

## **Mobiles Endgerät**

### **Aufgabenstellung**

Das Projekt Smartphone Im Rahmen dieses Projekts sollt ihr selbständig in Zweier-Gruppen euer bisheriges Wissen in der Java-Programmierung demonstrieren. Was sollen wir tun? Aufgabe: Das Projekt besteht aus zwei Teilen. Für jeden Teil müsst ihr zunächst das BlueJ-Projekt Figuren herunterladen, unter einem geeigneten Namen speichern und eine neue Klasse erstellen. Diese soll je nach bearbeitetem Teil Folgendes enthalten:

1. Für ein Smartphone sollen folgende Funktionen vorhanden sein: Bildschirmschoner (auf dunkelblauem Hintergrund sollen 4 hellblaue Luftblasen nach oben steigen), Ein- / Ausschalten (ist der Bildschirm schwarz (aus), soll er weiß (ein) werden und umgekehrt), Durchsuchen des Telefonbuches (z. B. Speichern aller Telefonnummern in einem Feld; Durchsuchen des Feldes nach einer Nummer (z. B. `meinFeld[i].contains("0049176")` und Rückgabe des Feldeintrags `i`). Außerdem soll das Smartphone noch eine weitere Fähigkeit eurer Wahl besitzen, z. B. zweiten Screensaver, Blockieren von Auslandsanrufen, Demolieren des Displays o.ä.

2. Im zweiten Teil könnt ihr eurer Fantasie freien Lauf lassen. Dabei könnt ihr euch von den bisherigen Programmierungen in der Schule oder vom ersten Teil leiten lassen oder euch etwas völlig Neues überlegen. Der Umfang sollte in etwa dem ersten Teil entsprechen. Was muss abgegeben werden? Bestandteile: Es muss ein gepackter Ordner mit drei Teilen abgegeben werden: • Ein Textdokument (plattformunabhängig abgespeichert, d. h. beispielsweise als html- oder pdf-Datei, nicht doc- oder docx-Datei!) mit der Aufgabenstellung, den „Autoren“ des Projekts, einer Beschreibung der einzelnen Teile, einer Klassenkarte und dem Struktogramm einer beliebigen Methode. • Ein BlueJ-Projekt für den ersten Teil (Figur) • Ein BlueJ-Projekt für den zweiten Teil (Kreativteil) Wie wird das Projekt bewertet? Kriterien: • Gestaltung, Umfang und Überschaubarkeit des Textdokuments • Größe der Klassenkarte und des Struktogramms • Übersichtlichkeit und Dokumentation der Programmierung • Geeignete Verwendung von Wiederholungen, bedingten Anweisungen, Feldern, Ein- und Rückgabewerten (dreifache Wertung) • Umfang und Originalität der Aufgabenerledigung Wie installiere ich BlueJ zu Hause? Damit BlueJ zu Hause läuft, muss die BlueJ- (inkl. JDK-)Software installiert werden: <http://www.bluej.org/> Wie packe ich die abzugebenden Dateien? Normalerweise ist inzwischen bei allen PCs „Senden An ZIP-komprimierten Ordner“ möglich. Sollte dies nicht gehen, kann man sich kostenfrei 7zip unter folgender Adresse herunterladen: <https://www.giga.de/p/7-zip>

### **Autoren**

Wiegand Einar Christian Kretschmer, Maximilian Rene Stefan und Johannes Florian Christian Preißinger

## **Bedienungsanleitung des Skripts und des Telefons**

### **Benutzung**

1. Download des .zip files von <https://github.com/Winux-Arch/Mobilesendgeraet>
2. Ausführen der Datei "main.exe"

3. Benutzung der Software (falls Bildschirm nicht weiß, wird mehrmals Option 1/Einschalttaste in der Konsole eingeben.)

### **Telefon**

Anleitungen wie in der Konsole beschrieben folgen.

Eine Ausgabe erfolgt in der Konsole und eventuell im Ausgabefenster.  
Bei Eingabe kann sowohl die Zahl vor der Funktion als auch der Text nach der Zahl angegeben werden. Groß- und Kleinschreibung sind hierbei nicht zu beachten.

Sollte das Telefonbuch genutzt werden, ist bei der Suche Groß- und Kleinschreibung trivial; bei Eingabe eines neuen Eintrags werden keine automatischen Modifikationen vorgenommen.

Einträge im Telefonbuch können nur manuell geändert werden.

## **Beschreibung der einzelnen Teile**

### **Main.py**

Z.0-13: Importierungen und Initialisierung

Z.14-28: Konsole wird gestartet und Handy erstmalig gemalt

Z.32-36: Überprüft, ob App geschlossen wird

Z.41-43: Zeigt Optionen und fragt nach einer Eingabe davon

Z.52-55: Schaltet Telefon ein bzw. aus

Z.60-61: Zeichnet das Display als schwarzes bzw. graues Rechteck

Z.70-80: Öffnet den Bildschirmschoner mit ruhigeren Farben und animierten Blasen, die sich nach oben bewegen

Z.87-129: Sucht bzw. fügt je nach Eingabe Einträge hinzu bzw. ruft an

Z.133-137: Error Catcher falls Option 1 nicht gewählt

### **Obj.py**

Z.1-9: Importierungen

Z.9-14: Das Handy an sich

Z.19-20: Gibt benötigte Werte, die beim Einschalten verwendet werden

Z.25-26: Gibt benötigte Werte, die beim Ausschalten verwendet werden

Z.30-42: Gibt benötigte Funktionen und Attribute für den Bildschirmschoner im Handy

Z.35-36: Farben des Hintergrundes und der Blasen

Z.41-42: Farbe des Hintergrundes wird zurückgegeben

Z.47-88: Telefonbuch, welches im Handy verwendet werden kann

Z.55-65: Funktionierendes Durchsuchen des Telefonbuches

Z.70-75: Eingabe neuer Werte ins Telefonbuch

Z.79-84: Schreibt Text langsam und jeden Buchstaben einzeln

Z.87-88: Leert die Konsole

Klassenkarte

Handy
randfarbe = Triple anfarbe = Triple ausfarbe = Triple homefarbe = Triple
einschalten() ausschalten()

Bildschirmschoner
hintergrund = Triple blasen = Triple
aktivieren()

Telefonbuch
tbpath = String
suche(name, nummer) eingabe(name, nummer)

Func
slowprint(text, typing_speed) clearconsole()

Struktogramm

