

## **Aufgabe 19.11.2024 Strukturierung und Implementierung**

### **Teil 1 der Aufgabe: Strukturierung: (Folie 15)**

- Überlege und notiere dir genau wie du die einzelnen Punkte implementieren könntest und auf was du achten musst bei der Implementierung.  
Was wird wichtig sein? Wie kannst du eine passende Lösung dafür finden?  
Strukturiere dein Vorgehen, so wie du konkret vorgehen würdest.
- Schreibe zu jedem der 6 Punkte mindestens 3 Sätze. (**Wichtig:** Lade diese Word Datei herunter, bearbeite sie schriftlich und lade diese Word Datei auch als Lösung bei Teams hoch!)

### **Teil 2 der Aufgabe: Implementierung von Punkt 1-3: (Folie 28)**

Implementiere die fehlenden Funktionen der Klasse *Board*. Strukturiere vorher dein Vorgehen.

Zusätzlich brauchen wir noch folgende Methoden für unser Board:

- Eine Methode zum Platzieren von Schiffen (**Beachte:** Wie beuge ich vor, dass Schiffe über den Spielfeldrand kommen?)
- Eine Methode zum Schießen und Markieren von Hits/Misses (**Tipp:** Hierfür benötigt Ship auch noch eine zusätzliche Methode)
- Eine Methode zum Darstellen in der Konsole des Spielfelds
- Eine Methode zum Hinzufügen eines Schiffes (**Tipp:** Es wird ein eigener Array Ships benötigt.)

Pflicht sind **zwei der vier** Methode zu schreiben. Allerdings benötigen wir für die Funktionalität später alle Methoden!

## Kontaktsystem – Zu Aufgabe 1:

Erstelle eine Anwendung zur Verwaltung von Kontakten. Jeder Kontakt besteht aus einem **Namen**, einer **Telefonnummer** und einer **E-Mail-Adresse**. Du sollst eine **ArrayList** von Kontakten verwenden, um diese zu speichern und mit ihnen zu interagieren. Die Anwendung soll die folgenden Funktionen bieten:

1. Kontakt hinzufügen:

- Der Benutzer kann einen neuen Kontakt mit einem Namen, einer Telefonnummer und einer E-Mail Adresse hinzufügen.

**Beschreibe dein Vorgehen:**

2. Kontakte anzeigen:

- Der Benutzer kann alle gespeicherten Kontakte anzeigen, wobei jeder Kontakt mit seinem Namen, der Telefonnummer und der E-Mail-Adresse angezeigt wird.

**Beschreibe dein Vorgehen:**

3. Kontakt suchen:

- Der Benutzer kann nach einem Kontakt anhand des Namens suchen. Die Anwendung soll den gefundenen Kontakt anzeigen oder eine Nachricht anzeigen, wenn der Kontakt nicht existiert.

**Beschreibe dein Vorgehen:**

4. Kontakt bearbeiten:

- Der Benutzer kann die Telefonnummer oder E-Mail Adresse eines bestehenden Kontakts ändern.

**Beschreibe dein Vorgehen:**

5. Kontakt löschen:

- Der Benutzer kann einen Kontakt nach dem Namen löschen.

**Beschreibe dein Vorgehen:**

6. Daten persistieren:

- Die Liste der Kontakte soll nach dem Beenden der Anwendung in einer Datei gespeichert und beim nächsten Start geladen werden

**Beschreibe dein Vorgehen:**