

Exercises**W06H01 - Fisch fuer die Pinguine**

Points	Submission due date	Status	Difficulty	Categories
0 / 5	in 6 days	No graded result		

Tasks:

Falls du die Aufgabe vor 18:00 Uhr geklont hast und einen Gradle-Fehler erhältst, klon das Repository bitte erneut. (Reset des Repositories ganz unten)

Fisch für die Pinguine

In dieser Aufgabe geht es um Linked-Listen und Sortieren.

Die Pinguine beschäftigen schon seit einer Woche mit Listen, und sie brauchen erst mal Verstärkung in Form von Fisch, bevor sie sich mit den neuen Datenstrukturen beschäftigen können. Als allererstes dürfen sich die jüngsten Pinguine bedienen und deshalb müsst ihr ihnen jetzt helfen, sich auf pinguingerechte Art zu sortieren. Das heißt, dass die Pinguine sich eins nach dem anderen einsortieren werden.

Part 1: An die richtige Stelle

Wir werden später die Alter von Pinguine in einer Liste speichern. Wir ergänzen aber erst mal die bestehende List-Implementierung.

In [W05P02](#) wurde die `add(int value, int index)` Methode implementiert. Jetzt fügen wir die `insertIntoSorted(int value)` Methode hinzu. Hier wird angenommen, dass die Liste bereits sortiert ist, und ein Element an der richtigen Stelle hinzugefügt wird. Das geht so:

1. Ist die Liste leer, wird einfach ein Element mit dem Value hinzugefügt. Das läuft analog zu `add`, wenn die Liste leer ist.
2. Ist die Liste nicht leer, wird immer das nächste Element angeschaut, bis ein Element gefunden wird, dessen Wert größer ist. Dann wird das neue Element davor hinzugefügt.
 1. Es kann sein das man kein größeres Element findet. Das merkt man unter anderem daran, dass das `.next` auf das aktuell angeschaute Element `null` wird. In dem Fall wird das neue Element ganz am Ende hinzugefügt.
 2. Es kann sein das bereits das erste Element größer ist.
 3. Man kann sich auch einfach das Index von dem letzten kleineren Element merken, und die `add` Funktion verwenden. Dann ist man sicher, dass die richtige Änderungen in der Liste gemacht werden.

Um diese Methode zu testen, empfiehlt sich eine sortierte Liste zu erstellen, indem ihr erst eine leere Liste erstellt, und eins nach dem anderen Elemente in aufsteigender Reihenfolge mit `add` hinzufügt. Danach könnt ihr einzelne Elemente per `insertIntoSorted` hinzufügen und schauen ob die Liste weiterhin richtig sortiert ist.

Wenn das alles funktioniert, sind die Pinguine bereit für die nächste Aufgabe!

1.  **Insert into sorted** No results
Implementiere die Methode `insertIntoSorted` in der Klasse `List`.

Part 2: Alle an die richtigen Stellen

Nun wollen wir die Pinguine nach ihrem Alter sortieren. Dabei fangen wir mit einem Array von Alter an und gehen wie folgt vor:

1. Eine leere Liste wird erstellt.
2. Alle Pinguinalter werden eins nach dem anderen an die richtige Stelle mit `insertToSorted` hinzugefügt. Dadurch bleibt die Liste immer sortiert.

Hinweis: Funktioniert das nicht wie erwartet, ist es sicher eine gute Idee, sich die Liste nach jedem hinzufügen ausdrucken zu lassen und zu schauen, wann was schief läuft.

2.  **Sort Penguins** No results
Implementiere die Methode `sortPenguins` in der Klasse `SortedPenguins`.

Sind alle Pinguine satt, geht es weiter mit der nächsten Hausaufgabe!

Exercise details

Release date Nov 28, 2025 17:00

Submission due date Dec 7, 2025 17:00

Complaint possible No