

```
operation == "MIRROR_X":  
    mirror_mod.use_x = True  
    mirror_mod.use_y = False  
    mirror_mod.use_z = False  
operation == "MIRROR_Y":  
    mirror_mod.use_x = False  
    mirror_mod.use_y = True  
    mirror_mod.use_z = False  
operation == "MIRROR_Z":  
    mirror_mod.use_x = False  
    mirror_mod.use_y = False  
    mirror_mod.use_z = True
```

```
@selection at the end - add  
mirror_ob.select= 1  
modifier_ob.select=1  
context.scene.objects.active  
= ("Selected" + str(modifier_ob.name))  
mirror_ob.select = 0  
= bpy.context.selected_objects  
data.objects[one.name].select  
print("please select exactly 1")
```

----- OPERATOR CLASSES -----

```
bpy.types.Operator):  
    X mirror to the selected  
    object.mirror_mirror_x"  
    mirror X"
```



GitHub
Copilot

04. Juli 2024

Agenda

- Willkommen & Intro
- Einführung Copilot - How to
- Die Aufgabe vorstellen
- Los geht's - let's play
- Diskussion und Learnings
- Aperero



GitHub
Copilot

Intro

- Wer hat Co-Pilot oder sonstige ChatBot schon in der Vergangenheit genutzt?
- CoPilot Integration in Confluence zur Erstellung von Dokumentation
z.B. [Coding Dojo Juni 2024 - CoPilot - GF STE - Bedag Wiki](#)

- Github CoPilot zur Unterstützung in der Applikations Entwicklung

eine spannende Aufgabe lösen und herausfinden, wie die Lösungsfindung mit GitHub Copilot optimiert werden kann. Der Fokus soll dabei weniger auf dem Problem an sich liegen, sondern vielmehr auf der effektiven Nutzung von Copilot.

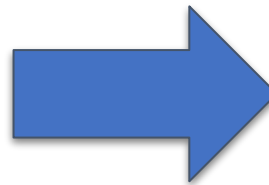
Faster and smarter code autocomplete

Get inspiration and simple solutions quickly

Explain and document code

Explain error messages

Generate tests



steigern von Tempo und Qualität
und somit die Produktivität

Einführung - how to Copilot



Einführung - how to Copilot

Optionen:

- Code Completion
<https://docs.github.com/en/copilot/using-github-copilot/using-github-copilot-code-suggestions-in-your-editor>
- Chat
<https://docs.github.com/en/copilot/github-copilot-chat/copilot-chat-in-ides/using-github-copilot-chat-in-your-ide>

```
public class CoPilot { no usages

    // method to compare two integers
    public boolean compare(int a, int b)
        return a == b;
}
```

Next

Previous

Accept

Accept Next Word

Accept Next Line

Copilot: Open Completions

Alt+]

Alt+[

Tab

Ctrl+Right

Ctrl+Alt+Right

```
// method to compare two integers
public boolean compare(int a, int b) { no usages
    return a == b;
}
```

GitHub Copilot

> 4 steps completed successfully

```
import org.junit.jupiter.api.Test;
import static org.junit.jupiter.api.Assertions.*;

class CoPilotTest {

    @Test
    void compareShouldReturnTrueWhenNumbersAreEqual() {

    }
}
```

CoPilotCopy.java ...adapter\src\main\java\ch\bedag\ste

CoPilot.java x

Die Aufgabe

Slides und Aufgabe:

<https://github.com/popavel/codo>

3 Algorithmen:

- Koordination zwischen mehreren Applikationen
- Sortieren mit Rahmenbedingungen
- Kleine unterhaltsame Aufgabe für alle, die schnell genug waren, um alles andere zu lösen

Die Aufgabe



Diskussion und Learning

- Wie hast du CoPilot genutzt und was hat dir geholfen?
- Wo siehst du Risiken, wo Chancen?
- Wie wirst du CoPilot bei deiner Arbeit mit einbeziehen?
- Wozu wird Copilot künftig fähig sein, was denkst du?
- Weitere Schulungsmöglichkeiten:
 - Udemy
[Suchergebnisse | Udemy](#)
 - Externer Tageskurs Sept/Nov mit Rainer Stropek
bzw. Inhouse bei grosser Nachfrage
[202403 Produktiver programmieren mit Github Copilot und ChatGPT - Education - Bedag Wiki](#)



Apero