



#### Lektion 6: Zeichnen

#### Übung 1: Grundformen

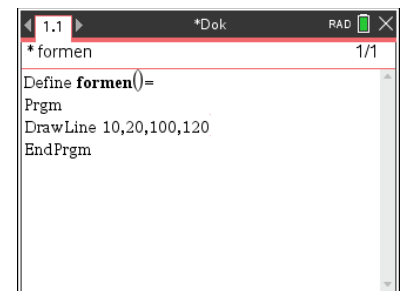
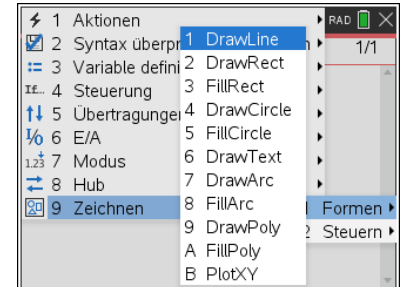
In dieser Übung wirst du lernen, wie man die Möglichkeiten von **Zeichnen** in der Programmierumgebung des **TI-Nspire™ CX II-T** nützt.

#### Lernziele:

- **Zeichnen** (*Draw*) einiger Formen
- Umgehen mit dem Zeichenschirm
- Ein Programm mit **Zeichnen**-Befehlen starten und beenden
- Ändern der Zeichenfarbe

#### Zeichnen einer Geraden

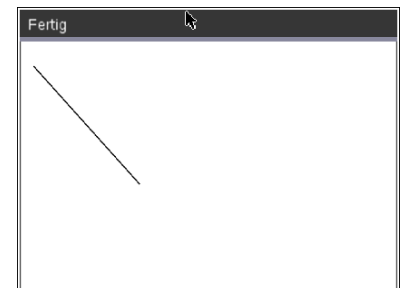
1. Beginn ein neues Dokument und öffne den Programmeditor. Gib dem Programm den Namen **formen**.
2. Drück **[menu]** und wähl **Zeichnen > Formen**. Hier siehst du die verfügbaren Formen (in englischer Sprache).
3. Wähl **DrawLine**.
4. Du musst die Endpunkte der Geraden,  $(x_1, y_1)$  und  $(x_2, y_2)$ , angeben.
5. Das Eingabeformat des Befehls ist:  
**DrawLine**  $x_1, y_1, x_2, y_2$
6. Beginn mit den folgenden Werten:  
**DrawLine 10, 20, 100, 120**



**Beachte:** Die **Zeichnen**-Befehle benötigen keine Klammern. Ihre Benützung führt zu einer Fehlermeldung.

7. Drück **[ctrl]** **[R]** um das Programm aufzurufen, und drück dann **[enter]** in der *Calculator App*, um es auszuführen.

Nun siehst du einen Schirm wie rechts abgebildet. Das ist ein eigenes „Zeichenfenster“ zur Ausgabe von **Zeichnen**-Befehlen. (10,20) ist der linke obere Endpunkt der Geraden und (100,120) der rechte untere. Die Titelleiste zeigt mit „Fertig“ an, dass das Programm beendet wurde. Der Rechner zeigt die Zeichnung weiterhin an. **Drück eine beliebige Taste, um das Zeichenfenster zu schließen und zur Calculator App zurückzukehren.**

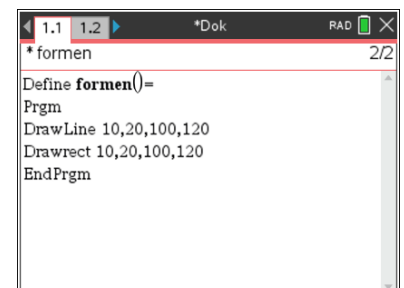


#### Zeichnen eines Rechtecks

8. Geh zum Programmeditor zurück und füg unter dem **DrawLine**-Befehl eine Programmzeile für das Zeichnen eines Rechtecks hinzu; wähl dazu die Anweisung **DrawRect** über **menu > Zeichnen > Formen**  
**DrawRect 10,20,100,120**

Nimm die gleichen Werte, die du für den **DrawLine**-Befehl verwendet hast.

*Kannst du dir vorstellen, was geschehen wird?*





## 10 Minuten Coding

### TI-NSPIRE™ CX II-T TECHNOLOGIE

9. Drück `ctrl` `R` und starte das Programm nochmals. Warum reicht die Gerade nicht bis in die untere rechte Ecke des Rechtecks?

Der **DrawRect**-Befehl hat das Eingabeformat:

**DrawRect**  $x, y, \text{Breite}, \text{Höhe}$

Welche *Breite* und *Höhe* müsste das Rechteck haben, dass die Gerade zur Diagonalen des Rechtecks wird?

*Sicherlich wirst du jetzt die Abmessungen des Schirms wissen wollen? Versuch vorher, mit einer Geraden eine Diagonale des Schirms zu zeichnen.*

#### Die Abmessungen des Bildschirms

- Die Zeichenfläche hat die Abmessungen 318 x 212 (Pixel = Bildpunkte).
- Die linke obere Ecke des Zeichenfensters hat die Koordinaten (0,0).
- Die rechte untere Ecke hat die Koordinaten (317,211).

10. Der Befehl **DrawLine 0, 0, 317, 211** zeichnet eine Bildschirmdiagonale.

11. Schreib den Befehl zum Zeichnen der anderen Diagonale.

#### Weitere Formen

**DrawCircle**  $x, y, r$  zeichnet einen Kreis mit dem Mittelpunkt in  $(x,y)$  und dem Radius  $r$ .

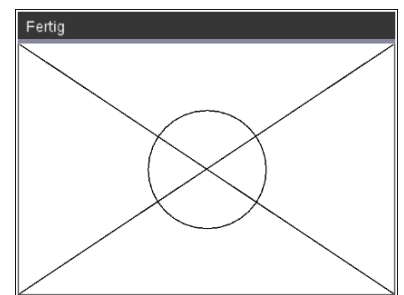
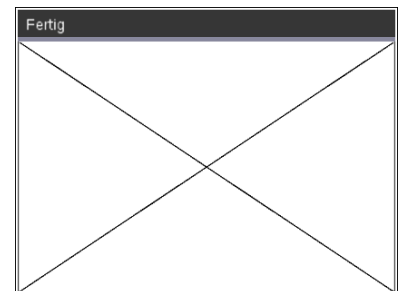
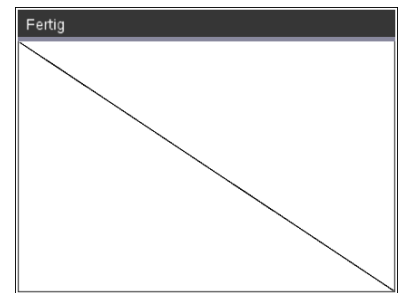
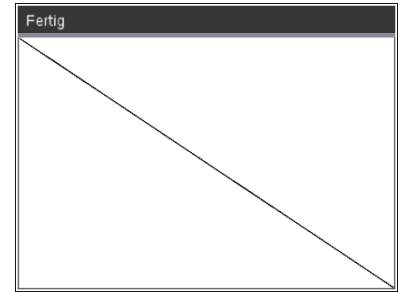
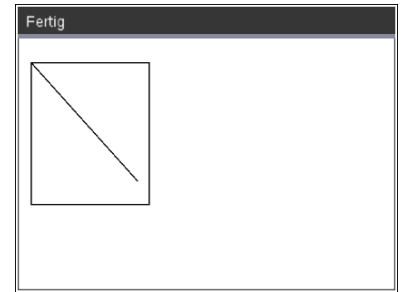
Beispiel:

12. **DrawCircle 159, 106, 50**

**FillRect** und **FillCircle** sind ähnlich; sie füllen die Formen mit der aktuellen Farbe.

## LEKTION 6: ÜBUNG 1

### SCHÜLERTÄTIGKEIT

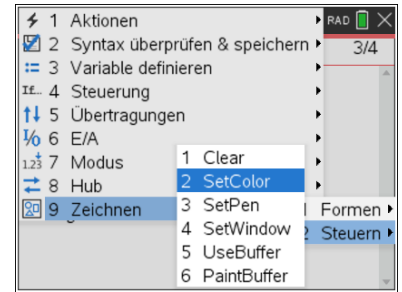




#### Ändern der Zeichenfarbe

13. Füg im Programmeditor zwischen zwei Zeilen den **SetColor**-Befehl ein:

**menu > Zeichnen > Steuern > SetColor**



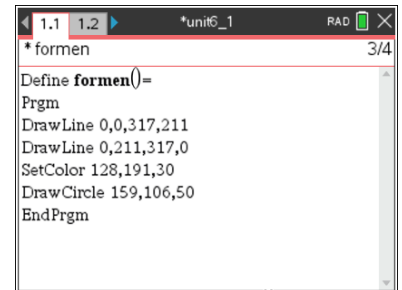
#### R, G, B

Die Zeichenfarbe wird durch drei Werte festgelegt: die Anteile von **Rot**, **Grün** und **Blau** werden gemischt. Jeder dieser Werte muss im Bereich zwischen 0 und 255 liegen.

**SetColor** *Rotwert, Grünwert, Blauwert*

Beispiel:

**SetColor 128, 191, 30**



14. Setz den **SetColor**-Befehl *vor* das Zeichnen des Objekts.
15. Versuch, ein Programm zu schreiben, das 100 zufällige Kreise in verschiedenen Farben zeichnen lässt. Verwende dazu eine **For-EndFor**-Schleife.

Wenn du die Kreislinie und die Kreisfläche in unterschiedlichen Farben zeichnen willst, dann musst du zwei Befehle verwenden:

<b>SetColor 255, 0, 0</b>	<i>Farbe Rot</i>
<b>FillCircle 100, 100, 50</b>	<i>die Kreisfläche</i>
<b>SetColor 0, 0, 0</b>	<i>Farbe Schwarz</i>
<b>DrawCircle 100, 100, 50</b>	<i>die Kreislinie</i>

... Das zeichnet einen schwarzen Kreis, der rot gefüllt ist. Aber, wenn du *zuerst* den Kreis zeichnest (*draw*) und *nachher* füllst (*fill*), wirst du den schwarzen Kreis nicht sehen, da er von der roten Fläche überdeckt wird. Versuch das!

