

# Unterprogramme: Prozeduren

---

## Projekt Olympische Ringe

**Zerlegung** einer komplexen Programmieraufgabe in **Teilaufgaben**

## Divide and Conquer

Kleinere, überschaubare **Unterprogramme (Prozeduren)** als

- **in sich geschlossene logische Einheit**, die eine bestimmte (grafische) **Teilaufgabe** löst.
- **Position und Größe** wird durch aufrufendes Programm bzw. Unterprogramm festgelegt und wird der Prozedur **als Parameter übergeben**
- Teilaufgabe ist durch grafische Darstellung gelöst  
**kein Rückgabewert**

# Unterprogramme: Prozeduren

---

## Projekt Olympische Ringe

**Zerlegung** einer komplexen Programmieraufgabe in **Teilaufgaben**

## Divide and Conquer

### Bsp.: Auto

Zerlegung in **Baugruppen**

- **Motor** (BMW, MTU, Daimler)
- **Räder** (Continental, Michelin)
- **Elektronik** (Licht, Blinker, etc.)

### Eigenschaften

- **Getriebe** (1,2,3,4,5 Automatic)
- **Licht** (Stand, Abblend, Fern)
- **Blinker** (links, rechts)

### Bsp.: Olympische Ringe

Zerlegung in **Prozeduren**

- **Kreis** (elementares Objekt)
- **Ringe** (aufgebautes Objekt)
- **Vollbild, Bewegung, Rotation**

### Parameter

- **Kreis** (Mittelpunkt, Radius)
- **Ringe** (Ankerpunkt, Radius)

# Unterprogramme: Prozeduren

---

## Projekt Olympische Ringe

**Zerlegung** einer komplexen Programmieraufgabe in **Teilaufgaben**

## Divide and Conquer

### Bsp.: Auto

Zerlegung in **Baugruppen**

Autofahrer muss die **Technik** nicht verstehen, um das Auto benützen zu können.

### Armaturenbrett

- **Getriebe** (1,2,3,4,5 Automatic)
- **Licht** (Stand, Abblend, Fern)
- **Blinker** (links, rechts)

### Bsp.: Olympische Ringe

Zerlegung in **Prozeduren**

Aufrufende Prozedur muss die **Implementation** der Prozedur nicht kennen!!

### Schnittstelle / Interface

- **Kreis** ( $X_m, Y_m, R$  : Integer)
- **Parameter** : Typ
- vgl. Auto: **D, S, B** : Automatik

# Unterprogramme: Prozeduren

## Projekt Olympische Ringe

**Deklaration der Prozedur** in der **Typdefinition der Schnittstelle**  
(vgl. Armaturenbrett beim Auto, Fernbedienung beim Fernseher)

```
type
  TFoOlympia = class(TForm)
  private
    procedure Kreis(Xm, Ym, R: Integer);
    procedure Ringe(X, Y, R: Integer);
    procedure Vollbild;
  public
  end;
```

**Übergabeparameter**  
und deren **Datentyp**

**Definition der Prozedur** im **Implementierungsteil der Unit**  
(vgl. Motorraum / Fernseher)

```
procedure TFoOlympia.Kreis(Xm, Ym, R: Integer);
begin
  Zeichne einen Kreis
      mit Radius R (Übergabeparameter R)
      um den Mittelpunkt (Übergabeparameter Xm, Ym)
end;
```

# Unterprogramme: Prozeduren

## Projekt Olympische Ringe

```
procedure TFoOlympia.Ringe(x, y, R: Integer);  
var Xm, Ym: Integer;  
begin
```

*Zeichne 5 Kreise mit gleichem Radius (Übergabeparameter R), deren relative Lage zum Ankerpunkt (Übergabeparameter X, Y) durch die lokalen Variablen (Xm, Ym) bestimmt wird.*

```
end;
```

```
procedure TFoOlympia.Vollbild;  
var i, x, y, xdir, ydir, r: Integer;  
begin
```

*Bewege das grafische Objekt **Ringe(x,y,r)** über den Bildschirm (**x:=x+xdir; y:=y+ydir**) längs der Richtung (xdir, ydir) und vergrößere es dabei (**r:=r+const**). Position (x,y), Richtung (xdir,ydir) und Größe (r) werden lokal durch Zufallsfunktion random(x) definiert*

```
end;
```