

Dokumentation Programmentwurf

Matrikelnummer: 1788307

Algorithmus

Rekursive Bestimmung von Primzahlen.

- Ausgelagert in MyPrimeHelper.cs
- Überladen von istPrimzahl
- **istPrimzahl(int zahl)** prüft, ob Übergabe Zahl kleiner gleich 1 ist
 - ja: Keine Primzahl
 - nein: Beginn rekursive Primzahlbestimmung => Aufruf istPrimzahl(int, int)
- **istPrimzahl(int zahl, int temp)** prüft, ob gesuchte zahl durch temporäre zahl ohne Rest teilbar ist
 - ja: wenn (gesuchte zahl == temporäre zahl) => keine Primzahl sonst Primzahl
 - nein: Aufruf der eigenen Funktion mit temporäre zahl +1
- Bestimmung der Primzahlen in einem Container mit einer Schleife über alle Elemente und dem Aufruf von istPrimzahl für jedes Element

Containerverwendung

Verwendung einer abgeleiteten Liste und einer abgeleiteten Queue

- Interface MyIdentify => Id
- Container können nur Objekte aufnehmen, die das Interface MyIdentify implementieren.
- MyList.cs
 - Erbt von Liste
 - Implementiert MyIdentify
 - Besitzt Methode getPrimes() zur Primzahl Bestimmung
- MyQueue.cs
 - Erbt von Queue
 - Implementiert MyIdentify
 - Besitzt Methode getPrimes() zur Primzahl Bestimmung

Erzeugung Eingangsdaten

- Daten werden in einer CSV-Datei gespeichert
- Trennung durch Semikolon
- Einlesen der Dateien beim Programmstart

Limitationen der Eingangsdaten

Die Id der Objekte muss zwischen 0 und 999 liegen, sonst individueller Fehler

Pokemon:

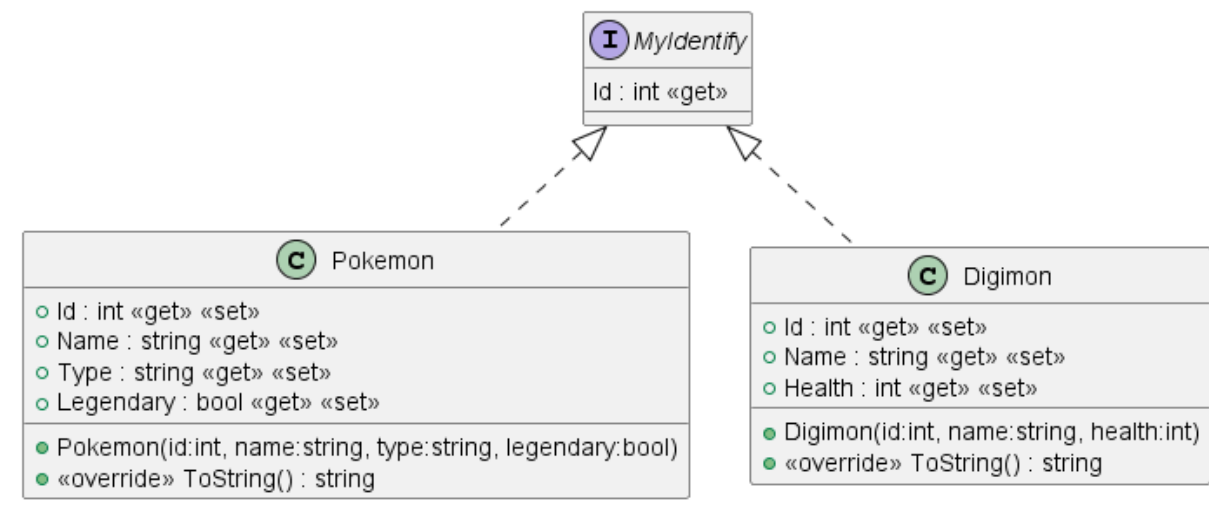
Spalte 1	Spalte 2	Spalte 3	Spalte 4
Id	Name	Type	Legendary
integer	string	string	boolean

Digimon:

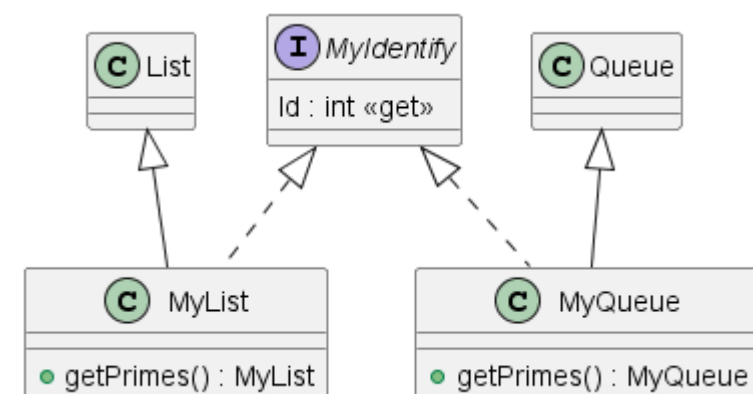
Spalte 1	Spalte 2	Spalte 3
Id	Name	Health
integer	string	string

Klassendiagramme

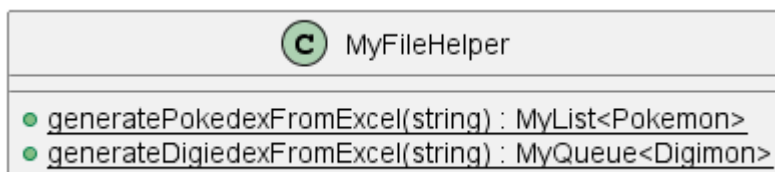
Objekte:



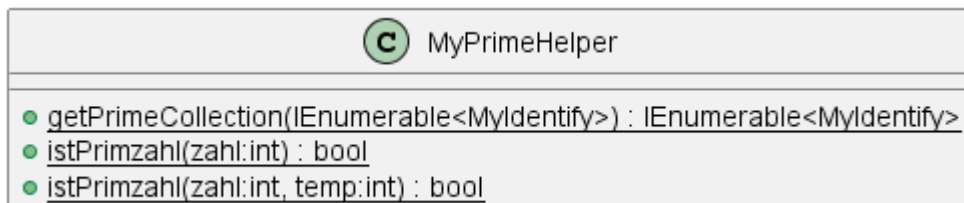
Containerklassen:



Hilfsklasse zum Dateiauslesen:



Hilfsklasse für Primzahlrechnung



Eigene Exception:

