnCount()
* public String getColumnName(int)
* public int getRowCount()
* public Object getValueAt(int, int)
* public Class<?> getColumnClass(int)
* public Produkt getProdukt(int)
* public List<Produkt> getZusatzProdukte()
* public void setZusatzProdukte(List<Produkt>)
* public void setAbfuellProdukte(List<Produkt>)