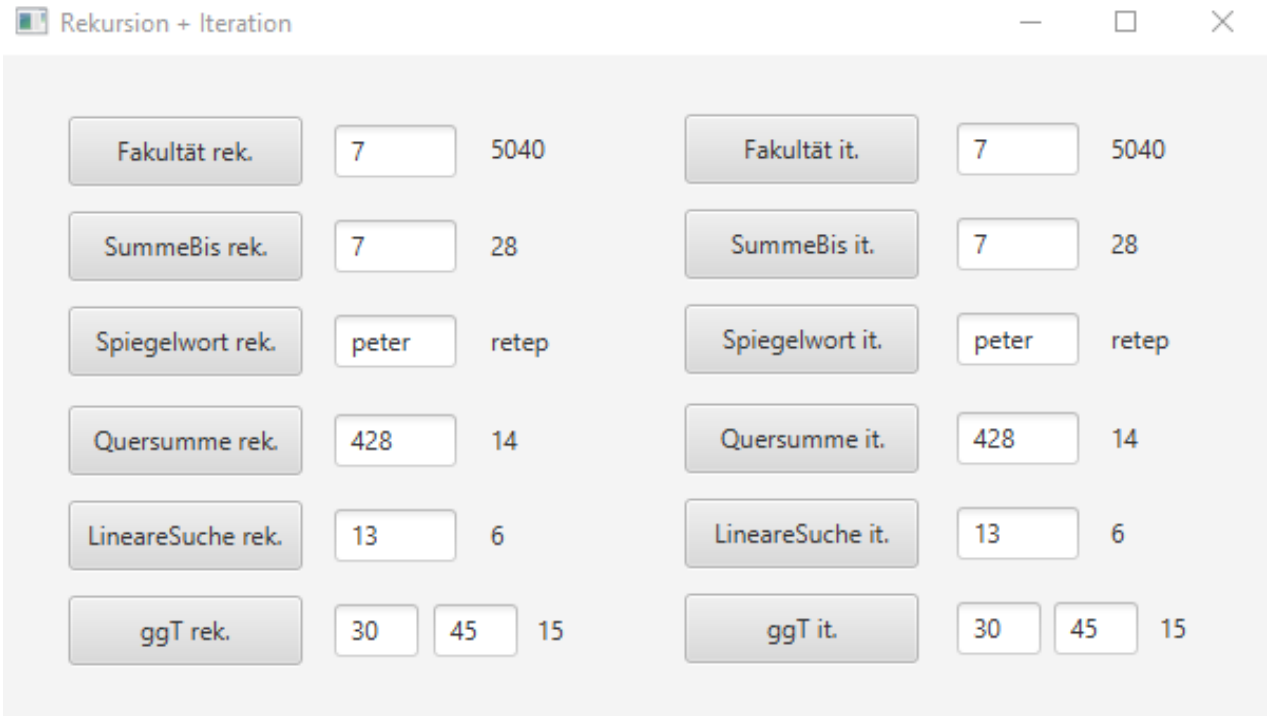
Rekursion in Java

Übungen an GUI

(inklusive iterativer Lösungen)




```
public class Controller {
   @FXML private Label lbFakultaetRek;  …    @FXML private TextField tfFakultaetRek; …
  …
   public void btSpiegelwortRek_onClick() {
      lbSpiegelwortRek.setText(Uebungen.SpiegelwortRek(tfSpiegelwortRek.getText()));
   }

   public void btSpiegelwortIt_onClick() {
      lbSpiegelwortIt.setText(Uebungen.SpiegelwortIt(tfSpiegelwortIt.getText()));
   }
  …
   public void btLineareSucheRek_onClick() {
      int[] a = {2,3,5,7,11,13,17,19};
      int b = Integer.parseInt(tfLineareSucheRek.getText());
      lbLineareSucheRek.setText(String.valueOf(Uebungen.LineareSucheRek(a,b,0)));
   }
  …
   public void btggTIt_onClick() {
      int a = Integer.parseInt(tfggTIt1.getText());
      int b = Integer.parseInt(tfggTIt2.getText());
      lbggTIt.setText(String.valueOf(Uebungen.ggTIt(a,b)));
   }
}
```

```
public class Uebungen {
 public static int FakultaetRek(int a) { if (a==1) return 1; else return a * FakultaetRek(a-1); }

 public static int FakultaetIt(int a) { int n=1; for (int i=2; i<=a; i++) n = n * i; return n; }

 public static int SummeBisRek(int a) { if (a==1) return 1; else return a + SummeBisRek(a-1); }

 public static int SummeBisIt(int a) { int n=1; for (int i=2; i<=a; i++) n = n + i; return n; }

 public static String SpiegelwortRek(String a) {
  if (a.length() == 1) return a;
  else return a.charAt(a.length()-1) + SpiegelwortRek(a.substring(0,a.length()-1));
 }

 public static String SpiegelwortIt(String a) {
  String s = new String(); for (int i=1; i<=a.length(); i++) s = s + a.charAt(a.length()-i); return s;
 }

 public static int QuersummeRek(int a) {
   if (a <= 9) return a;
   else return a%10 + QuersummeRek(a/10);
 }
```

123 % 10 = 3, 123 / 10 = 12
12 % 10 = 2, 12 / 10 = 1
1 % 10 = 1, 1 / 10 = 0

1+2+3 = 6

```
 public static int QuersummeIt(int a) {
  int summe = 0; while (a > 0) { summe = summe + (a % 10); a = a / 10; } return summe;
 }

 public static int LineareSucheRek(int[] a, int b, int c) {
  if (b==a[c]) return c+1; else if(c==a.length-1) return 0; else return LineareSucheRek(a,b,c+1);
 }

 public static int LineareSucheIt(int[] a, int b) {
  for (int i=0; i<a.length; i++) if(a[i]==b) return i+1;
  return 0;
 }

 public static int ggTRek(int a, int b) {
  if (a==0) return b;
  else if (a<b) return ggTRek(b-a, a);
  else return ggTRek(a-b, b);
 }

 public static int ggTIt(int a, int b) {
  int ggT=1;
  int c=0;
  if (a>b) c=b; else c=a;
  for (int i=2; i<=c; i++) if (a%i==0 && b%i==0) ggT=i;
  return ggT;
 }
}
```

105 − 90 = 15
90 − 15 = 75
75 − 15 = 60
60 − 15 = 45
45 − 15 = 30
30 − 15 = 15
15 − 15 = 0

ggT(105;90) = 15