

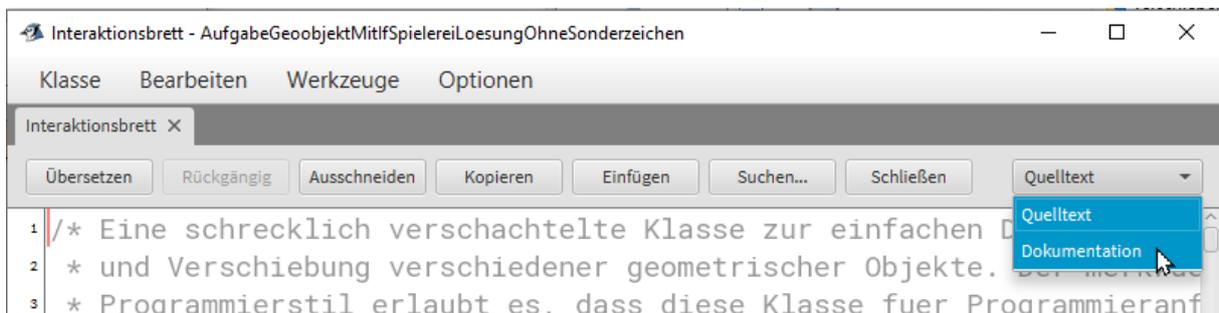
## Fragen, Antworten, Kommentare zur aktuellen Vorlesung

Denken Sie bitte daran, die beiden Lehrevaluationen zur Vorlesung und Ihrem Praktikum (Links waren in den E-Mails) auszufüllen.

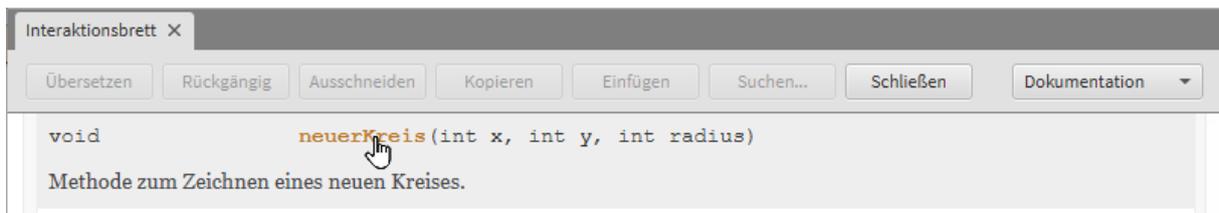
Frage: Ich male Kreise, diese sind aber an unerwarteten Positionen auf dem Bildschirm.

(Nach etwas Analyse) Sie müssen die Reihenfolge der Parameter beachten. Die Methoden zum Zeichnen erhält drei int-Werte und ohne Dokumentation kann nicht bestimmt werden, welcher Parameter welche Bedeutung hat.

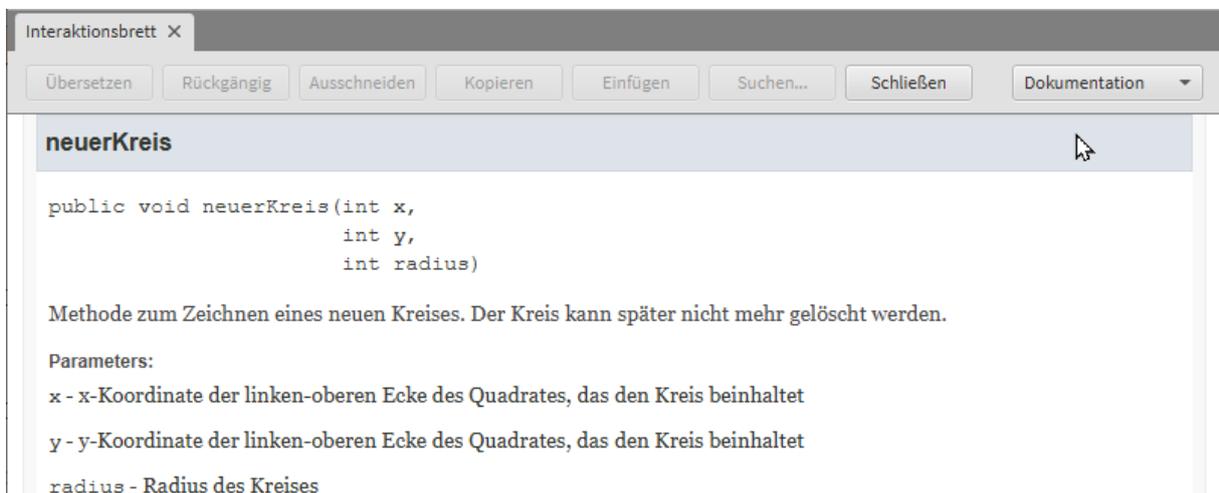
Die Dokumentation zum Interaktionsbrett ist von der Veranstaltungswebseite verlinkt. Wenn sich die Klasse im eigenen Projekt befindet, gibt es noch einen anderen Weg. Die Klasse wird geöffnet (den Code ist evtl. erst Ende des 4. Semesters verständlich) und rechts oben gibt es eine Drop-Down-Box, bei der auf „Dokumentation“ gewechselt werden kann.



Danach ist die JavaDoc-Dokumentation im HTML-Format sichtbar und die passende Methode suchbar (einfach nach unten zu Methoden scrollen).



Mit einem Klick auf den Methodennamen, wird zur detaillieren Dokumentation gesprungen.



Es ist erkennbar, dass der dritte Parameter der Radius ist. Das ist nebenbei Willkür der Entwicklung, der Radius könnte auch der erste Parameter sein. Wichtig ist es dann nur, dass bei anderen Methoden das gleiche Konzept bei der Reihenfolge der Parameter angewandt wird. Das hat bei der Entwicklung von Java an sehr wenigen Stellen nebenbei nicht geklappt. Sauberer wäre es, dass die Bedeutung der Parameter im Methodennamen erkennbar wäre, z. B. „zeichneNeuenKreisMitXundYPositionUndRadius“. Das wird seltenst gemacht, würde aber die Nutzung erleichtern. Bei einmal geschrieben und 1000-fach genutzt durchaus ein interessanter Gedanke. Damit solche langen Namen nicht eingetippt werden, haben fast alle Werkzeuge eine Code Completion, wird Variable und dann Punkt eingetippt und Strg (Ctrl) zusammen mit der Leertaste gedrückt, werden mögliche Methoden vorgeschlagen und sind mit den Pfeiltasten auswählbar.

```

26 // Kreis
27 void darstellen(Interaktionsbrett ib){
28     ib.

```

```

void    loescheObjekt(Object, String)
void    neueLinie(int, int, int, int)
void    neueLinie(Object, String, int,
void    neuerKreis(int, int, int)
void    neuerKreis(Object, String, int,
void    neuerPunkt(int, int)
void    neuerPunkt(Object, String, int,
void    neuerText(int, int, String)
void    neuerText(Object, String, int,
void    neuesRechteck(int, int, int, in
void    neuesRechteck(Object, String, i

```

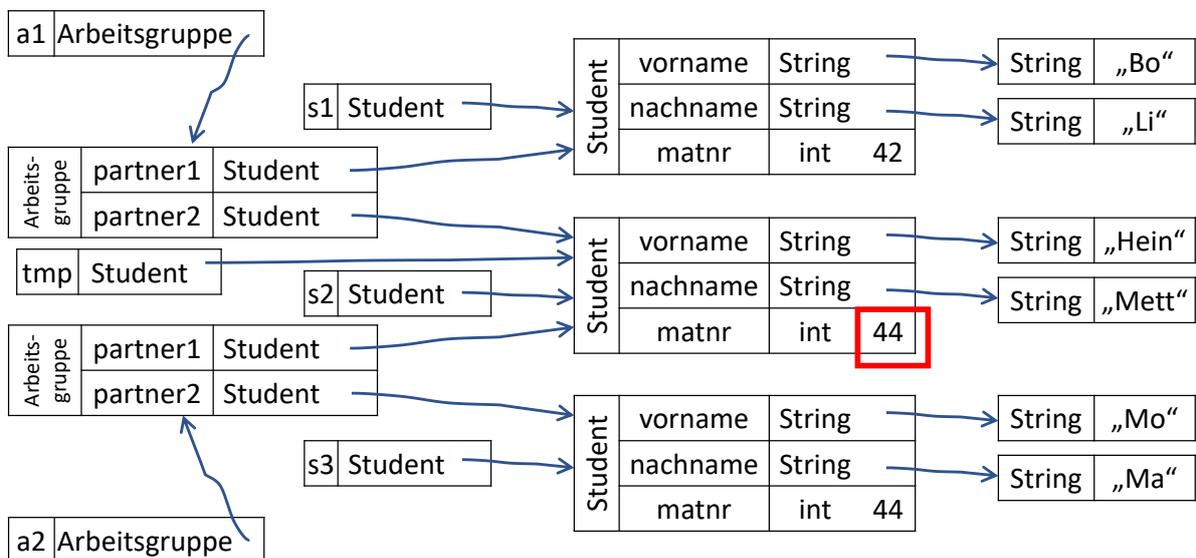
void **neuerKreis**(int x, int y, int radius)

Methode zum Zeichnen eines neuen Kreises. Der Kreis kann später nicht mehr gelöscht werden.

**Parameters**

- x - x-Koordinate der linken-oberen Ecke des Quadrates, das den Kreis beinhaltet
- y - y-Koordinate der linken-oberen Ecke des Quadrates, das den Kreis beinhaltet
- radius - Radius des Kreises

Da es einige Ungenauigkeiten bei den Lösungen zu Aufgabe 16 gab, hier das Bild am Ende der Methode. Die Pfeile zeigen immer auf den gesamten Kasten, nicht einen Teil des Kastens. Es damit egal, wo am Rand die Pfeilspitze andockt. Die Dreiteilung sollte immer erkennbar sein: Variablenname, Typ, Referenz oder Wert. Bei Objekten wird zusammenfassend am Rand der Klassenname angegeben.



Frage: Ich habe mal C++ gemacht, da würde ein ähnliches Programm doch Kopien erzeugen?

Antwort: Das stimmt, C++ unterstützt direkt zwei Wege, es kann entweder mit Objektreferenzen sehr ähnlich zu Java gearbeitet werden, was an der Nutzung der Operatoren \*, &, -> erkennbar ist, oder es werden immer vollständige Kopien von Objekten bei Zuweisungen und Methodenaufrufen erzeugt. In Java wurde der effizienteste Weg mit den Referenzen ausgewählt und die Operatoren in der Syntax weggelassen. Python arbeitet z. B. standardmäßig mit Kopien, Go unterstützt beide Ansätze. Generell haben beide Ansätze Vor- und Nachteile und können bezüglich der Funktionalität gegeneinander ersetzt werden und müssen von in der Software-Entwicklung tätigen Personen im Detail verstanden werden.