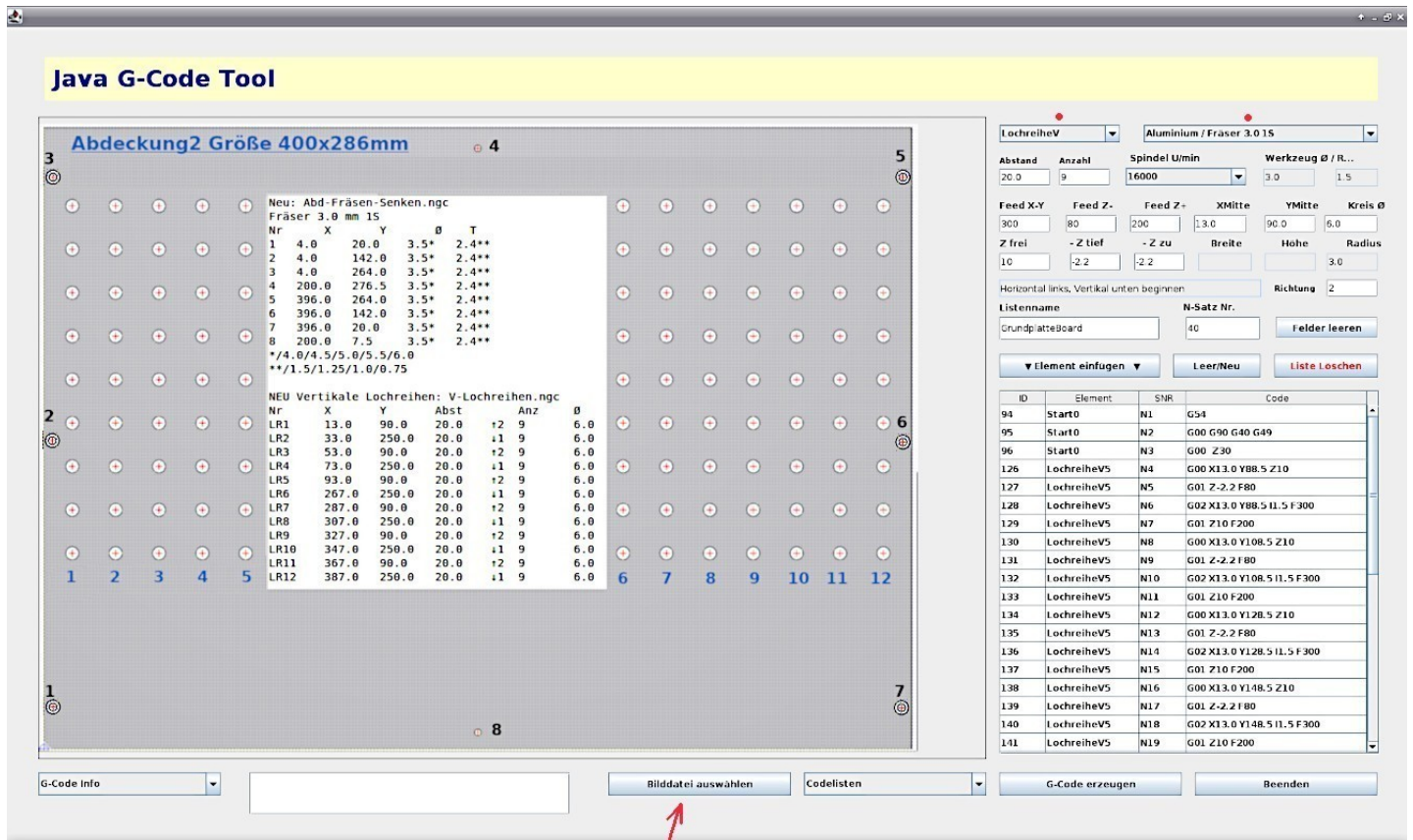


Das Java Gcode-Tool



Bilddatei auswählen

Unten kann mit **Bilddatei auswählen** eine Zeichnung eingeblendet werden, Code:

```

44 //*****
45 public static void choosePNG() throws InterruptedException {
46     Start.jLabel1.setIcon(null);
47     JFileChooser chooser = new JFileChooser();
48     FileFilter filter = new FileNameExtensionFilter("Bilder", "png");
49     chooser.setDialogTitle("Die Bilddatei des Projektes auswählen");
50     chooser.setCurrentDirectory(new File(home + "/GCodeDaten/codes/"));
51     //home/gottfried/GCodeDaten/codes/
52     chooser.addChoosableFileFilter(filter);
53     chooser.setFileFilter(filter);
54     int rueckgabeWert = chooser.showOpenDialog(null);
55     if (rueckgabeWert == JFileChooser.APPROVE_OPTION) {
56         String image = chooser.getSelectedFile().getAbsolutePath();
57         String imagedir = chooser.getSelectedFile().getParent();
58         Start.jLabel1.setIcon(new ImageIcon(image));
59     }
60 }

```

Danach sind oben rechts das zu programmierende Element und das Werkzeug auszuwählen. Erstes Element ist immer **Start**. (Code siehe Seite 4)
Wichtig ist auch die Eingabe von **Listenname** und **N-Satz-Nr** in der Mitte.

Listenname	N-Satz Nr.
GPGewinde	1

Element (Aufgabe) auswählen

GewindeM3 ▼

- 1 Element
- Start
- Bohren
- Teilkreis
- KreisFRK
- Rechteck
- RechteckFRK
- RETascheFRKH
- RETascheFRKV
- LinieH
- LinieV
- LochreiheH
- LochreiheV
- GewindeM3**
- GewindeM4
- Manuell
- NutenV
- Pause
- Ende

Wird in der ComboBox ein Element gewählt, werden die Eingabefelder mit dem Code **chooseElement()** diesem Element angepasst. Als Beispiel der Teil des Codes für **GewindeM3**. Der gesamte Code ist sehr lang.

```

405 //-----
406 if (("GewindeM3".equals(element)) || ("GewindeM4".equals(element))) {
407     Start.FXY.setText("150");
408     Start.FZminus.setText("150");
409     Start.FZplus.setText("");
410     Start.FZminus.setEditable(false);
411     Start.FZplus.setEditable(false);
412     Start.Zfrei.setText("5.0");
413     Start.ANZ.setText("");
414     Start.ANZ.setEditable(false);
415     Start.VAR.setText("");
416     Start.VAR.setEditable(false);
417     Start.WDM.setText("2.0");
418     Start.Wradius.setText("1.0");
419     Start.Ebreite.setText("");
420     Start.Ebreite.setEditable(false);
421     Start.Ehöhe.setText("");
422     Start.Ehöhe.setEditable(false);
423     Start.Ztief.setEditable(true);
424     Start.Zfrei.setEditable(true);
425     Start.Zzu.setEditable(false);
426     Start.Xmitte.setEditable(true);
427     Start.Xmitte.setText("Kernloch");
428     Start.Ymitte.setEditable(true);
429     Start.Ymitte.setText("Kernloch");
430     Start.jLabel19.setText("Hinweis");
431     if ("GewindeM3".equals(element)) {
432         Start.EDM.setText("3.0");
433         Start.Ztief.setText("-(KL+0.5)");
434         Start.Zzu.setText("-0.5");
435     }
436     if ("GewindeM4".equals(element)) {
437         Start.EDM.setText("4.0");
438         Start.Ztief.setText("-(KL+0.8)");
439         Start.Zzu.setText("-0.75");
440     }
441 }
442 }

```

Hier der für GewindeM3 angepassten Eingabebereich:

GewindeM3 ▼		Aluminium / Gewindefräser 2.0 ▼			
VAR-1	VAR-2	Spindel U/min		Werkzeug Ø / R...	
		20000 ▼		2.0	1.0
Feed X-Y	Feed Z-	Feed Z+	X-Mitte	Y-Mitte	Kreis Ø
150	150		Kernloch	Kernloch	3.0
Z frei	- Z tief	- Z zu	Breite	Höhe	Radius
5.0	-(KL+0.5)	-0.5			

Die Gewinde-Herstellung mit dem Gewindefräser wurde so programmiert, dass die gleichen Koordinaten einzugeben sind, wie beim Fräsen oder Bohren der Kernlöcher. Das Feld Tiefe **-Z tief** zeigt den Hinweis, dass eine 0.5mm geringere Tiefe einzugeben ist.

Code Beispiele Mit Klick auf einen der Codes wird die Seite aufgerufen

StartCode();	-der Start Code auf Seite 4
BohrenCode();	-der Bohren Code auf Seite 6
KreisCodeFRK();	-der Kreis Code mit Fräserradiuskorrektur Seite 8
RechteckCodeFRK();	-der Rechteck Code mit Fräserradiuskorrektur Seite 13
LinienCodeH();	-der Linien Code Horizontal Seite 17
ReihenCodeFRKV();	-der Lochreihen Code Vertikal Seite 20
GewindeFraesenCode();	-der Gewinde Fräsen Code Seite 23
endCode();	-der End Code auf Seite 27
ErstellenCode();	-der Erstellen Code (erstellt die .ngc Datei) Seite 28
Beispiele Gcode	-Beispiele aus GCode Dateien (Abschnitte) Seite 32

```

1 //*****
2 //die Startzeilen jeder GCode Datei werden immer als erstes erstellt
3 public static void startCode() {
4     pw = "*****";
5     Connection conn = null;
6     try {
7         Class.forName("org.postgresql.Driver");
8         conn = DriverManager.getConnection(jhpn, uname, pw);
9         Statement st = conn.createStatement();
10        String el1 = (String) Start.jComboBox1.getSelectedItemAt();
11        String mwz = (String) Start.jComboBox4.getSelectedItemAt();
12        String umin = (String) Start.jComboBox5.getSelectedItemAt();
13
14        String gcpos = Start.POS.getText();
15        String listname = Start.LIST.getText();
16        String zSIC = Start.Zfrei.getText();
17        String gcode = "";
18        //int vc = 0;
19        elnum = 0;
20        String num = (Integer.toString(elnum));
21        String el2 = (el1 + num);
22        //Satz 0 -----
23        String ngcpos = "%";
24        String el0 = "";
25        String kom = "";
26        String sSQL = null;
27        sSQL = "INSERT INTO codelist (gcpos, listname, element, gcode,      ↵
comment) " + "VALUES ( " + "'" + ngcpos + "'," + "'" + listname + "'," ↵
+ "'" + el0 + "',"
28            + "'" + gcode + "'," + "'" + kom + "'" + ")";
29        st.execute(sSQL);
30        //Satz 1 -----
31        ngcpos = "N" + gcpos;
32        gcode = "G54 G21";
33        kom = "(Nullpunkt, metrisch, " + mwz + ")";
34        sSQL = null;
35        sSQL = "INSERT INTO codelist (gcpos, listname, element, gcode,      ↵
comment) " + "VALUES ( " + "'" + ngcpos + "'," + "'" + listname + "'," ↵
+ "'" + el2 + "',"
36            + "'" + gcode + "'," + "'" + kom + "'" + ")";
37        st.execute(sSQL);
38        int ipos = Integer.parseInt(gcpos);
39        ipos = (ipos + 1);
40        npos = (Integer.toString(ipos));
41        Start.POS.setText(npos);
42        //Satz 2 -----
43        ngcpos = "N" + npos;
44        //String umin = Spindel.getText(); umin hier berechnen n=vcx1000/3,14xd
45        gcode = "G00 G90 G40 G49 M3 S" + umin;

```

```

46         kom = "(Eilgang, Absolutmaße, ohne FRK, ohne WLK, Spindelre)";
47
48         sSQL = "INSERT INTO  codelist (gcpos, listname, element, gcode,      ↗
comment) " + "VALUES ( " + "'" + ngcpos + "'," + "'" + listname + "',"      ↗
+ "'" + el2 + "',"
49             + "'" + gcode + "'," + "'" + kom + "'" + ")";
50         st.execute(sSQL);
51
52         gcpos = Start.POS.getText();
53         ipos = Integer.parseInt(gcpos);
54         ipos = (ipos + 1);
55         npos = (Integer.toString(ipos));
56         Start.POS.setText(npos);
57         //Satz 3 -----
58         ngcpos = "N" + npos;
59         gcode = "G00 " + "Z" + zSIC;
60         kom = "(Eilgang ZSIC)";
61         sSQL = "INSERT INTO  codelist (gcpos, listname, element, gcode,      ↗
comment) " + "VALUES ( " + "'" + ngcpos + "'," + "'" + listname + "',"      ↗
+ "'" + el2 + "',"
62             + "'" + gcode + "'," + "'" + kom + "'" + ")";
63         st.execute(sSQL);
64         gcpos = Start.POS.getText();
65         ipos = Integer.parseInt(gcpos);
66         ipos = (ipos + 1);
67         npos = (Integer.toString(ipos));
68         Start.POS.setText(npos);
69         conn.close();
70         aktualTable();
71     } catch (ClassNotFoundException | SQLException e) {
72         JOptionPane.showMessageDialog(null, "Fehler in startCode(): " + e);
73     }
74 }
75

```

```

1 //*****
2 // Bohrzyklus von LinuxCNC G83
3     public static void bohrenCode() {
4         pw = "*****";
5         Connection conn = null;
6         try {
7             Class.forName("org.postgresql.Driver");
8             conn = DriverManager.getConnection(jhpn, uname, pw);
9             Statement st = conn.createStatement();
10            String el1 = (String) Start.jComboBox1.getSelectedItemAt();
11            elnum = elnum + 1;
12            String num = (Integer.toString(elnum));
13            String el2 = (el1 + num);
14            String wdm = Start.WDM.getText();           //Werkzeug DM
15            String wrd = Start.Wradius.getText();       //Werkzeug Radius
16            Double dwr = Double.valueOf(wrd);
17            String Xmi = Start.Xmitte.getText();        //Pos X Mitte
18            String Ymi = Start.Ymitte.getText();        //Pos Y Mitte
19            String feedXY = Start.FXY.getText();
20            String feedZM = Start.FZminus.getText();
21            String feedZP = Start.FZplus.getText();
22            String zSIC = Start.Zfrei.getText();
23            //Double dzSic = Double.valueOf(zSIC);
24            String zAT = Start.Ztiefe.getText();
25            String zZU = Start.Zzu.getText();           //Zustellung Q
26            Double dZU = (-1 * Double.valueOf(zZU));
27            zZU = (Double.toString(dZU));
28            Double dR = (Double.valueOf(zSIC) / 5);     //Rückzugsebene oben
29            String sR = (Double.toString(dR));
30            String gcpos = Start.POS.getText();
31            String listname = Start.LIST.getText();
32            //Satz 0 -----
33
34            //Satz 1 -----
35            String ngcpos = "N" + gcpos;
36            String gcode = "G54 G21";
37            String kom = "(Nullpunkt, metrisch)";
38            String sSQL = null;
39            sSQL = "INSERT INTO codelist (gcpos, listname, element, gcode,      ↵
40            comment) " + "VALUES ( " + "'" + ngcpos + "'," + "'" + listname + "'," ↵
41            + "'" + el2 + "',"
42            + "'" + gcode + "'," + "'" + kom + "'" + ")";
43            st.execute(sSQL);
44            int ipos = Integer.parseInt(gcpos);
45            ipos = (ipos + 1);
46            npos = (Integer.toString(ipos));
47            Start.POS.setText(npos);
48            //Satz 2 -----
49            //G83 Tiefbohrzyklus X und Y = Bohrposition, Z = Bohrtiefe,

```

```

48      //Q=Zustellung je Durchlauf, R = Rückzugsebene, F = Vorschub
49      ngcpos = "N" + npos;
50      gcode = "G83 X" + Xmi + " Y" + Ymi + " Z" + zAT + " Q" + zZU + " R" +
      sR + " F" + feedZM;
51      kom = "(Tiefbohrzyklus mit R= zZU/2)";
52      sSQL = "INSERT INTO  codelist (gcpos, listname, element, gcode,
      comment) " + "VALUES ( " + "'" + ngcpos + "'," + "'" + listname + "',"
      + "'" + el2 + "',"
53          + "'" + gcode + "'," + "'" + kom + "'" + ")";
54      st.execute(sSQL);
55      gcpos = Start.POS.getText();
56      ipos = Integer.parseInt(gcpos);
57      ipos = (ipos + 1);
58      npos = (Integer.toString(ipos));
59      Start.POS.setText(npos);
60      //Satz 3 -----
61      ngcpos = "N" + npos;
62      gcode = "G80 G00 " + "Z" + zSIC;
63      kom = "(Eilgang ZSIC)";
64      sSQL = "INSERT INTO  codelist (gcpos, listname, element, gcode,
      comment) " + "VALUES ( " + "'" + ngcpos + "'," + "'" + listname + "',"
      + "'" + el2 + "',"
65          + "'" + gcode + "'," + "'" + kom + "'" + ")";
66      st.execute(sSQL);
67      gcpos = Start.POS.getText();
68      ipos = Integer.parseInt(gcpos);
69      ipos = (ipos + 1);
70      npos = (Integer.toString(ipos));
71      Start.POS.setText(npos);
72      conn.close();
73      aktualTable();
74  } catch (ClassNotFoundException | SQLException e) {
75      JOptionPane.showMessageDialog(null, "Fehler in startCode(): " + e);
76  }
77  }
78

```

```

1 //*****
2 //Kreis mit Fräserradiuskorrektur
3 public static void kreisCodeFRK() {
4     pw = "*****";
5     Connection conn = null;
6     try {
7         Class.forName("org.postgresql.Driver");
8         conn = DriverManager.getConnection(jhpn, uname, pw);
9         Statement st = conn.createStatement();
10        String el1 = (String) Start.jComboBox1.getSelectedItemAt();
11        elnum = elnum + 1;
12        String num = (Integer.toString(elnum));
13        String el2 = (el1 + num);
14        String wdm = Start.WDM.getText();           //Werkzeug DM
15        String wrd = Start.Wradius.getText();        //Werkzeug Radius
16        Double dwrđ = Double.valueOf(wrd);
17        String Xmi = Start.Xmitte.getText();         //Pos X Mitte
18        Double dXmi = Double.valueOf(Xmi);
19        String Ymi = Start.Ymitte.getText();         //Pos Y Mitte
20        String feedXY = Start.FXY.getText();
21        String feedZM = Start.FZminus.getText();
22        String feedZP = Start.FZplus.getText();
23        String zSIC = Start.Zfrei.getText();
24        String zAT = Start.Ztief.getText();           //Arbeitstiefe
25        Double dtief = Double.valueOf(zAT);
26        String zZU = Start.Zzu.getText();            //Zustellung
27        Double dzus0 = Double.valueOf(zZU);
28        String krd = Start.Kradius.getText();        // Kreis Radius
29        Double dkrd = Double.valueOf(krd);
30        String gcpos = Start.POS.getText();
31        String listname = Start.LIST.getText();
32        String gcode = "";
33        Double dXkorr = ((dXmi - dkrd) + dwrđ);
34        String Xkorr = (dXkorr.toString());
35        String senk = Start.VAR.getText();           //Senken: Ja Nein
36        //Satz 1 -----
37        //mit G00 zum korrigierten Startpunkt und auf zSIC fahren
38        String ngcpos = "N" + gcpos; //4
39        gcode = "G00 " + "X" + Xkorr + " Y" + Ymi + " Z" + zSIC;
40        String kom = "(Eilgang zum korrigierten Startpunkt " + el2 + " )";
41        String sSQL = null;
42        sSQL = "INSERT INTO codelist (gcpos, listname, element, gcode,      ↗
43        comment) " + "VALUES ( " + "'" + ngcpos + "'," + "'" + listname + "'," ↗
44        + "'" + el2 + "',"
45        + "'" + gcode + "'," + "'" + kom + "'" + ")";
46        st.execute(sSQL);
47        gcpos = Start.POS.getText();
48        int ipos = Integer.parseInt(gcpos);
49        ipos = (ipos + 1);

```

```

48      npos = (Integer.toString(ipos));
49      Start.POS.setText(npos);
50      //Satz 2 -----
51      //Eintauchen
52      Double dzus = dzus0;          //dzus = 1
53      while (dtief <= dzus) {        //Start Schleife Frästiefe/zustellg
54          zZU = (Double.toString(dzus));
55          ngcpos = "N" + npos; //6
56          gcode = "G01 " + "Z" + zZU + " F" + feedZM;
57          kom = "(Eintauchen)";
58          sSQL = "INSERT INTO codelist (gcpos, listname, element, gcode,  ↗
comment) " + "VALUES ( " + "'" + ngcpos + "','" + listname +  ↗
"'," + "'" + el2 + "','"
59              + "'" + gcode + "','" + kom + "'" + ")";
60          st.execute(sSQL);
61          gcpos = Start.POS.getText();
62          ipos = Integer.parseInt(gcpos);
63          ipos = (ipos + 1);
64          npos = (Integer.toString(ipos));
65          Start.POS.setText(npos);
66          //Satz 3 - hier erst zum korr Startpunkt fahren-----
67          // mit G02 den Kreisbogen mit FRK fräsen
68          ngcpos = "N" + npos; //8
69          Double dIkorr = ((dkrd) - (dwrdd));
70          String Ikorr = (dIkorr.toString());
71          gcode = "G02 " + "X" + Xkorr + " Y" + Ymi + " I" + Ikorr + " F" +  ↗
feedXY;
72          kom = "(Kreisbogen mit FRK)";
73          sSQL = "INSERT INTO codelist (gcpos, listname, element, gcode,  ↗
comment) " + "VALUES ( " + "'" + ngcpos + "','" + listname +  ↗
"'," + "'" + el2 + "','"
74              + "'" + gcode + "','" + kom + "'" + ")";
75          st.execute(sSQL);
76          gcpos = Start.POS.getText();
77          ipos = Integer.parseInt(gcpos);
78          ipos = (ipos + 1);
79          npos = (Integer.toString(ipos));
80          Start.POS.setText(npos);
81          dzus = (dzus + dzus0); // z.B. -2.0 + 1.0 = -1.0
82      } //end while Frästiefe
83      //Satz 4 SicPos einnehmen-----
84      ngcpos = "N" + npos; //9
85      gcode = "G01 " + "Z" + zSIC + " F" + feedZP;
86      kom = "(auf ZSIC fahren)";
87      sSQL = "INSERT INTO codelist (gcpos, listname, element, gcode,  ↗
comment) " + "VALUES ( " + "'" + ngcpos + "','" + listname + "','"  ↗
+ "'" + el2 + "','"
88          + "'" + gcode + "','" + kom + "'" + ")";
89      st.execute(sSQL);

```

```

90      gcpos = Start.POS.getText();
91      ipos = Integer.parseInt(gcpos);
92      ipos = (ipos + 1);
93      npos = (Integer.toString(ipos));
94      Start.POS.setText(npos);
95
96      if ("Ja".equals(senk)) {
97          //Satz 1 -----
98          //mit G01 zum zweiten Startpunkt fahren // Senken
99          dzus = -1.5;
100         dXkorr = (dXkorr - 0.25);
101         Double dIkorr = (0.75);
102
103         for (int i = 1; i < 5; i++) {
104             Xkorr = (dXkorr.toString());
105             ngcpos = "N" + gcpos; //4
106             gcode = "G01 " + "X" + Xkorr + " Y" + Ymi + " F" + feedZM;
107             kom = "(zum nächsten Startpunkt " + el2 + " )";
108             sSQL = null;
109             sSQL = "INSERT INTO codelist (gcpos, listname, element,
110                 gcode, comment) " + "VALUES ( " + "'" + ngcpos + "'," + "'" +
111                 listname + "'," + "'" + el2 + "',"
112                 + "'" + gcode + "'," + "'" + kom + "'" + ")";
113             st.execute(sSQL);
114             gcpos = Start.POS.getText();
115             ipos = Integer.parseInt(gcpos);
116             ipos = (ipos + 1);
117             npos = (Integer.toString(ipos));
118             Start.POS.setText(npos);
119             //Satz 2 -----
120             //Eintauchen
121             //dzus = dzus0;          //dzus = -1.5
122             zZU = (Double.toString(dzus));
123             ngcpos = "N" + npos; //6
124             gcode = "G01 " + "Z" + zZU + " F" + feedZM;
125             kom = "(Eintauchen)";
126             sSQL = "INSERT INTO codelist (gcpos, listname, element,
127                 gcode, comment) " + "VALUES ( " + "'" + ngcpos + "'," + "'" +
128                 listname + "'," + "'" + el2 + "',"
129                 + "'" + gcode + "'," + "'" + kom + "'" + ")";
130             st.execute(sSQL);
131             gcpos = Start.POS.getText();
132             ipos = Integer.parseInt(gcpos);
133             ipos = (ipos + 1);
134             npos = (Integer.toString(ipos));
135             Start.POS.setText(npos);
136             //Satz 3 - mit G02 den Kreisbogen mit FRK fräsen-----
137             ngcpos = "N" + npos; //8

```

```

135         String Ikorr = (dIkorr.toString());
136         gcode = "G02 " + "X" + Xkorr + " Y" + Ymi + " I" + Ikorr + " F" ↵
            + feedXY;
137         kom = "(Senken: Kreisbogen mit FRK)";
138         sSQL = "INSERT INTO  codelist (gcpos, listname, element, ↵
            gcode, comment) " + "VALUES ( " + "'" + ngcpos + "'," + "'" + ↵
            listname + "'," + "'" + el2 + "',"
139             + "'" + gcode + "'," + "'" + kom + "'" + ")";
140         st.execute(sSQL);
141         gcpos = Start.POS.getText();
142         ipos = Integer.parseInt(gcpos);
143         ipos = (ipos + 1);
144         npos = (Integer.toString(ipos));
145         Start.POS.setText(npos);
146         //Satz 4  Anheben-----
147         dzus = (dzus + 0.25);
148         zZU = (Double.toString(dzus));
149         ngcpos = "N" + npos; //9
150         gcode = "G01 " + "Z" + zZU + " F" + feedZP;
151         kom = "(auf nächste Tiefe fahren)";
152         sSQL = "INSERT INTO  codelist (gcpos, listname, element, ↵
            gcode, comment) " + "VALUES ( " + "'" + ngcpos + "'," + "'" + ↵
            listname + "'," + "'" + el2 + "',"
153             + "'" + gcode + "'," + "'" + kom + "'" + ")";
154         st.execute(sSQL);
155         gcpos = Start.POS.getText();
156         ipos = Integer.parseInt(gcpos);
157         ipos = (ipos + 1);
158         npos = (Integer.toString(ipos));
159         Start.POS.setText(npos);
160
161         dXkorr = (dXkorr - 0.25);
162         dIkorr = (dIkorr + 0.25);
163         // z.B. -1.5 + 0.25 = -1.25
164     } //end for Schleife
165     //Satz 4  SicPos einnehmen-----
166     ngcpos = "N" + npos; //9
167     gcode = "G01 " + "Z" + zSIC + " F" + feedZP;
168     kom = "(auf ZSIC fahren)";
169     sSQL = "INSERT INTO  codelist (gcpos, listname, element, gcode, ↵
        comment) " + "VALUES ( " + "'" + ngcpos + "'," + "'" + listname + ↵
        "'" + el2 + "',"
170         + "'" + gcode + "'," + "'" + kom + "'" + ")";
171     st.execute(sSQL);
172     gcpos = Start.POS.getText();
173     ipos = Integer.parseInt(gcpos);
174     ipos = (ipos + 1);
175     npos = (Integer.toString(ipos));
176     Start.POS.setText(npos);

```

```
177
178         }
179         conn.close();
180         aktualTable();
181     } catch (ClassNotFoundException | SQLException e) {
182         JOptionPane.showMessageDialog(null, "Fehler in kreisCode(): " + e);
183     }
184
185 }
186
```

```

1 //*****
2 //Rechteck mit Fräserradiuskorrektur (rechts der Kontur, Fräser innen im Rechteck)
3 public static void rechteckCodeFRK() {
4     pw = "*****";
5     Connection conn = null;
6     try {
7         Class.forName("org.postgresql.Driver");
8         conn = DriverManager.getConnection(jhpn, uname, pw);
9         Statement st = conn.createStatement();
10        String el1 = (String) Start.jComboBox1.getSelectedItem();
11        elnum = elnum + 1;
12        String num = (Integer.toString(elnum));
13        String el2 = (el1 + num);
14        String wdm = Start.WDM.getText();           //Werkzeug DM
15        String wr = Start.Wradius.getText();         //Werkzeug Radius
16        double dwr = (Double.valueOf(wr));
17        String elb = Start.Ebreite.getText();
18        double elb2 = (Double.valueOf(elb) / 2); // Elementbreite/2
19        String elh = Start.Ehöhe.getText();
20        double elh2 = (Double.valueOf(elh) / 2); // Elementhöhe/2
21        String Xmi = Start.Xmitte.getText();         //Pos X Mitte
22        double dXmi = (Double.valueOf(Xmi));
23        String Ymi = Start.Ymitte.getText();         //Pos Y Mitte
24        double dYmi = Double.valueOf(Ymi);
25        String feedXY = Start.FXY.getText();
26        String feedZM = Start.FZminus.getText();
27        String feedZP = Start.FZplus.getText();
28        String zSIC = Start.Zfrei.getText();
29        String zAT = Start.Ztief.getText();           //Arbeitstiefe
30        Double dtief = Double.valueOf(zAT);
31        String zZU = Start.Zzu.getText();             //Zustellung
32        Double dzus0 = Double.valueOf(zZU);
33        String krd = Start.Kradius.getText();         // Kreis Radius
34        String listname = Start.LIST.getText();
35        String kontur = Start.VAR.getText();
36        String gcpos = Start.POS.getText();
37        String gcode = "";
38        //Satz 1 -----
39        //mit G00 zur Startposition und auf zSIC fahren(Recteck unten links)
40        String ngcpos = "N" + gcpos;
41        double xpos = 0;
42        double ypos = 0;
43        if ("A".equals(kontur)) {
44            xpos = ((dXmi - elb2) - dwr);             //((50-50) -1.5) = -1.5
45            ypos = ((dYmi - elh2) - dwr);             //((25-25) -1.5) = -1.5
46        }
47        if ("I".equals(kontur)) {
48            xpos = ((dXmi - elb2) + dwr);
49            ypos = ((dYmi - elh2) + dwr);

```

```

50     }
51     String Xpos = (Double.toString(xpos));
52     String Ypos = (Double.toString(ypos));
53     ngcpos = "N" + npos;
54     String sSQL = "";
55     gcode = "G00 " + "X" + Xpos + " Y" + Ypos + " Z" + zSIC;
56     String kom = "(Eilgang zum korrigierten Startpunkt " + el2 + " )";
57
58     sSQL = "INSERT INTO  codelist (gcpos, listname, element, gcode,      ↗
comment) " + "VALUES ( " + "'" + ngcpos + "'," + "'" + listname + "'," ↗
+ "'" + el2 + "',"
59         + "'" + gcode + "'," + "'" + kom + "'" + ")";
60     st.execute(sSQL);
61
62     gcpos = Start.POS.getText();
63     int ipos = Integer.parseInt(gcpos);
64     ipos = (ipos + 1);
65     npos = (Integer.toString(ipos));
66     Start.POS.setText(npos);
67     //Satz 2 eintauchen-----
68     Double dzus = dzus0;
69     while (dtief <= dzus) {
70         zZU = (Double.toString(dzus));
71         ngcpos = "N" + npos;
72         gcode = "G01 " + "Z" + zZU + " F" + feedZM;
73         kom = "(Eintauchen mit feedZM)";
74         sSQL = "INSERT INTO  codelist (gcpos, listname, element, gcode,      ↗
comment) " + "VALUES ( " + "'" + ngcpos + "'," + "'" + listname +      ↗
+ "'" + el2 + "',"
75         + "'" + gcode + "'," + "'" + kom + "'" + ")";
76         st.execute(sSQL);
77         gcpos = Start.POS.getText();
78         ipos = Integer.parseInt(gcpos);
79         ipos = (ipos + 1);
80         npos = (Integer.toString(ipos));
81         Start.POS.setText(npos);
82
83         //Satz 3 mit feedXY bis zur Oberkante des Rechtecks fräsen ----
84         ngcpos = "N" + npos;
85         if ("I".equals(kontur)) {
86             ypos = ((dYmi + elh2) - dwr);    // ((25 + 25) -1.5) = 48.5
87         }
88         if ("A".equals(kontur)) {
89             ypos = (dYmi + elh2 + dwr);    // (25 + 25 + 1.5) = 51.5
90         }
91         Ypos = (Double.toString(ypos));
92         gcode = "G01 " + " Y" + Ypos + " F" + feedXY;
93         kom = "(mit feedXY bis oben links)";
94         sSQL = "INSERT INTO  codelist (gcpos, listname, element, gcode,      ↗

```

```

comment) " + "VALUES ( " + "'" + ngcpos + "','" + "'" + listname +
"',," + "'" + el2 + "','"
95         + "'" + gcode + "','" + "'" + kom + "'" + ")";
96 st.execute(sSQL);
97 gcpos = Start.POS.getText();
98 ipos = Integer.parseInt(gcpos);
99 ipos = (ipos + 1);
100 npos = (Integer.toString(ipos));
101 Start.POS.setText(npos);
102 //Satz 4 bis zur rechten oberen Ecke fräsen -----
103 ngcpos = "N" + npos;
104 if ("A".equals(kontur)) {
105     xpos = (dXmi + elb2 + dwr);    // (50 + 50 + 1.5) = 101.5
106 }
107 if ("I".equals(kontur)) {
108     xpos = ((dXmi + elb2) - dwr);    // ((50+50)-1.5) = 98.5
109 }
110 Xpos = (Double.toString(xpos));
111 gcode = "G01 " + "X" + Xpos;
112 kom = "(mit feedXY bis oben rechts)";
113 sSQL = "INSERT INTO codelist (gcpos, listname, element, gcode,
comment) " + "VALUES ( " + "'" + ngcpos + "','" + "'" + listname +
"',," + "'" + el2 + "','"
114         + "'" + gcode + "','" + "'" + kom + "'" + ")";
115 st.execute(sSQL);
116 gcpos = Start.POS.getText();
117 ipos = Integer.parseInt(gcpos);
118 ipos = (ipos + 1);
119 npos = (Integer.toString(ipos));
120 Start.POS.setText(npos);
121 //Satz 5 bis zur rechten unteren Ecke fräsen -----
122 ngcpos = "N" + npos;
123 if ("A".equals(kontur)) {
124     ypos = (dYmi - (elh2 + dwr));    // (25 - (25+1,5)) = -1.5
125 }
126 if ("I".equals(kontur)) {
127     ypos = (dYmi - (elh2 - dwr));    // (25 - (25 - 1.5) = 1.5
128 }
129 Ypos = (Double.toString(ypos));
130 gcode = "G01 " + "Y" + Ypos;
131 kom = "(mit feedXY bis unten rechts)";
132 sSQL = "INSERT INTO codelist (gcpos, listname, element, gcode,
comment) " + "VALUES ( " + "'" + ngcpos + "','" + "'" + listname +
"',," + "'" + el2 + "','"
133         + "'" + gcode + "','" + "'" + kom + "'" + ")";
134 st.execute(sSQL);
135 gcpos = Start.POS.getText();
136 ipos = Integer.parseInt(gcpos);
137 ipos = (ipos + 1);

```

```

138         npos = (Integer.toString(ipos));
139         Start.POS.setText(npos);
140         //Satz 6 // bis zur linken unteren Ecke (startpunkt)fräsen ----
141         ngcpos = "N" + npos;
142         if ("A".equals(kontur)) {
143             xpos = (dXmi - (elb2 + dwr));           // (50 - (50 + 1.5)) = -1.5
144         }
145         if ("I".equals(kontur)) {
146             xpos = ((dXmi - elb2) + dwr);           // ((50 -50) +1.5) = 1.5
147         }
148         Xpos = (Double.toString(xpos));
149         gcode = "G01 " + "X" + Xpos;
150         kom = "(mit feedXY bis unten links)";
151
152         sSQL = "INSERT INTO  codelist (gcpos, listname, element, gcode,      ↗
comment) " + "VALUES ( " + "'" + ngcpos + "'," + "'" + listname +      ↗
"''," + "'" + el2 + "',"
153             + "'" + gcode + "'," + "'" + kom + "'" + ")";
154         st.execute(sSQL);
155         gcpos = Start.POS.getText();
156         ipos = Integer.parseInt(gcpos);
157         ipos = (ipos + 1);
158         npos = (Integer.toString(ipos));
159         Start.POS.setText(npos);
160         dzus = (dzus + dzus0);
161     } //end while
162     //Satz 7 Sicherheitshöhe und Vorschub Zhoch -----
163     ngcpos = "N" + npos;
164     gcode = "G01 " + "Z" + zSIC + " F" + feedZP;
165     kom = "(mit feedZP auf ZSIC)";
166     sSQL = "INSERT INTO  codelist (gcpos, listname, element, gcode,      ↗
comment) " + "VALUES ( " + "'" + ngcpos + "'," + "'" + listname +      ↗
"''," + "'" + el2 + "',"
167         + "'" + gcode + "'," + "'" + kom + "'" + ")";
168     st.execute(sSQL);
169     gcpos = Start.POS.getText();
170     ipos = Integer.parseInt(gcpos);
171     ipos = (ipos + 1);
172     npos = (Integer.toString(ipos));
173     Start.POS.setText(npos);
174     conn.close();
175     aktualTable();
176 } catch (ClassNotFoundException | SQLException e) {
177     JOptionPane.showMessageDialog(null, "Fehler in rechteckCode(): " + e);
178 }
179 }
180

```

```

1 //*****
2 /* Automatischer Richtungswechsel bis Tiefe erreicht ist, beginnend links
3 */
4 public static void linienCodeH() {
5     pw = "*****";
6     Connection conn = null;
7     String ri = Start.VAR.getText();
8     int i = Integer.parseInt(ri);
9     try {
10         Class.forName("org.postgresql.Driver");
11         conn = DriverManager.getConnection(jhpn, uname, pw);
12         Statement st = conn.createStatement();
13         String el1 = (String) Start.jComboBox1.getSelectedItem();
14         elnum = elnum + 1;
15         String num = (Integer.toString(elnum));
16         String el2 = (el1 + num);
17         String wdm = Start.WDM.getText();           //Werkzeug DM
18         String wr = Start.Wradius.getText();         //Werkzeug Radius
19         double dwr = (Double.valueOf(wr));
20         String elb = Start.Ebreite.getText();
21         double elb2 = (Double.valueOf(elb)); // Elementbreite
22         String elh = Start.Ehöhe.getText();
23         double elh2 = (Double.valueOf(elh)); // Elementhöhe
24         String Xstart = Start.Xmitte.getText();      //Pos X Mitte
25         double dXstart = (Double.valueOf(Xstart));
26         String Ystart = Start.Ymitte.getText();      //Pos Y Mitte
27         double dYstart = Double.valueOf(Ystart);
28         String feedXY = Start.FXY.getText();
29         String feedZM = Start.FZminus.getText();
30         String feedZP = Start.FZplus.getText();
31         String zSIC = Start.Zfrei.getText();
32         String zAT = Start.Ztief.getText();          //Arbeitstiefe
33         Double dtief = Double.valueOf(zAT);
34         String zZU = Start.Zzu.getText();            //Zustellung
35         Double dzus0 = Double.valueOf(zZU);         //Zustellung
36         String krd = Start.Kradius.getText();       // Kreis Radius
37         String listname = Start.LIST.getText();
38         String gcpos = Start.POS.getText();
39         String vers = Start.ANZ.getText(); //Ziel-Versatz bei schräger Fahrt
40         double dvers = Double.valueOf(vers);
41         Double dzus = dzus0;
42         String gcode = "";
43         //Satz 1 zur Startposition und auf zSIC fahren, Linienbeginn -----
44         //+ oder minus, nach rechts oder links
45         gcpos = Start.POS.getText();
46         double xpos = 0;
47         if (i % 2 == 0) {
48             xpos = (dXstart - dwr); //nach rechts, Startpos links - dwr
49         }

```

```

50     if (i % 2 != 0) {
51         xpos = (dXstart + dwr);    //nach links, Startpos rechts + dwr
52     } // li 100+(100+1) 201
53     String gcpos = "N" + gcpos;
54     String Xpos = (Double.toString(xpos));
55     double ypos = dYstart;
56     String Ypos = (Double.toString(ypos));
57     ngcpos = "N" + npos;
58     String sSQL = "";
59     gcode = "G00 " + "X" + Xpos + " Y" + Ypos + " Z" + zSIC;
60     String kom = "(Eilgang zur Startposition und ZSIC)";
61     sSQL = "INSERT INTO codelist (gcpos, listname, element, gcode,      ↗
comment) " + "VALUES ( " + "'" + ngcpos + "'," + "'" + listname + "'," ↗
+ "'" + el2 + "',"
62         + "'" + gcode + "'," + "'" + kom + "'" + ")";
63     st.execute(sSQL);
64     gcpos = Start.POS.getText();
65     int ipos = Integer.parseInt(gcpos);
66     ipos = (ipos + 1);
67     npos = (Integer.toString(ipos));
68     Start.POS.setText(npos);
69     //Satz 2 eintauchen-----
70     while (dtief <= dzus) {
71         zZU = (Double.toString(dzus));
72         ngcpos = "N" + npos;
73         gcode = "G01 " + "Z" + zZU + " F" + feedZM;
74         kom = "(Eintauchen)";
75         sSQL = "INSERT INTO codelist (gcpos, listname, element, gcode,      ↗
comment) " + "VALUES ( " + "'" + ngcpos + "'," + "'" + listname +      ↗
+ "'" + el2 + "',"
76         + "'" + gcode + "'," + "'" + kom + "'" + ")";
77         st.execute(sSQL);
78         gcpos = Start.POS.getText();
79         ipos = Integer.parseInt(gcpos);
80         ipos = (ipos + 1);
81         npos = (Integer.toString(ipos));
82         Start.POS.setText(npos);
83         //Satz 3 bis zum Lienienende fräsen-----
84         ngcpos = "N" + npos;
85         if (i % 2 == 0) {
86             xpos = (dXstart + elb2 + dwr);    //re (100 +100 + 1) 201
87             kom = "(nach rechts fräsen)";
88         }
89         if (i % 2 != 0) {
90             xpos = ((dXstart) - (dwr)); // li (100-(100+1) -1
91             kom = "(nach links fräsen)";
92         }
93         Xpos = (Double.toString(xpos));
94         gcode = "G01 " + " X" + (Xpos) + " Y" + Ypos + " F" + feedXY;

```

```

95         sSQL = "INSERT INTO  codelist (gcpos, listname, element, gcode,      ↵
           comment) " + "VALUES ( " + "'" + ngcpos + "'," + "'" + listname +      ↵
           "'" + el2 + "',"
96             + "'" + gcode + "'," + "'" + kom + "'" + ")";
97         st.execute(sSQL);
98         gcpos = Start.POS.getText();
99         ipos = Integer.parseInt(gcpos);
100        ipos = (ipos + 1);
101        npos = (Integer.toString(ipos));
102        Start.POS.setText(npos);
103        dzus = (dzus + dzus0);
104        i = (i + 1);
105    } //end while
106    String as = Start.VAR3.getText();
107    if ("4".equals(as)) {
108        //Satz 4 Sicherheitshöhe und Vorschub Zhoch-----
109        ngcpos = "N" + npos;
110        gcode = "G01 " + "Z" + zSIC + " F" + feedZP;
111        kom = "(mit feedZP auf ZSIC)";
112        sSQL = "INSERT INTO  codelist (gcpos, listname, element, gcode,      ↵
           comment) " + "VALUES ( " + "'" + ngcpos + "'," + "'" + listname +      ↵
           "'" + el2 + "',"
113            + "'" + gcode + "'," + "'" + kom + "'" + ")";
114        st.execute(sSQL);
115        gcpos = Start.POS.getText();
116        ipos = Integer.parseInt(gcpos);
117        ipos = (ipos + 1);
118        npos = (Integer.toString(ipos));
119        Start.POS.setText(npos);
120        conn.close();
121    }
122    aktualTable();
123    } catch (ClassNotFoundException | SQLException e) {
124        JOptionPane.showMessageDialog(null, "Fehler in rechteckCode(): " + e);
125    }
126 }
127

```

```

1 //*****
2 /* Vertikale Lochreihen, Richtung durch Eingabe wählbar
3    vermeidet lange Leerfahrten*/
4 public static void reihenCodeFRKV() {
5     pw = "*****";
6     Connection conn = null;
7     try {
8         Class.forName("org.postgresql.Driver");
9         conn = DriverManager.getConnection(jhpn, uname, pw);
10        Statement st = conn.createStatement();
11        String ell = (String) Start.jComboBox1.getSelectedItem(); //element
12        elnum = elnum + 1; //Start mit 1 (0+1)
13        String num = (Integer.toString(elnum));
14        String el2 = (ell + num);
15        String wdm = Start.WDM.getText(); //Werkzeug DM
16        String wrd = Start.Wradius.getText(); //Werkzeug Radius
17        Double dwrd = Double.valueOf(wrd);
18        String Xmi = Start.Xmitte.getText(); //Pos X Mitte
19        String Ymi = Start.Ymitte.getText(); //Pos Y Mitte
20        Double dYmi = Double.valueOf(Ymi);
21        String feedXY = Start.FXY.getText();
22        String feedZM = Start.FZminus.getText();
23        String feedZP = Start.FZplus.getText();
24        String zSIC = Start.Zfrei.getText();
25        String zAT = Start.Ztief.getText(); //Arbeitstiefe
26        Double dtief = Double.valueOf(zAT);
27        String zZU = Start.Zzu.getText(); //Zustellung
28        Double dzus0 = Double.valueOf(zZU);
29        String krd = Start.Kradius.getText(); // Kreis Radius
30        Double dkrd = Double.valueOf(krd);
31        String gcpos = Start.POS.getText();
32        npos = gcpos;
33        String listname = Start.LIST.getText();
34        String anz = Start.ANZ.getText(); //Lochreihe Anz
35        Double danz = Double.valueOf(anz);
36        String abst = Start.VAR.getText(); //Lochreihe Abst
37        Double dabst = Double.valueOf(abst);
38        Double dYkorrr = ((dYmi - dkrd) + dwrd);
39        String ri = Start.VAR3.getText();
40        String gcode = "";
41        //weiterzählen der Anzahl mit for danz = double Anzahl
42        for (int i = 0; i < danz; i++) {
43            String Ykorrr = (Double.toString(dYkorrr)); //8.25
44            //Satz 1 zum korrigierten XStartpunkt und Y fahren, dazu auf zSIC
45            String ngcpos = "N" + npos; //4
46            gcode = "G00 " + "X" + Xmi + " Y" + Ykorrr + " Z" + zSIC;
47            String sSQL = null;
48            String kom = "(Eilgang, zum korrigierten Startpunkt und auf zSIC
                fahren)";

```

```

49      sSQL = "INSERT INTO  codelist (gcpos, listname, element, gcode,      ↵
comment) " + "VALUES ( " + "'" + ngcpos + "'," + "'" + listname +      ↵
"'," + "'" + el2 + "',"
50          + "'" + gcode + "'," + "'" + kom + "'" + ")";
51      st.execute(sSQL);
52      gcpos = Start.POS.getText();
53      int ipos = Integer.parseInt(gcpos);
54      ipos = (ipos + 1);
55      npos = (Integer.toString(ipos));
56      Start.POS.setText(npos);
57      //Satz 2 Eintauchen-----
58      Double dzus = dzus0;
59      while (dtief <= dzus) {                                     // -2.0 <= 1.0
60          zZU = (Double.toString(dzus));
61          ngcpos = "N" + npos; //6
62          gcode = "G01 " + "Z" + zZU + " F" + feedZM;
63          kom = "(Eintauchen)";
64          sSQL = "INSERT INTO  codelist (gcpos, listname, element,      ↵
gcode, comment) " + "VALUES ( " + "'" + ngcpos + "'," + "'" +      ↵
listname + "'," + "'" + el2 + "',"
65              + "'" + gcode + "'," + "'" + kom + "'" + ")";
66          st.execute(sSQL);
67          gcpos = Start.POS.getText();
68          ipos = Integer.parseInt(gcpos);
69          ipos = (ipos + 1);
70          npos = (Integer.toString(ipos));
71          Start.POS.setText(npos);
72          // hier erst zum korr Startpunkt fahren
73          //Satz 3 mit G02 den Kreisbogen mit FRK fräsen-----
74          ngcpos = "N" + npos; //8
75          Double dIkorr = ((dkrd) - (dwrđ));
76          String Ikorr = (dIkorr.toString());
77          gcode = "G02 " + "X" + Xmi + " Y" + Ykorr + " I" + Ikorr + " F" ↵
+ feedXY;
78          kom = "(Kreisbogen mit FRK)";
79          sSQL = "INSERT INTO  codelist (gcpos, listname, element,      ↵
gcode, comment) " + "VALUES ( " + "'" + ngcpos + "'," + "'" +      ↵
listname + "'," + "'" + el2 + "',"
80              + "'" + gcode + "'," + "'" + kom + "'" + ")";
81          st.execute(sSQL);
82          gcpos = Start.POS.getText();
83          ipos = Integer.parseInt(gcpos);
84          ipos = (ipos + 1);
85          npos = (Integer.toString(ipos));
86          Start.POS.setText(npos);
87          dzus = (dzus + dzus0);
88          dzus = round(dzus, 2);
89          //JOptionPane.showMessageDialog(null, "dzus: " + dzus);
90      } //end while

```

```

91 //Satz 4 SicPos einnehmen-----
92 ngcpos = "N" + npos; //9
93 gcode = "G01 " + "Z" + zSIC + " F" + feedZP;
94 kom = "(auf ZSIC fahren)";
95 sSQL = "INSERT INTO codelist (gcpos, listname, element, gcode,
comment) " + "VALUES ( " + "'" + ngcpos + "'," + "'" + listname +
"'," + "'" + el2 + "',"
96 + "'" + gcode + "'," + "'" + kom + "'" + ")";
97 st.execute(sSQL);
98 gcpos = Start.POS.getText();
99 ipos = Integer.parseInt(gcpos);
100 ipos = (ipos + 1);
101 npos = (Integer.toString(ipos));
102 Start.POS.setText(npos); //10
103 if ("2".equals(ri)) {
104     dYkorrr = (dYkorrr + dabst);
105 }
106 if ("1".equals(ri)) {
107     dYkorrr = (dYkorrr - dabst);
108 }
109 } //end while
110 conn.close();
111 aktualTable();
112 } catch (ClassNotFoundException | SQLException e) {
113     JOptionPane.showMessageDialog(null, "Fehler in kreisCode(): " + e);
114 }
115 }
116

```

```

1 //*****
2 public static void gewindeFraesenCode() {
3     pw = "*****";
4     Connection conn = null;
5     try {
6         Class.forName("org.postgresql.Driver");
7         conn = DriverManager.getConnection(jhpn, uname, pw);
8         Statement st = conn.createStatement();
9         String el1 = (String) Start.jComboBox1.getSelectedItem();
10        String umin = (String) Start.jComboBox5.getSelectedItem();
11        elnum = elnum + 1;
12        String num = (Integer.toString(elnum));
13        String el2 = (el1 + num);
14        String gcpos = Start.POS.getText();
15        String listname = Start.LIST.getText();
16        String gcode = "";
17        //Satz 1 die Position zum Gewindefräsen anfahren-----
18        String sSQL = null;
19        String ngcpos = "N" + gcpos;
20        String Xmi = Start.Xmitte.getText();
21        String Ymi = Start.Ymitte.getText();
22        double dYmi = (Double.valueOf(Ymi));
23        String zSIC = Start.Zfrei.getText();
24        double dzSIC = (Double.valueOf(zSIC));
25        double end = (dzSIC / 5);
26        String zTF = Start.Ztief.getText();
27        String feedZM = Start.FZminus.getText();
28        String feedXY = Start.FXY.getText();
29        String ERd = Start.Kradius.getText();
30        double dERd = (Double.valueOf(ERd));
31        String WRd = Start.Wradius.getText();
32        double dWRd = (Double.valueOf(WRd));
33        double diff = (dERd - dWRd);
34        double Yplus = (dYmi + diff);
35        double Ymin = (dYmi - diff);
36        String YPLUS = (Double.toString(Yplus));
37        String YMIN = (Double.toString(Ymin));
38        double dJ = 0.0;
39        gcode = "G00 X" + Xmi + " Y" + Ymi;
40        String kom = "(Position zum Gewinde fräsen anfahren)";
41        sSQL = "INSERT INTO codelist (gcpos, listname, element, gcode,      ↵
42        comment) " + "VALUES ( " + "'" + ngcpos + "'," + "'" + listname + "'," ↵
43        + "'" + el2 + "',"
44        + "'" + gcode + "'," + "'" + kom + "'" + ")";
45        st.execute(sSQL);
46        gcpos = Start.POS.getText();
47        int ipos = Integer.parseInt(gcpos);
48        ipos = (ipos + 1);
49        npos = (Integer.toString(ipos));

```

```

48      Start.POS.setText(npos);
49      //Satz 2 im Kernloch auf Tiefe absenken-----
50      ngcpos = "N" + npos;
51      //String Zzu = Start.Zzu.getText();
52      gcode = "G01" + " Z" + zTF + " F" + feedZM;
53      kom = "(im Kernloch auf " + zTF + "mm absenken)";
54      sSQL = "INSERT INTO  codelist (gcpos, listname, element, gcode,      ↵
comment) " + "VALUES ( " + "'" + ngcpos + "','" + listname + "','" ↵
+ "'" + el2 + "','"
55              + "'" + gcode + "','" + kom + "'" + ")";
56      st.execute(sSQL);
57      gcpos = Start.POS.getText();
58      ipos = Integer.parseInt(gcpos);
59      ipos = (ipos + 1);
60      npos = (Integer.toString(ipos));
61      Start.POS.setText(npos);
62      //Satz 3 Fräsen Start zB G01 F150 X10 Y20.5-----
63      ngcpos = "N" + npos;
64      gcode = "G01" + " F" + feedXY + " X" + Xmi + " Y" + YPLUS;
65      kom = "(Startdaten mit Differenz WZradius ELradius)";
66      sSQL = "INSERT INTO  codelist (gcpos, listname, element, gcode,      ↵
comment) " + "VALUES ( " + "'" + ngcpos + "','" + listname + "','" ↵
+ "'" + el2 + "','"
67              + "'" + gcode + "','" + kom + "'" + ")";
68      st.execute(sSQL);
69      gcpos = Start.POS.getText();
70      ipos = Integer.parseInt(gcpos);
71      ipos = (ipos + 1);
72      npos = (Integer.toString(ipos));
73      Start.POS.setText(npos);
74      //Satz 4 Inkrementeller Entfernungsmodus G91.1-----
75      ngcpos = "N" + npos;
76      gcode = "G91.1";
77      kom = "(Inkrementeller Entfernungsmodus)";
78      sSQL = "INSERT INTO  codelist (gcpos, listname, element, gcode,      ↵
comment) " + "VALUES ( " + "'" + ngcpos + "','" + listname + "','" ↵
+ "'" + el2 + "','"
79              + "'" + gcode + "','" + kom + "'" + ")";
80      st.execute(sSQL);
81      gcpos = Start.POS.getText();
82      ipos = Integer.parseInt(gcpos);
83      ipos = (ipos + 1);
84      npos = (Integer.toString(ipos));
85      Start.POS.setText(npos);
86      //Satz 5 bis Ende: Gewindefräsen Halbkreise i=Wechselzähler auf/ab--
87      double dZTF = (Double.valueOf(zTF)); // -5.0
88      int i = 1;
89      String sY = "";
90      String sJ = "";

```

```

91         while (dZTF < end) { //dZTF ist Gew.Tiefe end oben
92             gcpos = Start.POS.getText();
93             ngcpos = "N" + gcpos;
94             dZTF = (dZTF + 0.25);
95             dZTF = round(dZTF, 2);
96             zTF = (Double.toString(dZTF));
97             if (i % 2 == 0) { //Zähler i = gerade Zahl
98                 dJ = 0.5;
99                 sJ = (Double.toString(dJ));
100                 sY = YPLUS;
101             }
102             if (i % 2 != 0) { //zähler i = ungerade Zahl
103                 dJ = -0.5;
104                 sJ = (Double.toString(dJ));
105                 sY = YMIN;
106             }
107             gcode = "G03 X" + Xmi + " Y" + sY + " Z" + zTF + " I0 J" + sJ + " F" + feedXY;
108             kom = "(Gewinde fräsen)";
109             sSQL = "INSERT INTO codelist (gcpos, listname, element, gcode, comment) " + "VALUES ( " + "'" + ngcpos + "'," + "'" + listname + "'," + "'" + el2 + "'," + "'" + gcode + "'," + "'" + kom + "'" + ")";
110             st.execute(sSQL);
111             gcpos = Start.POS.getText();
112             ipos = Integer.parseInt(gcpos);
113             ipos = (ipos + 1);
114             npos = (Integer.toString(ipos));
115             Start.POS.setText(npos);
116             i = i + 1;
117         } //end while
118         //Satz 6 SicPos einnehmen-----
119         ngcpos = "N" + npos; //9
120         gcode = "G01 " + "Z" + zSIC + " F" + feedXY;
121         kom = "(auf ZSIC fahren)";
122         sSQL = "INSERT INTO codelist (gcpos, listname, element, gcode, comment) " + "VALUES ( " + "'" + ngcpos + "'," + "'" + listname + "'," + "'" + el2 + "'," + "'" + gcode + "'," + "'" + kom + "'" + ")";
123         st.execute(sSQL);
124         gcpos = Start.POS.getText();
125         ipos = Integer.parseInt(gcpos);
126         ipos = (ipos + 1);
127         npos = (Integer.toString(ipos));
128         Start.POS.setText(npos);
129         conn.close();
130         aktualTable();
131     } catch (ClassNotFoundException | SQLException e) {
132         JOptionPane.showMessageDialog(null, "Fehler in gewindeCode(): " + e);

```

135

136

137

}

}

```

1 //*****
2 // End-Code, immer nach Abschluss der zu fräsenden Elemente einzugeben
3 public static void endCode() {
4     pw = "*****";
5     Connection conn = null;
6     try {
7         Class.forName("org.postgresql.Driver");
8         conn = DriverManager.getConnection(jhpn, uname, pw);
9         Statement st = conn.createStatement();
10        String el1 = (String) Start.jComboBox1.getSelectedItemAt();
11        elnum = elnum + 1;
12        String num = (Integer.toString(elnum));
13        String el2 = (el1 + num);
14        String gcpos = Start.POS.getText();
15        String listname = Start.LIST.getText();
16        String zSIC = Start.Zfrei.getText();
17        gcpos = Start.POS.getText();
18        String gcode = "";
19        String xyz = JOptionPane.showInputDialog("Positionen für X, Y, Z am Ende eingeben:", "X-50 Y00 Z10");
20        if (xyz == null) {
21            return;
22        }
23        //Satz 1 -----
24        String ngcpos = "N" + gcpos;
25        gcode = "G00 " + xyz + " M05";
26        String sSQL = null;
27        String kom = "(Programmende, zur angegebenen Position fahren)";
28        sSQL = "INSERT INTO codelist (gcpos, listname, element, gcode, comment) " + "VALUES ( " + "'" + ngcpos + "'," + "'" + listname + "'," + "'" + el2 + "'," + "'" + gcode + "'," + "'" + kom + "'" + ")";
29        st.execute(sSQL);
30        int ipos = Integer.parseInt(gcpos);
31        ipos = (ipos + 1);
32        npos = (Integer.toString(ipos));
33        Start.POS.setText(npos);
34        //Satz 2 -----
35        ngcpos = "N" + npos;
36        gcode = "M30";
37        kom = "";
38        sSQL = "INSERT INTO codelist (gcpos, listname, element, gcode, comment) " + "VALUES ( " + "'" + ngcpos + "'," + "'" + listname + "'," + "'" + el2 + "'," + "'" + gcode + "'," + "'" + kom + "'" + ")";
39        st.execute(sSQL);
40        //Satz 0 -----
41        ngcpos = "%";
42        String el0 = "";

```

```
45         gcode = "";
46         kom = "";
47         sSQL = null;
48         sSQL = "INSERT INTO  codelist (gcpos, listname, element, gcode,      ↵
comment) " + "VALUES ( " + "'" + ngcpos + "'," + "'" + listname + "',"      ↵
+ "'" + el0 + "',"
49             + "'" + gcode + "'," + "'" + kom + "'" + ")";
50         st.execute(sSQL);
51         conn.close();
52         aktualTable();
53     } catch (ClassNotFoundException | SQLException e) {
54         JOptionPane.showMessageDialog(null, "Fehler in endCode(): " + e);
55     }
56 }
57
```

```

1 //*****
2 //Button GCode erzeugen - aus den in die Datenbank geschriebenen Zeilen wird
3 // die GCode-Datei erstellt, die Codelist kann dabei gelöscht werden
4     public static void exportNGC() throws SQLException, IOException {
5         String listname = Start.LIST.getText();
6         String gcode = Start.LIST.getText() + ".ngc";
7
8         JFileChooser chooser = new JFileChooser();
9         FileFilter filter = new FileNameExtensionFilter("GCode", "ngc");
10        chooser.setDialogTitle("Speicherort auswählen");
11        chooser.setCurrentDirectory(new File(home + "/GCodeDaten/codes/"));
12        chooser.addChoosableFileFilter(filter);
13        chooser.setFileFilter(filter);
14        chooser.setSelectedFile(new File(gcode));
15        String path = "";
16        int rueckgabeWert = chooser.showSaveDialog(null);
17        if (rueckgabeWert == JFileChooser.APPROVE_OPTION) {
18            path = chooser.getSelectedFile().getAbsolutePath();
19        }
20        Connection conn = DriverManager.getConnection(jhpn, uname, pw);
21        try {
22            Class.forName("org.postgresql.Driver");
23            CopyManager copyManager = new CopyManager((BaseConnection) conn);
24            File file = new File(path);
25            FileOutputStream fileOutputStream = new FileOutputStream(file);
26            //finally execute the COPY command to the file with this method:
27            copyManager.copyOut("COPY (SELECT gcpos, gcode, comment FROM codelist
                WHERE listname = " + "'" + listname + "'" + "ORDER BY lid) TO STDOUT ",
                fileOutputStream);
28            conn.close();
29            File f = new File(path);
30            // Check if the specified file
31            if (f.exists()) {
32                JOptionPane.showMessageDialog(null, "GCode wurde erstellt!");
33                //hier sed exec
34            } else {
35                JOptionPane.showMessageDialog(null, "GCode Erstellung
                fehlgeschlagen!");
36            }
37            int eingabe = JOptionPane.showConfirmDialog(null, "Wollen Sie die
                Codelist aus der Datenbank löschen?", "Codelist Löschen", JOptionPane.
                YES_NO_OPTION);
38            if (eingabe == 0) {
39                Class.forName("org.postgresql.Driver");
40                conn = DriverManager.getConnection(jhpn, uname, pw);
41                String sSQL = "DELETE FROM codelist WHERE listname = " + "'" +
                listname + "'" + " ";
42                Statement st = conn.createStatement();
43                st.execute(sSQL);

```

```
44         conn.close();
45     }
46 } catch (ClassNotFoundException | SQLException e) {
47     JOptionPane.showMessageDialog(null, "Fehler in exportNGC(): " + e);
48 }
49 }
50
```