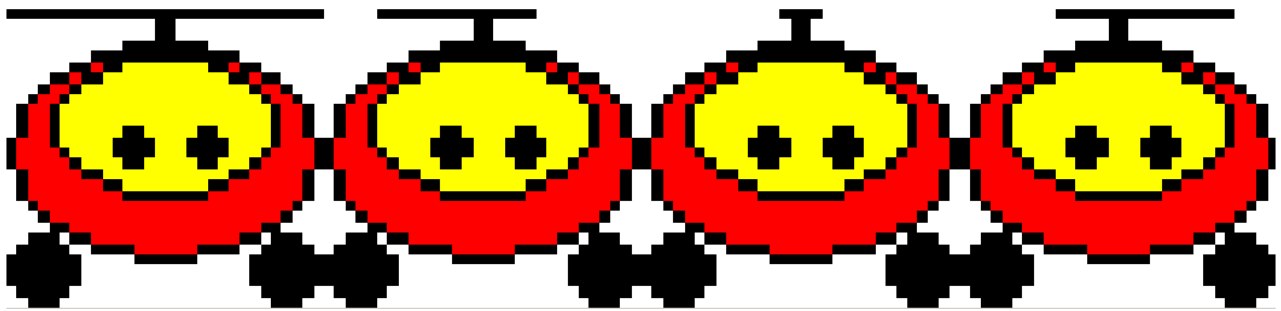


Spiel mit Programmieren mit Java

für Anfänger

unter Linux Fedora Gnome



Schule
Auftragnehmer
Auftraggeber
Bildungsgang
Klasse
Fach
Bearbeitungszeitraum

Johann-Conrad-Schlaun-Gymnasium
Dean Schneider
Herr M. Schmidt
Berufliches Gymnasium Technik
GYTU
Informatik
18.08.2023 – 05.02.2024

Inhaltsverzeichnis

Benötigte Programme installieren	4
Erste schritte	5
1. Projekt erstellen	5

Dokumentationsaufbau:

- Deckblatt
 - Bildüberthema
 - Beschreibung
 - Thema
 - Auftragnehmer
 - Auftraggeber
 - Schule
 - Bildungsgang
 - Klasse
 - Fach
- Inhaltsverzeichnis
- Einführung in das Thema
 - Installation benötigter Software
- Voraussetzung für Umsetzung des Tutorials
-

Erste schritte

1. Wichtige hinweise

Fedora, das Betriebssystem, und alle Programme sind auf Englisch. Wenn man etwas online sucht, sucht man besser auf Englisch, weil es viel mehr Ergebnisse gibt. Die Anleitungen haben ihr System dann auch auf Englisch, was es einfacher macht ihnen zu folgen.

2. Benötigte Programme installieren

Programme über den Appstore:

- Office z.B. LibreOffice
- IDE (Entwicklungsumgebung) z.B. NetBeans
- Java Runtime/Development JDK
- UML Diagramme z.B. Dia

Ich benutze NetBeans IDE 18 anstatt von 8.2, weil es für Linux scheinbar keinen 8.2 Download mehr gibt.

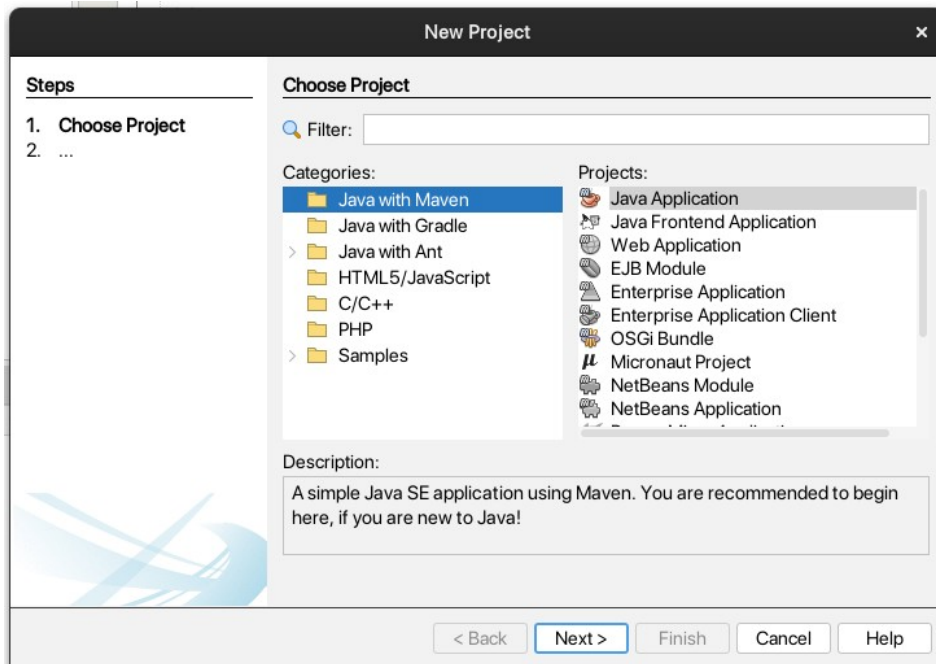
```
sudo dnf install maven
```

3. Projekt erstellen

1.



2.



3. Project Name: INF_GamePanel
Project Location: an einem gutem Ort speichern
Group Id: de.inf
Create main class: Ja

4. Die .java Datei sollte so aussehen:

```
package de.inf.inf_gamepanel;

public class INF_GamePanel
{
    // Bei main() beginnt jedes Programm
    public static void main(String[] args)
    {
        /**
         * Hier wird „Hello World“ in der Console ausgegeben
         * ln (line) steht für neue Zeile nach der ausgabe
         */
        System.out.println("Hello World");
    }
}
```

5. Den Code Kompilieren (Startbar machen). Öffne zuerst den Terminal in dem Projektordner.

```
mvn clean package
java -jar target/[JAR_FILE]
```

Keywords

;	Befehl ende – beendet ein Befehl
//	Kommentar – Diese Zeile wird vom Programm ignoriert
/** **/	Kommentar Block – Kommentar über mehrere Zeilen
{ }	Block/Scope – In dem Scope erstellte Variablen werden bei dem Ende gelöscht

Syntax

Curly bracke

```
public class  
{  
}
```